

A Dissertation
On
Mathematical Models of Nature Inspired
Computational Intelligence for Terrain
Understanding

Submitted in fulfillment of the requirements of the degree of

Doctor of Philosophy
(Computer Engineering)

By

LAVIKA GOEL

(University Roll No. 04 /Ph.D./CS/2010)

Under the Joint Supervision of:

Prof. Daya Gupta,

Professor, Dept. of Computer Engineering, Delhi Technological University (DTU)

And

Prof. V.K. Panchal,

Director, SBIT, Sonipat

&

(Retd.) Scientist-'G' and Associate Director,

Defense & Terrain Research Lab,

Defense & Research Development Organization (DRDO)



To

DEPARTMENT OF COMPUTER ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

NEW DELHI - 110042

2015

DECLARATION

I, Lavika Goel, student of Ph.D. (roll no. 04/PhD/CS/2010), hereby declare that the thesis entitled “*Mathematical Models of Nature Inspired Computational Intelligence for Terrain Understanding*” which is being submitted for the award of the degree of doctor of philosophy in Computer Engineering, is a record of bonafide research work carried out by me in the Department of Computer Engineering, Delhi Technological University in collaboration with Defense & Terrain Research Laboratory, Defense & Research Development Organization (DRDO), MetCalfe House, Delhi.

I further declare that the work presented in the thesis has not been submitted to any University or Institution for any degree or diploma.

Date: _____

Place: New Delhi

Lavika Goel

(Candidate)

04/PhD/CS/2010

Department of Computer Engineering,
Delhi Technological University (DTU),
New Delhi - 110042.



DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

CERTIFICATE

Date: _____

This is to certify that the work embodied in the thesis entitled ***“Mathematical Models of Nature Inspired Computational Intelligence for Terrain Understanding”*** submitted by Ms. ***Lavika Goel*** with ***roll no. 04/PhD/CS/2010*** as a ***full-time*** research scholar in the Department of Computer Engineering is an authentic work carried out by her under our guidance and is submitted to Delhi Technological University for the award of the Degree of **Doctor of Philosophy**.

The work is original and has not been submitted, in part or full, to any other University or Institution for the award of any other degree.

Supervisors

Prof. Daya Gupta,
Department of Computer Engineering,
Delhi Technological University (DTU),
New Delhi - 110042.

Prof. V.K. Panchal,
Director, SBIT, Sonipat &
(Retd.) Scientist-‘G’ and Associate
Director, Defense & Terrain Research
Laboratory (DTRL),
Defense & Research Development
Organization (DRDO),
New Delhi -110054.

ACKNOWLEDGEMENT

I address my sincere thanks to Almighty God for giving me the inner strength to complete my dissertation. He was always there protecting and saving me.

It is a great pleasure to have the opportunity to extend my heartiest felt gratitude to everybody who helped me throughout the course of this research. It is a distinct pleasure to express my deep sense of gratitude and indebtedness to my learned supervisors Prof. Daya Gupta and Prof. V.K. Panchal for their invaluable guidance, encouragement and patient reviews. This research has been a dedicated effort towards the development of highly autonomous artificially intelligent systems which primarily would not have been possible at the first place without the apt guidance of **Prof. Daya Gupta (Professor, Dept. of Computer Engineering)** and **Prof. V.K. Panchal (Director, SBIT, Sonipat & (Retd.) Associate Director and Scientist-‘G’, DTRL, DRDO)**. Their motivation and encouragement during the research made me go further than expected in the development process and the redaction of this current dissertation. I am deeply indebted to both of my supervisors for not only guiding me in carrying out the research work but also morally supporting me in every way during the difficult times of the course.

I would like to extend my gratitude to **Prof. Vishal Verma, Dean (PG)**, **Prof. O. P. Verma (DRC Chairman)**, and **Dr. Awadhesh Kumar (Ph.D. coordinator)**. I would also like to thank **Mr. R.K. Yadav, Assistant Professor and Time Table-in-Charge**, for providing me flexibility in the workload schedules so that I could manage time to visit DRDO regularly for carrying out my research work.

This research work has been accomplished in collaboration with DTRL lab at Defense & Research Development Organization (DRDO) and hence I would like to thank people at DRDO for their invaluable guidance & support throughout the completion of this work. I would like to thank **Prof. Harish Kundra (Head of Dept., COE, Rayat University, Ropar)** and **Col. Joshil Raj (Ph.D. research fellow at DRDO)** with whom I had plentiful opportunities to discuss my concepts and ideas which resulted into this thesis. I express my deep sense of gratitude towards **Col. Rajiv Bhola (Scientist at DRDO)** who helped me a lot in the military project of designing an automated system for intelligent battlefield planning based on the new algorithms proposed in the thesis. He provided me with all the expert knowledge required to successfully accomplish this project which proved to be a milestone in the history of automated recommender system design and is currently under practical implementation by scientists at DRDO. I would also like to thank **Dr. Vartika Singh (JRF at DRDO)** and **Devender Saini (Senior Technical Assistant at DRDO)** for providing me with

the invaluable satellite data for the experimental study and also for providing technical help in MATLAB software required for coding purposes of modules of the research work.

I am deeply grateful to **Prof. Dan Simon, Dept. of Electrical and Computer Engineering, Cleveland State University** who is the founder and pioneering author of the very popular and core technique of biogeography based optimization which forms the base of this thesis, for his continuous support of my academic pursuits. He was the person whose invaluable comments and suggestions made me go an extra mile and based on his reviews and inspiration from his research works, made me develop my own generic optimization algorithm. He channelized the thought processing that brought me not only an international exposure but also an appreciation from the computational intelligence pioneers **Prof. David-Al-Dabass (Dept. of Computing and Mathematics, Nottingham Trent University, U.K.)** and **Dr. Zuwairie Ibrahim (Associate Professor, Department of Mechatronics and Robotics, Faculty of Electrical Engineering, Universiti Teknologi Malaysia, Malaysia)** in person, during the IEEE conference on Computational Intelligence, Modelling and Simulation (CIMSIm) held at Kuantan, Malaysia where I also received a merit certificate of outstanding contribution to the conference by the Organizing Chairperson himself.

I am really indebted to the people at ORACLE for supporting me in proceeding with taking up research as my career for accomplishment of the work done in this thesis. I am thankful to my **Senior Manager, Neetibodh Agrawal** and **Manager, Ravi Gupta, at ORACLE**, who were the ones who developed a deep sense of understanding of my extraordinary research interests and extended heights of cooperation and support towards me by helping me in dealing with the formalities related to resignation from the organization which was a difficult step taken during the final stages of training and company funded employ skill development program, in order to pursue research as my career. I owe my sincere thanks to a group of dynamic M. Tech students from the department of computer engineering who further strengthened my proposed concepts by showing interest in implementing them in varied applications. I would like to express my heartfelt gratitude to the other students, senior colleagues and my friends those have played an important role in the successful completion of this research endeavor.

My parents (Mr. M.K. Goel and Mrs. Sunita Goel) and my Gurudev (Param Pujya Sant Shri Asharamji Bapu) have always been inspiration and therefore, this research work is dedicated to them. It is the unconditional love, affection & continuous support from my sister (Charu) and brother (Prince), that I could complete this endeavor.

Lavika Goel

June, 2015

ABSTRACT

Nature inspired intelligence has emerged as a crucial means of implementing machine intelligence with human-like reasoning capabilities and an efficient mechanism for handling diverse uncertainty characteristics. These techniques can form the basis of building optimization algorithms which can adapt itself to suit the purpose of natural terrain understanding, and prove to be better by giving more accurate results than the other existing optimization techniques. The purpose of this research work is to develop new optimization models inspired from emerging geo-sciences techniques, extend the original optimization models of recent nature inspired techniques or build hybrid models of different nature inspired techniques by analyzing their performance governing factors, and hence present an adaptive framework for the terrain understanding problem. The research focuses on two major categorizations in the terrain understanding application: geo-spatial feature extraction, path planning and location prediction on remote sensing inputs.

The research work starts by establishing the concept of information sharing in swarm intelligence techniques which is the key factor governing the heuristic function definition and leading towards optimal solutions. It establishes entropy and similarity index as the performance governing factors influencing the classification efficiency of biogeography based classifier. Based on this analysis, two new hybrid classifiers are developed, the first is an integration of BBO with ACO2/PSO and second, is an integration of BBO- GS with ACO2/PSO. These classifiers achieve very high kappa coefficients and prove to be great advancements over the existing classifiers.

The research work moves ahead and focuses on the design of a new optimization algorithm – BBO/EE and establishment of the factors of extinction and evolution for the development of extended model of species abundance in biogeography. The above factors contribute towards the sophistication of the existing model of biogeography and thereby the concept of different HSI functions for different habitats according to the habitat characteristics is proposed. The developed extended model is then adapted for the design of a military application of enemy base station prediction and best feasible path finding targeted for two natural terrain scenarios- serving a dual purpose. The above application is a milestone in the history of automated recommender system design for battlefield planning. The research also implements feature extraction using BBO/EE with dynamic HSI function. This application when integrated with the existing hybrid bio-inspired classifier is able to achieve the highest kappa coefficient amongst all the classifiers developed till date.

The research work finally aims to design a new algorithm based on plate tectonics called as the plate tectonics based optimization. This algorithm introduces the concept of plate mobility index – weights with decision variables having tolerance and the concept of dynamic adjustment of weights. The developed algorithm will utilize an optimal set of decision variables until all the problem constraints are satisfied. The research work concludes finally by extending the taxonomy of Nature Inspired Computational Intelligence to include the techniques inspired from geo-sciences successfully.

The contents of this thesis are based on research papers accepted in refereed and mainstream international journals and conferences. The thesis is concluded with pointers for prospective future work. Additionally referred briefs are some of the prospective problems in the field that are unresolved as of now.

PUBLICATIONS AND AWARDS FROM THE THESIS

Following is the list of publications in reputed international journals and conferences. Also, mentioned are the related publications with M. Tech students, the list of related pre-Ph.D. publications, and the list of book chapters accepted in reputed publishing house.

A. International Journals:

- [1] Lavika Goel, Daya Gupta, V.K. Panchal, "Hybrid bio-inspired techniques for land cover feature extraction: A remote sensing perspective," **Applied Soft Computing**, 12(2), pp. 832-849, **Elsevier publications**, 2012. **[Impact Factor = 2.526]**
[This published work is presented in chapter 3, section 3.1]
- [2] Lavika Goel, Daya Gupta, V.K. Panchal, "Biogeography and Geo-Sciences Based Land Cover Feature Extraction: A Remote Sensing Perspective", **Applied Soft Computing**, 13, pp. 4194 – 4208, **Elsevier Publications**, 2013. **[Impact Factor = 2.526]**
[This published work is presented in chapter 3, section 3.2]
- [3] Lavika Goel, Daya Gupta, V.K. Panchal, "Land Cover Feature Extraction of Satellite Images Based on Extended Species Abundance Model of Biogeography", **Transactions on Computational Science, Springer - Verlag Berlin Heidelberg Publications**, vol. 21, pp. 90-118, 2013. **[Impact Factor = 0.75]**
[This published work is presented in chapter 5, section 5.2]
- [4] Lavika Goel, Daya Gupta, V.K. Panchal, "Performance analysis of biogeography based land cover feature extractor for building artificially intelligent hybrid classifiers", **International Journal of Applied Evolutionary Computation, IGI Global Publications**, vol. 4(4), pp. 1-26, 2013.
[This published work is presented in chapter 2, sections 2.3 and 2.4]
- [5] Lavika Goel, Daya Gupta, V.K. Panchal, "Information Sharing In Swarm Intelligence Techniques: A Perspective Application for Natural Terrain Feature Elicitation in Remote Sensing Images", **International Journal of Computer Applications (IJCA)**, Volume 32 (2), pp. 34 - 40, 2011. **[Impact Factor = 0.835]**
[This published work is presented in chapter 2, section 2.5]
- [6] Lavika Goel, Daya Gupta, V.K. Panchal, "Biogeography Based Optimization with Extinction and Evolution", **Information Sciences Journal, Elsevier publications**, 2015. (revision under process) **[Impact factor = 3.67]**
[This work is presented in chapter 4, sections 4.1, 4.2 and 4.4]

- [7] Lavika Goel, Daya Gupta, V.K. Panchal, "Applying Nature Inspired Computational Intelligence Techniques for Land Cover Feature Extraction: An Analytical Study", **Artificial Intelligence Review Journal, Springer publications**, 2015. (communicated)

[Impact Factor = 2.5]

[This work is presented in chapter 1, section 1.3 and chapter 7]

B. Book Chapters:

- [8] Lavika Goel, Daya Gupta, V.K. Panchal, " Biogeography and Plate Tectonics based Optimization for Water body Extraction in Satellite Images", International Conference on Soft Computing and Pattern Recognition (SocProS), **Springer Verlag Berlin Hiedelberg 2011, Advances in Soft Computing series (AISC)**, Volume 131, pp. 1-10, 2011. **(presented).**

[This published work is presented in chapter 3, section 3.2]

- [9] Lavika Goel, Daya Gupta, V.K. Panchal, "Dynamic model of blended biogeography based optimization for land cover feature extraction", International Conference on Contemporary Computing (IC3), **Springer - Verlag Berlin Heidelberg 2012, Communications in Computer & Information Sciences (CCIS-LNCS)**, Volume 306, pp. 8-19, 2012. **(presented).**

[This published work is presented in chapter 5, section 5.2]

- [10] Lavika Goel, "Nature Inspired Intelligence: A Modern Tool for Warfare Strategic Decision Making" in the book titled "Case Studies in Intelligent Computing: Achievements and Trends", **Taylor and Francis Publications, CRC Press**, pp. 226-244, 2014.

[This published work is presented in chapter 5, section 5.1]

- [11] Lavika Goel, V.K. Panchal, "Feature Extraction through Information Sharing in Swarm Intelligence Techniques", in the book titled "Knowledge based processes for software development", **IGI Global Publishers**, pp. 151-175, 2014.

[This published work is presented in chapter 2, section 2.5]

C. International Conferences:

- [12] Lavika Goel, Daya Gupta, V.K. Panchal, "Performance Governing Factors of BBO for Land Cover Feature Extraction: An Analytical Study", World Congress on Information and Communication Technologies (WICT), Mumbai, **IEEE Publications**, pp. 165-170, 11-14 December, 2011. **(presented).**

[This published work is presented in chapter 2, sections 2.3 and 2.4]

- [13] Lavika Goel, Daya Gupta, V.K. Panchal, "Extended Species Abundance Models of Biogeography Based Optimization", IEEE conference on Computational Intelligence,

Modelling and Simulation (CIMSIm), **IEEE Publications and CSDL**, Kuantan, **Malaysia**, pp. 7-12, 25-27 September, 2012. **(presented)**.

[This published work is presented in chapter 4, sections 4.3 and 4.4]

- [14] Lavika Goel, Daya Gupta and V.K. Panchal, “Biogeography based Anticipatory Computing Framework for Intelligent Battlefield Planning”, International Conference on Hybrid Intelligent Systems (HIS), **IEEE Publications**, Pune, pp. 41-46, 4-7 December, 2012. **(presented)**.

[This published work is presented in chapter 5, section 5.1]

D. Publications with M.Tech Students (During Ph.D):

- [15] Daya Gupta, Lavika Goel, Abhishek, “An Efficient Biogeography Based Face Recognition Algorithm”, International Conference on Advances in Computer Science and Engineering (CSE), Los Angeles, U.S.A., **Atlantis Press**, pp. 64-67, 1-2 July, 2013.
- [16] Daya Gupta, Lavika Goel, Vipin Aggarwal, “A Hybrid Biogeography Based Heuristic for the Mirrored Traveling Tournament Problem”, Sixth International Conference on Contemporary Computing (IC3), Noida, **IEEE Publications**, Catalog no. CFP 1381U-CDR, pp. 325-330, 8-10 August, 2013. **(presented)**
- [17] Tanu Varshney, Lavika Goel, Daya Gupta, V.K. Panchal, “Efficient mixed pixel resolution using bio-inspired heuristics”, **IEEE Publications**, International Conference on Optimization, Reliability, and Information Technology (ICROIT), Faridabad, Haryana, pp. 364-368, 6-8 February, 2014. **(presented)**.
- [18] Daya Gupta, Lavika Goel, Kanishka Bansal, “A Novel Approach for Face Recognition Based on Extended Species Abundance Model of Biogeography”, In: Proceedings of Second International Conference on Emerging research in Computing, Information, Communication and Applications (ERCICA), **Elsevier Publications**, Bangalore, 1-2 August, vol. 2, pp. 893-899, 2014. **(presented)**
- [19] Daya Gupta, Lavika Goel, Ashish Chopra, “Enhanced Heuristic Approach for Travelling Tournament Problem based on Extended Species Abundance Models of Biogeography”, In: Proceedings of International Conference on Advances in Computing, Communications and Informatics (ICACCI), Delhi, **IEEE Publications**, Delhi, 25-27 September, pp. 1118-1124, 2014. **(presented)**

E. Pre-Ph.D. Publications:

- [20] Lavika Goel, Daya Gupta and V.K. Panchal, “Embedding Expert Knowledge to Hybrid Bio-Inspired Techniques: An Adaptive Strategy towards Focused Land Cover Feature Extraction”, **International Journal of Computer Science & Information Security (IJCSIS)**, 1(2), pp 244-253, 2010.

- [21] Lavika Goel, “Land cover Feature Extraction using Hybrid Swarm Intelligence Techniques-A Remote sensing perspective”, **ACEEE International Journal on Signal & Image Processing**, 1(3), Dec 2010.
- [22] Lavika Goel, V.K. Panchal, Daya Gupta, Rajiv Bhola, “Hybrid ACO-BBO Approach for predicting the Deployment Strategies of enemy troops in a military Terrain Application”, in **4th International Multi Conference on Intelligent Systems & Nanotechnology (IISN)**, 24-26 February, 2010.

2. Awards and recognitions from the research

Following are some awards and recognitions that were received from the research work done in the thesis. The author of the thesis was also invited for delivering lectures and seminars in various institutes on the research accomplished during the course of the thesis particularly on the military application work and the framing of the new theory of deriving optimization framework from the theory of plate tectonics both of which are concepts which have proved to be of historic and critical importance in the real world.

A. List of achievements

- [1] **2015: Quality Assurance Head & Technical Committee member to Computational Intelligence and Research Group (CiRG).**
- [2] **2014: The author was awarded the role of a regular reviewer to the Elsevier Journal of Applied Soft Computing in the year 2014** and reviewed about 5-10 manuscripts since then.
- [3] **2013: The idea of extending the taxonomy of nature inspired computational intelligence techniques to include the geo-sciences based techniques was highly appreciated and thereafter the author received an offer of sponsorship from MIR labs for publication and presentation of papers listed in Scopus.**
- [4] **2012: After the presentation on extended models of biogeography, the author was offered to chair two sessions on “Neural Networks” and “Evolutionary Computation” during IEEE conference on Computational Intelligence, Modelling and Simulation (CIMSIm) held at Kuantan, Malaysia during 25 - 27 September, 2012 and also received a merit certificate of outstanding contribution to the conference by the Organizing Chairperson, Prof. David Al Dabass, Nottingham Trent University, U.K.**
- [5] **2012: The author was offered the role of a regular reviewer to the Elsevier Journal of Information Sciences in the year 2012** and has reviewed about 10 manuscripts since then.
- [6] **2012: Based on the significance and findings of the research that the author has accomplished in the thesis on nature inspired intelligence, the author was offered**

membership by Machine Intelligence & Research (MIR) Labs, Greater Seattle Area, Washington, U.S. and is currently a member of MIR labs.

- [7] **2011:** The author worked in close correspondence with **Dan Simon**, the founder of the very popular BBO technique, to formulate the BBO/EE technique, who showed interest in associating with this project and gave his valuable inputs at every step. The author received appreciation by Simon on the formulation of this technique and **he served as a reviewer to the manuscript and recognized the work by listing on his website** dedicated to new research on this technique.
- [8] **2011:** The author has worked extensively at DTRL Lab at DRDO, New Delhi for the experimental work and accomplishment of the objectives of the thesis. The author thus holds a research experience of about **3.5 years working at Defense & Terrain Research Lab (DTRL), Defense & Research Development Organization (January 2011 – June 2014).**

B. Lectures Delivered and Invited Talks

- [1] **2015: Executive Committee Member, Head Coordinator and Keynote speaker** at the **CiRG, IEEE and CSI sponsored workshop** held at **SBIT, DCRUST, Sonipat** during 11-12th February 2015.
- [2] **2014:** Invited for a guest lecture on “Evolving new Computational Intelligence Paradigms” during the National Workshop on Advance Computational Intelligence Paradigms and Future Directions organized by Dept. of CSE, **SRM University, NCR Campus** held on 17th May, 2014.
- [3] **2013:** Invited for **keynote address** on “Nature Inspired Optimization Algorithms” during the 2nd International Conference on Recent Trends in Computing (ICRTC) organized by **Dept. of CSE, SRM University, NCR Campus** during **4th -5th October, 2013.**
- [4] **2012: Chaired two sessions** in different tracks during IEEE conference on Computational Intelligence, Modelling and Simulation (CIMSIm), held at Kuantan, **Malaysia during 25-27 September, 2012** and **received a certificate of outstanding contribution to the conference by the Organizing Chairperson.**
- [5] **2012: Invited for delivering lecture** on Nature Inspired Algorithms as a **guest speaker** by **Delhi Technological University (DTU)** in the technical paper presentation contest held during 25-29 February, 2012.
- [6] **2012:** Invited for delivering lecture on “Plate Tectonics Based Optimization” by **Department of Information Technology, SGTBIT, GGSIPU, New Delhi** during seminar on “Recent Trends in Nature Inspired Computing” on 24 February, 2012.

- [7] **2012: Revolutionized ideas of a practically implementable recommender system design for critical military applications** through nature inspired optimization by delivering invited lecture by Defense & Terrain Research Lab, **Defense & Research Development Organization (DRDO)** during workshop on “Nature Inspired Computational Models for Terrain Understanding” held during 29th November – 2nd December, 2012.
- [8] **2012: Received certificate of appreciation** on delivering invited lecture on “Nature Inspired Optimization Algorithms for Geo-Spatial Feature Extraction” by **Department of Information Technology, Institute of Innovation in Technology & Management (IITM), New Delhi**, during the National Seminar On "Emerging Trends in Soft Computing” held on 28th April, 2012.
- [9] **2012: Invited for training research scholars and faculty** on the very new research area of “Computational Intelligence for developing Nature Inspired Optimization Algorithms for terrain understanding applications” by **Indian Agricultural Research Institute (IARI), New Delhi**, during the National Seminar On "Emerging Trends in Soft Computing” held on 28th December, 2012.
- [10] **2011: Delivered invited talk in the Annual Convention of Indian Society of Remote Sensing ISRS-2011** and National Symposium on Empowering Rural India through Space Technology held at M.P. Council of Science & Technology, Bhopal during November 9-11 2011.
- [11] **2011: Invited for delivering lecture by Geetaratan International Business School (GIBS), New Delhi, 2011** during seminar on “Intelligent Technologies for Terrain Understanding” on 25th April 2011.

LIST OF ACRONYMS

ACO	Ant Colony Optimization
AIS	Artificial Immune System
ANN	Artificial Neural Network theory
AS	Ant System
ASF	Artificial Swarm fish
BBO	Biogeography Based Optimization
BBO/GS	Biogeography and Geo-Sciences based Optimization
BBO/EE	Biogeography Based Optimization with Extinction / Evolution
BBO/RE	Biogeography Based Optimization with Extinction / Evolution
BBO/ES	BBO / Evolutionary Strategy
BCO	Bee Colony Optimization
BFO/FPAB	Bacterial Foraging Optimization/Flower Pollination by artificial bees
CI	Computational Intelligence
CS	Cuckoo Search
CSS	Charged System Search
DE	Differential Evolution
DEM	Digital Elevation Model
DN	Digital number
DNA Computing	Deoxyribo Nucleic Acid Based Computing
DPSO	Discrete Particle Swarm Optimization
EA	Evolutionary Algorithm
ECG	Electro Cardio Gram
ERT	Expected running time
ES	Evolutionary Strategy
FCM	Fuzzy c-means
FI	Fuzzy Inference Mechanism
FL	Fuzzy Logic
GA	Genetic Algorithm
GIS	Geographic Information Systems
GS	Gravitational Search
HSI	Habitat Suitability Index (Similarity Threshold)

IEPSO	Immunity Enhanced Particle Swarm Optimization
IWD	Intelligent Water Drop
K-coefficient	Kappa coefficient
LBBO	Linearized BBO
LISS	Linear Imaging Self- Scanning Memory
LU/LC	Landuse/Landcover
MC	Membrane Computing
MDMC	Minimum Distance to Mean Classifier
MIR	Middle Infra-Red
MLC	Maximum Likelihood Classifier
NIR	Near Infra-Red
PBIL	Probability based Incremental Learning
PR	Probabilistic Reasoning
PBO	Plate Tectonics Based Optimization
PMI	Plate Mobility Index
PSO	Particle Swarm Optimization
RCBBO	Real Coded Biogeography Based Optimization
RCGA	Real Coded Genetic Algorithm
RFD	River Formation Dynamics
ROSETTA	Rough set toolkit Theory Toolkit
RS	Rough set theory
RS1	Radarsat-1
RS2	Radarsat-2
SAR	Synthetic Aperture Radar
SC	Soft Computing
SI	Swarm Intelligence
SIV	Suitability Index Variables
SGA	Stud Genetic Algorithm
SOFM	Self Organizing Feature Maps
SDS	Stochastic Diffusion Search
TSP	Travelling Salesman Problem

CONTENTS

Acknowledgement	III
Abstract	V
Publications and Awards from the thesis	VII
List of Acronyms	XIII
Contents	XV
List of Figures	XIX
List of Tables	XXII
Chapter 1: Introduction	1
1.1. Research Objective	2
1.2. Components of Terrain Understanding	3
1.2.1. Geo-spatial feature extraction	4
1.2.2. Path planning and location prediction	6
1.3. Computational Intelligence: State of the art	8
1.3.1. Human Cognition Based Techniques	8
1.3.2. Nature Inspired Computational Intelligence Techniques	10
1.4. Problem Statement	16
1.5. Thesis Approach	19
1.5.1. Performance Analysis Models	19
1.5.2. Hybrid Models	20
1.5.3. Extended Models	21
1.5.4. Adaptive Models	22
1.5.5. New Generic Models	23
1.6. Contribution of the thesis	23
1.7. Outline of the Thesis	25
1.8. Observations	27
Chapter 2: Performance Analysis Model of Biogeography Based Feature Extractor	28
2.1. Biogeography based Optimization	30
2.1.1. BBO concept	30
2.1.2. BBO Definitions and Algorithm	31
2.2. Biogeography Based Framework for Land Cover Feature Extraction	34
2.2.1. Defining Parameters for the Biogeography Based Feature Extractor	34

2.2.2. Algorithm for biogeography based land cover feature extraction	38
2.3. Performance Analysis of Biogeography Based Land Cover Feature Extractor	40
2.3.1 Proposed Framework	40
2.3.2 Performance governing factors of BBO for land cover feature extraction	41
2.3.3 Detailed Framework for studying BBO behavior on natural terrain features	44
2.4. Implementation, Results and Analysis	46
2.4.1. Case study: 7-band satellite image of Alwar Region in Rajasthan	47
2.4.2. Case study: 4-band satellite image of Patalganga region in Shivaliks	54
2.5. Information Sharing In Swarm Intelligence Techniques	59
2.5.1. Information sharing in BBO	60
2.5.2. Information sharing in ACO	61
2.5.3. Information sharing in BCO	62
2.5.4. Information sharing in PSO	64
2.5.5. Discussion	65
2.6. Observations	65
Chapter 3: Hybrid Models of Nature Inspired Computational Intelligence For Geo-Spatial Feature Extraction	67
3.1. Hybrid Bio-Inspired Techniques for Land Cover Feature Extraction	69
3.1.1. A Brief Review of hybrid ACO2/PSO Technique	69
3.1.2. Land Cover Feature Extraction using ACO2/PSO technique	70
3.1.3. Proposed Framework for the hybrid land cover feature extractor	71
3.1.4. Results and Discussion	77
3.1.5. Accuracy assessment of the proposed algorithm	79
3.1.6. Classification comparison of Hybrid ACO2/PSO/BBO Classifier with the traditional probabilistic classifiers	81
3.1.7. Classification results of other computational intelligence based techniques used for satellite image classification	81
3.2. Biogeography and Geo-Sciences Based Geo-spatial Feature Extraction	84
3.2.1 Basics of Geo-Sciences	85
3.2.2. Proposed framework for the BBO-GS Based Feature Extractor	86
3.2.3. Results and Analysis	95
3.2.4. Comparison of BBO-GS based classifier with the other recent geo-spatial feature extraction techniques	107
3.3. Observations	110

Chapter 4: Extended Optimization Models of Nature Inspired Computational Intelligence	112
4.1. Biogeography with Extinction and Evolution	114
4.1.1. Factors of evolution and extinction	114
4.1.2. Biogeography with Extinction / Evolution	116
4.1.3. Species abundance model	118
4.2. Biogeography based Optimization with Extinction and Evolution	122
4.2.1. BBO/EE Definitions	123
4.2.2. BBO/EE Algorithm	126
4.3. Extended Models of Species Abundance in Biogeography	127
4.3.1. Linear Models	127
4.3.2. Non-Linear Models	129
4.4. Performance Analysis	130
4.4.1. Performance analysis of BBO/EE	130
4.4.2. Performance analysis of the extended models of species abundance	134
4.5. Observations	135
 Chapter 5: Adaptive Models of Nature Inspired Computational Intelligence	 137
5.1. Two-phase recommender system design based on BBO/EE	138
5.1.1. Proposed Problem Solving Framework Based on BBO/EE	139
5.1.2. Two-Phase Recommender System Architecture	141
5.1.3. Two-Phase Recommender System on varied natural terrain scenarios	160
5.1.4. Performance comparison results of the two-phase recommender system	176
5.1.5. Applications	178
5.2. BBO/EE for Land Cover Feature Extraction	179
5.2.1. Proposed Architecture	179
5.2.2. Results and Discussion	184
5.3. Observations	198
 Chapter 6: New Generic Models of Nature Inspired Computational Intelligence	 200
6.1. Plate Tectonics	202
6.1.1. Forces acting on a plate	203
6.1.2. Governing Forces on a Plate	203
6.1.3. Equations governing the forces	204
6.1.4. Plate Motion Model	208

6.2. Plate Tectonics Based Optimization	211
6.2.1. PBO Definitions	212
6.2.2. PBO Algorithm	214
6.2.3. Mathematical formulation of PBO	216
6.2.4. Differences between PBO and the other EAs	218
6.3. Performance Analysis and Results	219
6.3.1. Performance analysis on benchmark functions	219
6.3.2. Plate Tectonics Based Optimization for Travelling Salesman Problem	225
6.4. Observations	230
 Chapter 7: Conclusion and future directions	233
7.1. Conclusion	234
7.2. Research Outcome and Future Scope	236
 References	240
Biography of Author	255

LIST OF FIGURES

Figure 1.1	Overview of research work in the thesis	27
Figure 2.1	Illustration of two candidate solutions	31
Figure 2.2	Eco-system – Initial state of Habitats	35
Figure 2.3	Biogeography based land cover feature extractor	39
Figure 2.4	Proposed framework for validation of performance equation	41
Figure 2.5	Detailed framework for relationship between efficiency & entropy	45
Figure 2.6	Detailed framework for relationship b/w efficiency & similarity index	46
Figure 2.7	7-band image of Alwar	47
Figure 2.8	Biogeography Based Classification of Alwar image	47
Figure 2.9	Histogram plot in Radarsat-2 band	50
Figure 2.10	Histogram plot in Radarsat-1 band	50
Figure 2.11	Graphical representation of efficiency v/s entropy in both bands	51
Figure 2.12	Classification efficiency v/s Normalized Similarity Index	53
Figure 2.13	Graph plot of efficiency, Similarity Index and Entropy for Alwar	53
Figure 2.14	4 band image of Patalganga	54
Figure 2.15	Biogeography Based Classification of patalganga image	54
Figure 2.16	Histogram plot in NIR band	56
Figure 2.17	Histogram plot in MIR band	57
Figure 2.18	Graphical representation of the efficiency v/s the entropy in both bands	57
Figure 2.19	BBO Classification efficiency v/s Normalized Similarity Index	58
Figure 2.20	Graph plot of efficiency, Similarity Index and Entropy for Patalganga	59
Figure 2.21	Information sharing in swarm-based approaches for feature extraction	63
Figure 3.1	Hybrid ACO2/PSO algorithm	70
Figure 3.2	Modified algorithm for hybrid ACO2/PSO based feature extraction	71
Figure 3.3	Software architecture of the proposed intelligent classifier	73
Figure 3.4	Graph plot of the DMMs of each feature v/s 7-bands in Alwar image	74
Figure 3.5	Mathematical formulation of the ACO2/PSO/BBO Algorithm	76
Figure 3.6	Water body extraction from the Alwar image with each iteration	78
Figure 3.7	Classified image after applying BBO	79
Figure 3.8	Hybrid ACO2/PSO/BBO classified image	79
Figure 3.9	A comparison with the Traditional Probabilistic Classifiers	81
Figure 3.10	Classified images of Alwar region after applying various CI techniques	83
Figure 3.11	Adapting the geo-sciences phenomenon for feature extraction	85

Figure 3.12	Attempt at a new AI technique	86
Figure 3.13	Summary of the steps followed for developing the hypotheses	88
Figure 3.14	Surface topography demarcates different land use types in Alwar image	88
Figure 3.15	DEM & Hill Shading of Alwar	90
Figure 3.16	Biogeography and Geo-sciences based land Cover Feature Extraction	92
Figure 3.17	Mathematical formulation of the BBO-GS based feature extraction	94
Figure 3.18	Feature Entropy v/s Plate Entropy in Classes 1-XX for Alwar	99
Figure 3.19	Final classified image of Alwar with BBO-GS	100
Figure 3.20	Feature Entropy v/s Input Plate Entropy in DEM band for Iterations II-V	102
Figure 3.21	Water body extraction from the Alwar image with each iteration	102
Figure 3.22	Feature Entropy v/s Plate Entropy in Classes 1-XV for Patalganga	106
Figure 3.23	Final classified image of Patalganga with BBO-GS	107
Figure 3.24	Comparison of feature extraction results on the dataset of Alwar region	109
Figure 4.1	Graph representing the species evolution rate curve on a habitat	115
Figure 4.2	Graph representing the species extinction rate curve on a habitat	117
Figure 4.3	Species abundance model (a) Simon's model. (b) Proposed model.	120
Figure 4.4	Steady state probability (a) Simon's model. (b) Proposed model.	121
Figure 4.5	Algorithm for BBO with Extinction / Evolution	128
Figure 4.6	Graph plot of the minimum cost values for the extended migration models	135
Figure 5.1	Extended BBO/EE based on HSI function modeling.	142
Figure 5.2	Two-phase recommender system architecture based on extended BBO/EE	149
Figure 5.3	Functional architecture of the proposed two-phase recommender system	151
Figure 5.4	Pseudo-code for the obstacle detector	157
Figure 5.5	Morphological operations	157
Figure 5.6	Pseudo-code for effort factor calculation	158
Figure 5.7	An Integrated Framework for the Two-Phase Recommender System	159
Figure 5.8	Mathematical Framework for the two-phase recommender system	160
Figure 5.9	7-band FCC satellite image of size 472 X 546 of Alwar area in Rajasthan	161
Figure 5.10	7-band Alwar image with the candidate base stations marked in red.	161
Figure 5.11	Candidate base stations categorized into respective classes.	164
Figure 5.12	Portion of the Alwar image with the source and the target marked in red.	165
Figure 5.13	Paths extracted after applying PSO	166
Figure 5.14	Refined paths after morphological operations	166
Figure 5.15	Alwar image with the safest and the shortest path	167
Figure 5.16	Google earth Mussourie image of size 472 X 576 in Himachal Pradesh	168

Figure 5.17	Google earth Mussourie image with the suited stations marked in red	168
Figure 5.18	Mussourie image with the stations categorized into respective classes	172
Figure 5.19	Portion of the Mussourie image with the source and target marked in red.	173
Figure 5.20	Paths extracted after applying PSO	174
Figure 5.21	Refined paths after morphological operations	174
Figure 5.22	Mussourie image with the safest and the shortest path	175
Figure 5.23	Algorithm for feature extraction based on the extended BBO/EE	181
Figure 5.24	Software architecture of the proposed land cover feature extractor	182
Figure 5.25	Bar graph representing the % Difference Matrix for both HSI functions	187
Figure 5.26	Final classified image of Alwar with HSI function based on DMM	189
Figure 5.27	Final classified image of Alwar with HSI function based on 'entropy'.	189
Figure 5.28	Final classified image of Alwar with extended BBO/EE	189
Figure 5.29	Bar graph representing the % difference matrix for both HSI functions	192
Figure 5.30	Final classified image of Patalganga with HSI function based on DMM	194
Figure 5.31	Final classified image of Patalganga with HSI function based on 'entropy'	194
Figure 5.32	Final classified image of Patalganga with extended BBO/EE	194
Figure 5.33	Classified images of Alwar region with various computing techniques	197
Figure 6.1	Graph plots of the behavior of the contributing entities in plate movement v/s temperature (T) (a) Viscosity (b) Frictional Drag (c) Ridge-push (d) Slab-pull (e) Net Force (due to Ridge-push) and (f)-Net Force (due to Slab-pull)	210
Figure 6.2	Combined graph plot of the net force acting on the plate v/s temperature for the slab-pull and the ridge- push condition	211
Figure 6.3	Plate-tectonics Based Optimization (PBO)	215
Figure 6.4	PBO Approach	217
Figure 6.5	Graph plot of the minimal cost estimate on 14 benchmark functions.	224
Figure 6.6	TSP problem solving by PBO.	226
Figure 6.7	Ideal solution for TSP problem solving through PBO on 5-city dataset	227
Figure 6.8	Solution graph for TSP problem solving by PBO for 5 iterations.	228

LIST OF TABLES

Table 2.1	Error matrix of BBO for Alwar region	48
Table 2.2	Producer's accuracy for Alwar	48
Table 2.3	User's accuracy for Alwar	48
Table 2.4	Combined accuracy of BBO for each land cover feature for Alwar	48
Table 2.5	Entropy in RS-1 and RS-2 bands	49
Table 2.6	Entropy calculation in Radarsat-2 band	49
Table 2.7	Entropy calculation in Radarsat-1 band	50
Table 2.8	BBO Classification efficiency v/s Entropy in NIR and MIR bands	51
Table 2.9	HSI values of the training set for each land cover features in Alwar image	51
Table 2.10	Average of differences D_u for each feature for the Alwar image	52
Table 2.11	Calculation of Similarity index 'S' from D_u for each feature	52
Table 2.12	Combined representation of η_p , S and S	53
Table 2.13	Combined table of η_p , S and E for each land cover feature	53
Table 2.14	Error Matrix for the Satellite image of Patalganga region when BBO	54
Table 2.15	Producer's accuracy for Patalganga	54
Table 2.16	User's accuracy for Patalganga	55
Table 2.17	Combined accuracy of BBO for each land cover feature for Patalganga	55
Table 2.18	Feature Entropy in NIR and MIR bands	55
Table 2.19	Entropy calculation in NIR band	56
Table 2.20	Entropy calculation in MIR band	56
Table 2.21	BBO classification efficiency v/s the entropy in NIR and MIR bands	56
Table 2.22	Table of HSI values of the training set for each feature in Patalganga image	57
Table 2.23	Calculation of Similarity index 'S' from D_u for each feature	58
Table 2.24	Average of differences D_u for each feature for the Patalganga image	58
Table 2.25	Combined representation of η_p , S and S_N	58
Table 2.26:	Combined table of η_p , S and E for each land cover feature	59
Table 3.1	Error matrix when only BBO is applied	80
Table 3.2	Error matrix when hybrid ACO2/BBO technique is applied.	80
Table 3.3	Error matrix when hybrid ACO2/PSO/BBO technique is applied.	80
Table 3.4	Producer's Accuracy	80
Table 3.5	User's Accuracy	81
Table 3.6	A comparison of the Kappa coefficients of traditional classifiers	81
Table 3.7	Kappa coefficient ($\hat{k}$) of recent CI based classifiers v/s probabilistic	83

Table 3.8	HSI calculations and the corresponding classified feature for the first iteration	98
Table 3.9	DEM entropies obtained in each iteration through texture analysis	101
Table 3.10	Categorization of plates into similarity classes	101
Table 3.11	Feature categorization step in order to give a class tag to the plates	101
Table 3.12	HSI calculations and the corresponding classified feature for the first iteration	101
Table 3.13	HSI calculations and feature classification for Iterations II-V	103
Table 3.14	Error matrix of biogeography and geo-sciences based extractor on Alwar	104
Table 3.15	HSI calculations and the corresponding classified feature for the first iteration	105
Table 3.16	Error matrix of BBO-GS based extractor on Patalganga	107
Table 3.17	Comparison of kappa coefficient of recent CI based classifiers	109
Table 3.18	Error matrix of hybrid ACO2/PSO/(BBO-GS) on Alwar region.	109
Table 3.19	Producer's accuracy	109
Table 3.20	User's accuracy	109
Table 4.1	Mean normalized results and CPU times on benchmark functions.	133
Table 4.2	Best normalized optimization results on benchmark functions.	133
Table 4.3	Mean optimization results and CPU times on benchmark function 'Step'.	134
Table 4.4	Mean optimization results and CPU times on function 'Griewank'.	135
Table 5.1	Scales for Mobility Factor	144
Table 5.2	Scales for Aspect ratio	145
Table 5.3	Scales for area covered under slope	145
Table 5.4	Scales for areas of strategic importance	146
Table 5.5	Scales for ease of Terrain activities	146
Table 5.6	SIV's and their assigned weights.	147
Table 5.7	Fixing the SIVs and their values on a plain / desert region	161
Table 5.8	HSI, effort and selectivity factor calculations for each candidate stations	163
Table 5.9	Categorization of base stations based on normalized selectivity factor values	164
Table 5.10	Table of categorized output	164
Table 5.11	Local and Global feature habitat update	167
Table 5.12	HSI Computation and path planning by ACO	168
Table 5.13	SIV and their values for mountainous region	169
Table 5.14	HSI, effort and selectivity factor calculation for each candidate base stations	171
Table 5.15	Categorization of base stations based on Normalized selectivity factor values	171
Table 5.16	Table of categorized output	171
Table 5.17	Local and Global feature habitat Update	174
Table 5.18	HSI Computation and Path planning by ACO.	175

Table 5.19	HSI calculations using traditional decision making	176
Table 5.20	Prediction efficiency of traditional decision making v/s the proposed model	177
Table 5.21	Results of path planning by various optimization techniques.	178
Table 5.22	Resultant percentage difference matrix for the Alwar image.	186
Table 5.23	Classified feature index matrix for the Alwar image	188
Table 5.24	HSI functions chosen dynamically for each class in the Alwar image.	188
Table 5.25	Error matrix of BBO with the HSI function based on DMM	189
Table 5.26	Error matrix of BBO with the HSI function based on ‘entropy’	189
Table 5.27	Error matrix of the extended BBO/EE based feature extractor on Alwar	190
Table 5.28	Producer’s Accuracy	190
Table 5.29	User’s Accuracy	190
Table 5.30	Resultant percentage difference matrix for the Patalganga image.	191
Table 5.31	Classified feature index matrix for the Patalganga image.	193
Table 5.32	HSI functions chosen dynamically for each class in the Patalganga image.	193
Table 5.33	Error matrix of BBO with the HSI function based on DMM	195
Table 5.34	Error matrix of BBO with the HSI function based on ‘entropy’	195
Table 5.35	Error matrix of extended BBO/EE based feature extractor on Patalganga	195
Table 5.36	Producer’s Accuracy	195
Table 5.37	User’s Accuracy	195
Table 5.38	Error matrix of Hybrid ACO2/PSO/(BBO/EE) on Alwar region.	196
Table 5.39	Producer’s Accuracy	196
Table 5.40	User’s Accuracy	196
Table 5.41	Kappa coefficients of various recent CI based classifiers	197
Table 6.1	Table representing the values of viscosity, frictional drag, slab pull force and the net force values with the increasing temperature when the slab pull force acts between the temperature ranges from 0 – 500 K.	211
Table 6.2	Table representing the values of viscosity, frictional drag, ridge push force and the net force values with the increasing temperature when the ridge push force acts between the temperature range from 750 – 5000 K.	211
Table 6.3	Summary of differences between PBO and the other EAs	220
Table 6.4	Mean normalized optimization results and CPU times on 14 benchmarks	224
Table 6.5	Best normalized optimization results on 14 benchmark functions.	225
Table 6.6	Summarizing the values of PMI and the best path by PBO in each iteration.	229
Table 6.7	Comparison of PBO with other algorithms on instances of the TSP problem	230

Chapter 1: Introduction

Chapter 1

- 1.1. Research Objective
- 1.2. Components of Terrain Understanding
- 1.3. Computational Intelligence: State of the art
- 1.4. Problem Statement
- 1.5. Thesis Approach
- 1.6. Contribution of the thesis
- 1.7. Thesis Organization
- 1.8. Observations

Nature-Inspired algorithms have been gaining much popularity in recent years due to the fact that many real-world optimization problems have become increasingly large, complex and dynamic. The size and complexity of the problems nowadays require the development of methods and solutions whose efficiency is measured by their ability to find acceptable results within a reasonable amount of time, rather than an ability to guarantee the optimal solution. Thus, unlike the exhaustive approaches to problem solving which may lead to a combinatorial explosion, the nature inspired techniques have emerged as efficient heuristics which are capable of leading to near-optimal solutions within the given time and space constraints. These techniques have emerged as a crucial means of implementing machine intelligence with human-like reasoning capabilities and an efficient mechanism for handling diverse uncertainty characteristics [(Johnson, C.G., 2009), (Kari, L., et. al., 2008)]. Researchers have shown keen interest on the applications of nature inspired computing in divergent domains. Scanty references are available on their applications in the area of terrain understanding in remote sensing. All such problems, which have been looked for their solution in classical artificial intelligence platforms like Probabilistic and Bayesian, have been addressed and presented in this research work with new thoughts processing, based on nature inspired techniques of computational intelligence. Various well founded matrices in the remote sensing community for assessing the accuracy of the terrain understanding problems such as the kappa coefficient, etc., are used for comparative study of the results [(Congalton, R.G., 1983), (Cohen, J., 1960), (Rosenfield, G., et. al., 1986)].

Closely working in collaboration with Defense & Terrain Research Laboratory at DRDO, the research is based on two very popular categorizations in the terrain understanding application that of geo-spatial feature extraction and path planning (which includes location prediction) mostly on remote sensing datasets [(Long, W. III, et. al., 2004), (Lillesand, T. M., et. al., 2007)] (provided courtesy of DRDO). The applications targeted are military, commercial and scientific applications of high impact and critical importance in real world.

1.1. Research Objective

Computational intelligence techniques provide good solutions in diverse application domains ranging from land-use / land-cover classification, battlefield assessment, obstacle free path planning, object identification, pattern classification, estimation of groundwater availability, etc. Human cognition based computational intelligence techniques were initially used however, the focus shifted to nature inspired computing because of their ability to deal with the inherent uncertainty in geo-spatial data and granular structure of the remote sensing data from the earth observing system.

Very recently, the optimization technique inspired from the concepts of biogeography [(Simon, D., 2008), (Simon, D., et. al., 2013), (Dawei, D., et. al., 2009), (Ma, H., 2010), (Goel, S., et. al.,

2011)] called as the biogeography based optimization technique proposed by Dan Simon has been used in the domain of terrain understanding because of its improved optimization capabilities and flexibility for applications such as in feature extraction, mixed pixel resolution, travelling salesman problem, etc. [(Song, Y., et. al., 2010), (Gupta, N., et. al., 2011B), (Varshney, T., et. al., 2014)]. Original or extended forms of this technique when integrated with the existing nature inspired optimization techniques can significantly improve their optimization capability hence leading to better terrain classification and understanding [(Panchal, V. K, et. al., 2009), (Gupta, S., et. al., 2011),(Gupta, D., et. al., 2014)]. Thus, the first objective of the research work is to study and analyze the very recent technique of biogeography based optimization to develop hybrid optimization algorithms by blending with the other existing nature inspired algorithms and to develop extended forms of the original biogeography based algorithm by modifying the basic species abundance model of this technique. The developed algorithms can be used for the given terrain understanding application.

Also, very recently geo-sciences based natural phenomenon such as the theories about the evolution of the earth [Bonabeau, E., et. al., 1999], energy transformations in the earth's crust leading to earthquakes, plate tectonics, big-bang crunching, etc. and energy transformations on sun and moon in the form of oceanic currents and tidal waves, etc. have also drawn attention since these natural phenomenon can be very well applied for the development of optimization algorithms which can adapt itself to suit the purpose of natural terrain understanding, and prove to be better by giving more accurate results than the other existing optimization techniques for certain specific applications. Also, these phenomena can be exploited to develop optimization algorithms for the purpose of natural terrain understanding [Momm, H.G., et. al., 2009]. Hence, the second objective of the research work is to develop new generic nature-inspired techniques from the emerging geo-sciences concepts for tackling various kinds of optimization problems and then applying them to terrain understanding applications. Thus, the focus of the thesis will be the development of adaptive optimization models from the terrain perspective.

In this chapter, a brief survey has been done on the terrain understanding applications of geo-spatial feature extraction, path planning (and location prediction), and their importance for inclusion as the prime case study in the thesis along with the respective metrics used for each. In addition, various computational intelligence techniques currently being used have been discussed briefly. Thereafter, the problems addressed by the thesis are derived and the proposed solution is outlined. Finally the contribution made in the thesis is discussed.

1.2. Components of Terrain Understanding

This section presents briefly about the terrain understanding applications that have been taken up in the thesis along with the respective accuracy assessment metrics used for each

application. The focus of this research is on two major categorizations in the terrain understanding application as outlined before, that of geo-spatial feature extraction and second that of path planning (and location prediction). In path planning, the travelling salesman problem and the shortest and safest obstacle free path planning will be undertaken. Location prediction will incorporate the military critical application of enemy base station prediction both on remote sensing inputs.

Land use / Land cover and equally important the path planning and location prediction applications dealt with in this thesis have been taken up for this research study because they are critical factors for the societal missions and the applications dealt with are very important real world applications.

1.2.1. Geo-spatial feature extraction

In remote sensing, the geospatial features means all the objects on the planet Earth with their location referenced onto Earth. The most significant characteristic of the image data in a remote sensing system is the wavelength, or range of wavelengths, used in the image acquisition process. The energy emitted by the earth itself (dominant in the so-called *thermal* infrared wavelength range) can also be resolved into different wavelengths that help in understanding the properties of the earth surface region being imaged. The purpose of acquiring remote sensing image data is to be able to identify and assess, by some means, surface materials and their spatial properties which can be possible if the sensor gathers data at several wavelengths. From this, it may be inferred that every object has its own characteristic curve in different wavelengths. And these is known as their *spectral signatures* [(Hand, J., 1997), (Long, W. III, et. al., 2004)].

The accuracy of spatial data has been defined by the United States Geological Survey (USGS) as: "*The closeness of results of observations, computations, or estimates to the true values or the values accepted as being true*" [USGS, 1990] and can be subjected to different quantitative accuracy assessment methods [Lunetta, R., et. al., 2004]:

Thematic mapping via an image classification is one of the most commonly undertaken analyses of remotely sensed data. Thematic map accuracy is most commonly estimated through the use of site-specific techniques based on the analysis of the entries in a confusion or error matrix [(Wulder, M. A., et. al., 2006), (Anderson, J. R. et. al., 1976), (Rogan, J., et. al., 2003), (Congalton, R. G., 1983), (Foody, G. M., 2002A), (Foody, G. M., et. al., 2002B), (Foody, G. M., 2006)]. This matrix should provide a simple summary of classification accuracy and highlight the two types of thematic error that may occur: omission and commission. Thus, the estimates of accuracy derived from confusion matrices constructed from testing sets drawn by simple random and stratified random sampling, without any allowance for the difference in the

sample design, may differ substantially if the classes vary in abundance and spectral separability. There have been, for example, many calls for the remote sensing community to adopt measures such as the kappa coefficient of agreement in the assessment of classification accuracy [(Congalton, R. G., 1983), (Lunetta, R., et. al., 2004)].

A **confusion matrix** lists the values for known cover types of the reference data in the columns and for the classified data in the rows [Congalton, R. G., 1983]. The main diagonal of the matrix lists the correctly classified pixels. One basic accuracy measure is the overall accuracy, which is calculated by dividing the correctly classified pixels (sum of the values in the main diagonal) by the total number of pixels checked. Besides the overall accuracy, classification accuracy of individual classes can be calculated in a similar manner. Two approaches are possible:

1. *User's accuracy*, and
2. *Producer's accuracy*.

The producer's accuracy is derived by dividing the number of correct pixels in one class divided by the total number of pixels as derived from reference data (column total in confusion matrix). The producer's accuracy measures how well a certain area has been classified. It includes the error of omission which refers to the proportion of observed features on the ground that is not classified in the map. The more errors of omission exist, the lower the producer's accuracy. *Producer accuracy (%) = 100% – error of omission (%)*. If the correct classified pixels in a class are divided by the total number of pixels that were classified in that class, this measure is called user's accuracy.

The **Kappa coefficient** is a measure of overall agreement of a matrix. In contrast to the overall accuracy (the ratio of the sum of diagonal values to total number of cell counts in the matrix), the Kappa coefficient takes also non-diagonal elements into account [Rosenfield, G., et. al., 1986]. Developed originally by [Cohen, J., 1960] and introduced by [Congalton, R. G., 1983] to assess early Landsat satellite image classifications, the Kappa coefficient measures the proportion of agreement after chance agreements have been removed from considerations. Kappa increases to one as chance agreement decreases and becomes negative as less than chance agreement occurs. A Kappa of zero occurs when the agreement between classified data and verification data equals chance agreement. Most articles cite [Bishop, Y., et. al., 1975] as a source of formulation:

$$\hat{K} = \frac{N \sum_{i=1}^r x_{ii} - \sum_{i=1}^r (x_{i+} \cdot x_{+i})}{N^2 - \sum_{i=1}^r (x_{i+} \cdot x_{+i})}$$

Where,

r = number of rows and columns in error matrix,

N = total number of observations,

x_{ii} = observation in row i and column i ,

x_{i+} = marginal total of row ' i ', and

x_{+i} = marginal total of column ' i '.

The Kappa coefficient has become a standard for calculation of classification accuracy amidst the remote sensing community [Rosenfield, G., et. al., 1986] since this measure of calculation corrects for chance agreement, the entire confusion matrix is used in its derivation, a variance term can be calculated for it which facilitates statistical comparisons and also because scales exist to aid its interpretation [Congalton, R. G., 1983]. It has been widely used as a measure of accuracy in remote sensing problem of land use / land cover classification. This research shall also use this measure of accuracy for assessment of the performance of different CI algorithms.

1.2.2. Path planning and location prediction

The second application of terrain understanding undertaken in the thesis is path planning which includes location prediction and solving of the travelling salesman problem as its major components.

Location prediction is predicting the most suited destination from the available set of facts. In the thesis, the military application of intelligent battlefield preparation by anticipating the destination base station most suited for the enemy troops to position their forces, has been taken up. The ability to accurately predict the deployment strategy of the enemy troops is of critical importance to the commanders in the battlefield [Tzu, S., 6th Century B.C.]. There are a few works available in this field of anticipatory computing for situation awareness some of which are disaster situation awareness [Jaiswal, R. K., et. al., 2002], terrorist camps detection [Yasar, G. S., et. al., 2008], electronic situation awareness [Patrick, C. H., 2006], etc. However, no computing method is available for the prediction of enemy base station directly and the known available methods are still the conventional methods employed by military which involve manual study of geography of the area and various other factors. Thus, the main aim of the work done is to enhance the decision capability which will enable the commanders in the battlefield to make better strategies to combat the actions of enemy. This all would be possible if the ability to determine the enemy's probable base stations is available thereby leading to intelligent battlefield planning.

For the purpose, the application of location prediction assesses the best solution out of a set of candidate solutions based on their fitness values. This fitness value is modeled in terms of a *selectivity factor* associated with each solution. The selectivity factor for good solutions will be more and therefore these will have a higher chance of being selected as the best solution and

the selectivity factor for bad solutions will be less and hence these will have a lesser chance of being selected as the best solution. The selectivity factor is calculated for each of the candidate base stations and the selectivity factor values are arranged in descending order. These values are hence categorized according to standard ranges into good and bad solutions, the topmost element in the list being the one that is output as the best solution by the algorithm. The output of this application is then used by the path planning application to find the best path of attack on this predicted base station in order to design a two-phase recommender system.

Path planning is finding the safest which means an obstacle free as well as the shortest path of reaching the destination, given the source. The application integrates the process of finding the best path to a destination after fixing the most feasible destination for which it uses the location prediction application discussed above. The path planning application works on remote sensing input images. The aim is to find the *minimum no. of pixels* required to reach the destination pixel from the source pixel which is the shortest path. However, before proceeding with finding the shortest path, the application needs to categorize the image into accessible and inaccessible pixels of the image. The inaccessible pixels of the image are considered as obstacles. To make a path more smooth morphological operations are implemented to minimize the effects of shadows, trees and inconvenient areas so that an obstacle free path is found. The shortest path is found out from the remaining set of accessible regions of the image. The measure for finding out the next nearest pixel is taken as the *Euclidean distance*.

The **travelling salesman problem (TSP)** is given as: Given a list of cities and the distances between each pair of cities, the aim is to find the *shortest possible route that visits each city exactly once and returns to the origin city*. It is an NP-hard problem in combinatorial optimization, important in operations research and theoretical computer science. The problem is one of the most intensively studied problems in optimization and is used as a benchmark for many optimization methods. The TSP has several applications even in its purest formulation, such as planning, logistics, and the manufacture of microchips. Slightly modified, it appears as a sub-problem in many areas, such as DNA sequencing. Even though the problem is computationally difficult, a large number of heuristics and exact methods are known, so that some instances with tens of thousands of cities can be solved. Modern methods can find solutions for extremely large problems (millions of cities) within a reasonable time which are with a high probability just 2–3% away from the optimal solution. For the TSP problem, the fitness measure of the solution has been taken as the Plate Mobility Index which is defined as the equilibrium point indicating an optimal solution, the measure being inspired from the concepts of plate tectonics (discussed in chapter 6). For both of the above path planning

applications, the expected running times (ERT) of the algorithms are also found out based on which the computational complexity of the proposed algorithms in the thesis are discussed.

1.3. Computational Intelligence: State of the art

With the availability of high spatial resolution (0.5 meter) & radiometry (11-bit) data from 4th generation earth observing system, interpretation of data set using hitherto, techniques were yielding limited results. The concepts in geospatial sciences are generally vague, ambiguous and imprecise like the land-use / land-cover, extent of Great Indian Desert Thar, the paleo-channels and the urban sprawl. Also, a combination of spectral, spatial and radiometric resolution of space-borne sensors presents a selective & incomplete look of the geospatial feature/object under its view from the space. The resolution translates the satellite imageries into granular imprints of the geospatial features [Lillesand, T. M., et. al., 2008]. These necessitated newer procedures going and incorporating the structural knowledge into the terrain understanding process. Computational intelligence techniques like rough sets, fuzzy sets, BBO, ACO, PSO and their hybrid models, etc. have been used independently or in an integrated way for the development of optimization algorithms that have been targeted for some specific applications such as land cover feature extraction [(Panchal, V. K, et. al., 2009), (Bansal, S., et. al., 2009), (Gupta, S., et. al., 2011)], mixed pixel resolution [Varshney, T., et. al., 2014], face recognition [Gupta, D., et. al., 2013A], travelling tournament problem [Gupta, D., et. al., 2013B], aircraft engine health estimation [Simon, D., 2008], ECG diagnosis [Ovreiui, M., et. al., 2010], optimal power flow problem [Rarick, R., et. al., 2009] and many more.

The computational intelligence techniques [(Engelbrecht, Andries P., 2007), (Demetris S., et. al., 2006), (Alpaydin, E., 2004)] are composed of two major categorizations, first, the human cognition based techniques and the other those of nature inspired intelligence [(Johnson C.G., 2009), (Kari, L., et. al., 2008), (Yang, X.S., 2008)].

1.3.1. Human cognition based techniques

The role model for the human cognition based computational intelligence techniques is the human mind. All the approaches based on the modeling of human mind are tolerant of imprecision, uncertainty and partial truth and are used for improved decision-making and inductive reasoning. These techniques are able to achieve robust, low cost and best solutions however, may lead to exhaustive search and combinatorial explosion. The techniques require a static operating environment and run in polynomial time.

The computing techniques under this category are briefly described below.

- *Rough Set Theory* [(Pal, S.K., et. al., 1999), (Panchal, V. K, et. al., 2008)]: Rough set theory is a traditional mathematical approach for dealing with imprecise knowledge. The theory is based on the concept of finding hidden patterns in data in order to reduce search space and

find minimal sets of data with the same knowledge as in the original data. The main mechanism is to select an optimal subset called as the minimal reduct by applying the discernibility function and thus generate sets of decision rules from data thus leading to a straightforward interpretation of obtained results.

- *Fuzzy Set Theory* [(Pal, S.K., et. al., 1999), (Pawlak, Z., 1997), (Zadeh, L.A., 1965), (Kim, B. N., et. al., 1999)]: Fuzzy set theory is based on the concept of gradual assessment of the membership of elements in a set by using a graded scale with many intermediate values, like a number between 0 and 1 similar to what probability theory does. A fuzzy algorithm then uses fuzzy logic to operate on inputs by creating a small no. of flexible rules that use approximate or subjective values generating by describing the phenomenon or process linguistically and hence outputs the result.
- *Granular Computing* [(Bargiela, A., et. al., 2003), (Bargiela, A., et. al., 2006)]: Granular computing is based on the processing of complex information entities called information granules, which arise in the process of data abstraction and derivation of knowledge from information or data. The common goal of these methods in general is to find a representation of the data in terms of new variables, which are a linear or nonlinear transformation of the original variables, and in which important statistical relationships emerge. The resulting variable sets are almost always smaller than the original variable set and hence are very useful in reducing dimensionality, redundancy, and storage requirements of the original data thus leading to the design of more-effective machine-learning and reasoning systems.
- *Perception-Based Computing* [Yang, X.S., 2008]: Perception based Computing methods are based on the concept that learning can be accomplished at different perception resolution. It uses hierarchical levels of resolution in which each level adds some details to the constructed function till the final level can successfully reconstruct the whole function. This type of computing is suited for use in data mining and knowledge discovery in databases with dynamically evolving complex data (e.g. stream data sources, sensory data) [Yang, X.S., 2008].
- *Wisdom Technology* [(Johnson, C.G., 2009), (Yang, X.S., 2008)]: *Wisdom technology* is based on adapting judgments by taking into account iteratively all the knowledge available in the knowledge source network. Thus, the solutions are improved through interactive processing iteratively via observation and analysis of the experience during interaction with the environment and hence an optimal solution is reached [(Johnson, C. G., 2009), (Yang, X.S., 2008)].
- *Anticipatory Computing* [Yang, X.S., 2008]: Anticipatory computing systems simulate a process in discrete time steps and evaluate its behavior over time using forward or backward

appreciation of the differential equations. These equations are then used to generate a predictive model which anticipates the next best state of the system when at the current state thus progressing iteratively towards the optimal solution [Yang, X.S., 2008].

Initially researchers were using human cognition based techniques such as fuzzy logic and rough set theory for terrain understanding applications. The above computing techniques however, were not able to provide good result in case of ambiguity since the main goal of these techniques was to synthesize approximation of concepts from the acquired data. In addition, these techniques did not provide very much accurate results with low spatial resolution images. Also these techniques were not able to handle the crisp and continuous data separately and hence the focus shifted towards nature inspired techniques which aptly fit into the problem solving framework.

1.3.2. Nature Inspired Computational Intelligence techniques

The nature inspired computational intelligence techniques [Yang, X.S., 2008] have emerged as efficient optimization heuristics for problem solving in varied application domains and are categorized into techniques based on artificial immune system [(Dasgupta D., et. al., 2011), (Timmis J., et. al., 2008)] and those based on concepts of swarm intelligence [(Bonabeau, E., et. al., 1999), (Demetris S., et. al., 2006)].

A. Based on Artificial Immune System

Artificial Immune System [(Dasgupta D., et. al., 2011), (Timmis, J., et. al., 2008)] is guided by properties of human immune system for solving many information-processing systems and can be applied to many real life problems to obtain an optimized solution such as computer virus detection, fault diagnosis, robotics and control, anomaly detection, pattern recognition, optimization, etc. [Dasgupta, D., et. al., 2011]. The working principles of these techniques are pattern matching, feature extraction, learning and memory, diversity, distributed processing, self-organization, and self-protection. The main characteristics of the techniques are that they are robust, adaptive, dynamic, error-tolerant and decentralized. The advantage of these techniques is that they are able to implement much higher-level operations. However, the artificial immune system based techniques have no planned action and respond slowly. The running time of these techniques is exponential, however, it may be reduced to polynomial and linear times. The nature inspired computational intelligence techniques under this category are described below.

- *Neuro-Computing (ANN) [(Fausett, L., 1994), (Heerman, P. D., et. al., 1992), (Patterson, D., 1996), (Haykin, S., 1999)]:* Neuro-computing uses a statistical model with adaptive weights that can be tuned by a learning algorithm using parallel and local processing. These weights are used for training non-linear functions of the inputs and hence predict the optimal solutions based on the knowledge acquired by the neural network during the training phase.

Thus, neural networks are capable of handling non-convex decisions [(Fausett, L., 1994), (Heerman, P. D., et. al., 1992), (Patterson, D., 1996), (Haykin, S., 1999)].

- *Genetic Algorithm (GA)* [(Frietas, A. A., 1997), (Pappula, L., 2012), (Momm, H. G., et. al., 2009), (Niu, Q., et. al., 2010), (Brumby, S. P., et. al., 1999)]: Genetic algorithms are based on the concepts of natural evolution, such as inheritance, mutation, selection, and crossover [(Frietas, A. A., 1997), (Momm, H. G., et. al., 2009), (Niu, Q., et. al., 2010), (Brumby, S. P., et. al., 1999)]. In a genetic algorithm, a population of candidate solutions (called individuals, creatures, or phenotypes) to an optimization problem is evolved toward better solutions. Each candidate solution has a set of properties which can be mutated and altered based on the fitness value of the objective function. The algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population. The GA is applied to complex high dimensional, multimodal problems and often requires very expensive fitness function evaluations.
- *Cellular Automata* [Harald, N., 2006]: Cellular automata are dynamical systems discrete in space, time, and state variables, which are based on the concept of possession of exclusively local mechanisms of interaction by the use of reversible and bijective objective function for the search of an optimal solution. They constitute good models for the study of non-linear complex systems.
- *Membrane computing* [(Gupta, D., et. al., 2011), (Paun, G., 2002), (Gupta, N., et. al., 2011A), (Niall, M., et. al., 2011), (Velayutham, P., et. al., 2010)]: Membrane computing enriches the model of molecular computing by providing a spatial structure for molecular computation, inspired by the structure of living cell. The fundamental features that are used in this computing model are a membrane structure where objects evolve discretely according to specified evolution rules. The evolution rules are applied in a non-deterministic and maximally parallel way, and each object evolves according to some rule until no rules can be applied and the result is the multi-sets formed by the objects contained in a specific output membrane.
- *DNA Computing* [(Paun, G., et. al., 1998), (Lewin, D. I, 2002)]: DNA computing, a form of bio-molecular computing is based on the concept of parallel information processing where the different molecules of DNA try many different possibilities simultaneously. DNA computing allocates independent letter code to each object of interest. In this way, DNA sequences corresponding to the number of possible combinations are prepared. Next, they are hybridized in super parallel fashion and the remaining DNA fragments are amplified to obtain optimal result sequence thus leading to faster processing, remarkable energy efficiency and high storing capacity.

However, the problem solving methodology of the above techniques is not a planned one and hence responds slowly leading to exponential time complexities and therefore these techniques have limited applicability for terrain understanding applications.

B. Based on Swarm Intelligence

The solution to the above drawbacks was provided by the recently introduced concept of swarm intelligence [Engelbrecht, A. P., 2007]. *Swarm intelligence* techniques are guided by the concepts of information sharing to obtain the globally optimal solutions iteratively since these focus on collective behavior of “unintelligent” agents working in a decentralized manner. Swarm intelligence is any attempt to design algorithm or distributed problem-solving devices inspired by collective behavior of social insect colonies that can solve the problems in a very flexible and robust way: flexibility allow adaptation to changing environment, while robustness endows the colony with the ability to function even though some individuals may fail to perform their tasks [(Bonabeau, E., et. al., 1999), (Omkar, S.N., et. al., 2007)]. The swarm intelligence techniques are characterized by their autonomous, robust and self-adaptive nature. These techniques have predefined protocols and work in extremely complex and dynamic environments and achieve polynomial time complexity however, may be reduced to linear time. The advantages of these techniques is their faster convergence rate and a fewer parameters to adjust. The main application areas include engineering, biomedical and social sciences, data mining, networking and military applications. The nature inspired techniques under this category are described below:

- *Ant Colony Optimization* [(Dorigo, M., et. al., 2004), (Holden, N., et. al., 2005), (Parpinelli, R. S., et. al., 2002), (Bansal, S., et. al., 2009)]: Ant colony optimization algorithm is a multi-agent time-space variant probabilistic technique for solving computational problems which is based on the concept of simulation of ant’s behavior for finding shorter paths. The technique is based on the concept that ants secrete pheromone in order to communicate with one another to find the shortest path from the nest to the food. Since, pheromone accumulation is faster on the shorter path, the difference in pheromone content between the two paths over time makes the ants choose the shorter path. The idea of the ant colony algorithm is to use the concepts of positive feedback, distributed computation, and constructive greedy heuristic to mimic this behavior with simulated ants walking around the graph representing the problem to solve. Thus, based on the concept of information sharing, globally optimal solutions are found iteratively.
- *Particle Swarm Optimization* [(Kennedy, J., et.al., 1995), (Holden, N., et. al., 2005), (Wang, D., et. al., 2008), (Bansal, S., et. al., 2009)]: Particle Swarm Optimization is a multi-agent space variant population-based stochastic optimization algorithm that optimizes a problem by

iteratively trying to improve a candidate solution with regard to a given measure of quality. PSO optimizes a problem by having a population of candidate solutions called particles, and moving these particles around in the search-space according to simple mathematical formulae over the particle's position and velocity. Each particle's movement is influenced by its local best known position but, is also guided toward the best known positions in the search-space, which are updated as better positions are found by other particles. This is expected to move the swarm toward the best solutions. It has been applied successfully for solving highly mixed integer and nonlinear complex optimization problems.

- *Biogeography Based Optimization* [(Simon, D., 2008), (Panchal, V.K., et. al., 2009), (Goel, S., et. al., 2011)]: Biogeography Based Optimization is a living multi-agent time-space invariant population based evolutionary algorithm based on the migration, speciation, and extinction of species. BBO maintains a population of candidate solutions and optimizes a problem by stochastically and iteratively improving the candidate solutions by combining existing ones according to a given measure of quality, or fitness function. In this way the objective function is treated as a black box that merely provides a measure of quality given a candidate solution, and the function's gradient is not needed. BBO belongs to the class of meta-heuristics since it includes many variations, and since it does not make any assumptions about the problem and can therefore be applied to a wide class of problems.
- *Bacterial Foraging Optimization* [(Das, S., et. al., 2009), (Johal, N. K., et. al., 2010), (Passino, K. M., 2002)]: The bacterial foraging optimization algorithm is a living multi agent space variant algorithm belonging to a new class of biologically encouraged stochastic global search algorithms. The algorithm is based on mimicking the foraging behavior of E. coli bacteria. During foraging, a bacterium can exhibit two different actions: tumbling or swimming. The tumble action modifies the orientation of the bacterium. During swimming means the chemotaxis step, the bacterium will move in its current direction. Chemotaxis movement is continued until a bacterium goes in the direction of positive - nutrient gradient. After a certain number of complete swims, the best half of the population undergoes the reproduction and eliminating the rest of the population. In order to escape local optima, an elimination dispersion event is carried out where some bacteria are liquidated at random with a very small probability and the new replacements are initialized at random locations of the search space eventually leading to an optimal solution.
- *Bee Clustering Algorithm* [(Karaboga, D., 2005), (Karaboga, D., et. al., 2007), (Karaboga, D., et. al., 2009), (Dusan, T., et. al., 2005), (Banerjee, S., et. al., 2012), (Wong, L.P., et. al., 2008)]: Bee Colony Optimization is a living multi-agent space variant population-based search algorithm which mimics the foraging strategy of honey bees to look for the best

solution to an optimization problem. Each candidate solution is thought of as a food source (flower), and a population (colony) of agents (bees) is used to search the solution space. Each time an artificial bee visits a flower (lands on a solution), it evaluates its profitability (fitness) and performs a kind of neighborhood search combined with random search. Each search cycle is composed of five procedures: recruitment, local search, neighborhood shrinking, site abandonment, and global search. The cycle is iterated for a given number of times, or until a solution of acceptable fitness is found.

- *Artificial Swarm fish [Gao, Y., et. al., 2010]*: Artificial Fish Swarm Intelligence is a living multi-agent time-space variant stochastic, random and parallel search algorithm. The basic idea of the algorithm is to simulate the fish behaviors such as swarming, preying, local search of fish individual for reaching the global optimum. The Artificial Fish is characterized by its external perception of vision, its current state, the visual distance, and its visual position at some moment. If the state at the visual position is better than the current state, it goes forward a step in this direction, and arrives the state; otherwise, continues an inspecting tour in the vision. The greater number of inspecting tour the artificial fish does, the more knowledge about overall states of the vision the fish obtains. Certainly, it does not need to travel throughout complex or infinite states, which is helpful to find the global optimum by allowing certain local optimum with some uncertainty.
- *Cuckoo Search [(Johnson, C.G., 2009), (Kari, L., et. al., 2008), (Yang, X. S., et. al., 2010), (Gupta, D., et. al., 2013C)]*: Cuckoo search is a living single-agent space variant algorithm. The basic concept is that each egg in a nest represents a solution, and a cuckoo egg represents a new solution. Each cuckoo lays one egg at a time, and dumps its egg in a randomly chosen nest; the best nests with high quality of eggs will carry over to the next generation. The number of available host nests is fixed, and the egg laid by a cuckoo is discovered by the host bird with a probability ranging from 0-1. Discovering operates on some set of worst nests, and discovered solutions dumped from farther calculations. Thus, the algorithm uses the new and potentially better solutions (cuckoos) to replace not-so-good solution in the nests thus reaching optimal solutions.
- *River Formation Dynamics [Rabanal, P., et. al., 2010]*: River formation dynamics is a non-living single agent time-space invariant heuristic method similar to ant colony optimization and can be seen as a gradient version of ACO where instead of associating pheromone values to edges, altitude values are associated to nodes. River drops erode the ground (they reduce the altitude of nodes) or deposit the sediment (increase it) as they move. The probability of the drop to take a given edge instead of others is proportional to the gradient of the down slope in the edge, which in turn depends on the difference of altitudes between both nodes

and the distance (i.e. the cost of the edge). Drops are unleashed at the origin node, which spread around the flat environment until some of them fall in the destination node. This erodes adjacent nodes, which creates new down slopes, and in this way the erosion process is propagated. New drops are inserted in the origin node to transform paths and reinforce the erosion of promising paths. After some steps, good paths from the origin to the destination are found thus finding the optimal solution.

- *Gravitational Search* [(Rashedi, E., et. al., 2010), (Bhattacharya, A., et. al., 2012)]: Gravitational search algorithm is a non-living multi-agent space variant based on the law of gravity and the notion of mass interactions where it is considered that the gravitational force on the masses is inversely proportional to the distance between them. In the algorithm, agents are considered as objects and their performance is measured by their masses. The agent has four parameters which are position, inertial mass, active gravitational mass, and passive gravitational mass. The position of the mass represents the solution of the problem, where the gravitational and inertial masses are determined using a fitness function. The algorithm is navigated by adjusting the gravitational and inertia masses, whereas each mass presents a solution. Masses are attracted by the heaviest mass. Hence, the heaviest mass presents an optimum solution in the search space.
- *Charged System Search* [Kaveh, A., et. al., 2010]: Charged System Search is a new non living multi-agent space variant optimization algorithm. In the algorithm, each candidate solution containing a number of decision variables is considered as a charged particle. The charged particle is affected by the electrical fields of the other agents. The quantity of the resultant force is determined by using the electrostatics laws, and the quality of the movement is determined using the Newtonian mechanics laws and thus the amount of charge will be defined considering the objective function value. The charged particle moves towards good agents which exert a stronger force than the bad ones, thus improving their performance through self-adaptation and leading towards optimal solutions.
- *Intelligent Water Drop* [(Duan, H., et. al., 2008), (Kamkar, I., et. al., 2010)]: Intelligent water drops algorithm is a non-living single-agent space variant inspired by natural rivers and how they find almost optimal paths to their destination. These near optimal or optimal paths follow from actions and reactions occurring among the water drops and the water drops with their riverbeds. Almost every algorithm is composed of two parts: a graph that plays the role of distributed memory on which soils of different edges are preserved, and the moving part of the IWD algorithm, which is a few number of intelligent water drops. These intelligent water drops construct their solution gradually until all IWDs have completed their solutions which constitute one iteration. After each iteration, the iteration-best solution is found and it is used

to update the total-best solution. The amount of soil on the edges of the iteration-best solution is reduced based on the goodness (quality) of the solution. Then, the algorithm begins another iteration with new IWDs but with the same soils on the paths of the graph and the whole process is repeated. The algorithm stops when it reaches the maximum number of iterations or the total-best solution reaches the expected quality.

- *Firefly Optimization* [(Johnson, C.G., 2009), (Yang, X. S., 2010)]: Firefly algorithm is a living single agent time-space invariant algorithm inspired by the flashing behavior of fireflies. The algorithm initializes any one of the generated population as the brightest firefly and the rest of the fireflies are then moved towards that brightest firefly which represents the optimal solution. During this process, the distance and attractiveness of each firefly from the brighter one is calculated based on an objective function which is proportional to the brightness and affects the process of movement for each firefly differently. On successful completion of this moving procedure, the fireflies are ranked according to their performance and the objective function is modified accordingly and the next iteration continues until the algorithm converges to the optimal solution.
- *Stochastic Diffusion Search* [(Johnson, C.G., 2009), (Bishop, J. M., 1989), (Beattie, P. D., et. al., 1998)]: Stochastic diffusion search is a space variant and an agent-based probabilistic global search optimization technique. In SDS agents perform cheap, partial evaluations of a hypothesis (a candidate solution to the search problem). They then share information about hypotheses (diffusion of information) through direct one-to-one communication. As a result of the diffusion mechanism, optimal solution can be identified from clusters of agents with the same hypothesis.

Swarm Intelligence techniques like the Ant Colony Optimization [(Holden, N., et. al., 2005), (Omkar, S.N., et. al., 2007)], Particle Swarm Optimization [(Omkar, S.N., et. al., 2007), (Kaur, P., et. al., 2010), (Bansal, S., et. al., 2009), (Wang, D., et. al., 2008)], etc. improve the terrain understanding process of satellite multi-spectral images and are more accurate when working with low spatial resolution images and continuous data, however, only few swarm intelligence based techniques could be adapted and implemented for real time scenarios of terrain understanding applications. Recently, some swarm intelligence techniques are known like biogeography based optimization, bacterial foraging optimization, intelligent water drops etc. which have been explored to some extent but their detailed behavior for terrain understanding applications has not yet been fully analyzed.

1.4. Problem Statement

The biogeography based optimization technique has gained widespread popularity recently because of its ability to incorporate expert knowledge into the terrain understanding process

and model selective aspects or properties of the terrain under consideration over the others. The technique improves its solutions probabilistically with each iteration and is flexible for application in diverse terrain understanding applications.

As the frontiers of space technology advance, the knowledge derived from the satellite data has also grown in sophistication. There is a need for more accurate and efficient terrain understanding algorithms and therefore, no single technique can prove to satisfactorily understand all the basic properties of the terrain under consideration. It is observed that hybrid versions of these techniques with the existing optimization techniques can prove to drastically improve the optimization capability hence leading to better terrain classification and understanding. For example, hybrid versions of a few swarm intelligence techniques have been developed like the hybrid ACO2/PSO approach for feature selection application, hybrid GA-ACO methodology for multiple assignment, vehicle routing applications, etc. [(Freitas, A.A., 1997), (Pappula, L., 2012), (Johal, N. K., et. al., 2010), (Niu, Q., et. al., 2010), (Kaur, P., et. al., 2010)]. In addition to this, there is a scope to enhance the biogeography based optimization technique by incorporating the factors of extinction and evolution in the species abundance model and a dependency factor in its migration process. This enhancement is likely to streamline the process of migration by improving the quality of the solutions preliminarily and enable the applicability of the technique to non-separable problems thus leading to better results than the original model in those cases too where the knowledge about the problem structure is required for problem solving.

Another domain of computational models can be developed based on the geo-sciences concepts under the nature inspired techniques. This domain has never been explored to develop such models. The nature and natural phenomenon like that of earthquakes, the phenomenon of plate tectonics [(Gornyi, V.I., 2002), (Briggs, J.C., 1987), (Moresi, L., et. al., 2000)], etc. all are based on the principle of nature which says that every constituent part of nature all exhibit an inherent quality of stabilizing to an equilibrium point, deviating from the equilibrium point either in the positive or the negative direction will lead to instability. The equilibrium point can be considered as the optimum solution to be found out and the positive and negative excursions from it as the good and the bad solutions. There is a need to reach the equilibrium point to get the optimum solution since there is always a tradeoff between the cost of evaluating a heuristic optimization function and the improvement in the solution it provides. Thus, these models possess the characteristic of providing a guaranteed optimal solution.

Hence the research goal of the thesis is extracted as below:

“Developing optimization techniques based on nature inspired computational intelligence by formulating their mathematical model and then adapting them for the terrain understanding application.”

This goal can be achieved by addressing the following sub-problems:

- [a] To improve the optimization technique inspired from biogeography which is also most popular and flexible by analyzing the performance governing factors of this algorithm for the terrain understanding application. These factors governing the optimization process can act as key in developing mathematical model with improved optimization capabilities. In addition to this, there is a need to analyze the BBO behavior on feature extraction application which can be used to develop theoretical framework for the concept of information sharing in swarm intelligence techniques leading towards optimal solutions.
- [b] Explore the hybrid models of the biogeography based optimization technique with the other nature inspired intelligent techniques and adapt these models for the targeted terrain understanding applications. Then, evaluate their performance using the respective accuracy assessment metrics and compare with the existing models. This requires exploring the behavioral analysis of the optimization techniques by developing mathematical models governing their performance and studying the merits and demerits of each in order to proceed to propose their hybridization model, which can improve the results obtained individually.
- [c] In the process of developing hybrid models and studying the factors governing their performance from the terrain understanding perspective, there is a need to move ahead and focus on extending the basic functionalities of optimization of existing nature inspired optimization techniques. Thus, a good attempt can be made to extend the original species abundance model of BBO by incorporating the factors of extinction and evolution and formulate the extended model of BBO.
- [d] The generated extended models can then be adapted for both of the terrain understanding applications taken in the thesis that of path planning and geo-spatial feature extraction. The key factors identified during the performance analysis such as the HSI function of BBO can be enhanced by making its behavior dynamic which is static in original BBO. This enhancement is likely to improve the optimization process.
- [e] Based on the study of mechanisms used for developing extended optimization models of biogeography, the final goal is to develop a novel optimization paradigm based on the phenomena of plate tectonics. This model will use the concept of plate mobility index (PMI) – weights with decision variables having tolerance, dynamic adjustment of weights.

This will lead to the development of an optimization algorithm that will utilize an optimal set of decision variables until all the problem constraints are satisfied.

Hence, the thesis attempts to achieve the following sub-goals:

- (i) Development of “performance analysis models” of the biogeography based optimization technique for terrain understanding application of geo-spatial feature extraction.**
- (ii) Development of “hybrid models” of BBO with other nature inspired techniques for terrain understanding application of geo-spatial feature extraction.**
- (iii) Development of “extended models” of the biogeography based optimization technique to develop the BBO/EE technique.**
- (iv) Development of “adaptive models” of the BBO/EE technique for both the terrain understanding applications of feature extraction and military application of path planning including location prediction.**
- (v) Development of “new generic model” inspired from plate tectonics phenomenon, PBO, and its application to the travelling salesman problem.**

1.5. Thesis Approach

This section describes the solution approach that the thesis adopted for solving the above discussed problems. The above sub-problems of the thesis can be viewed as composed of five models. These models are not independent but are related to each other. The thesis begins with the performance analysis of BBO and proposes hybrid models based on the above analysis for geo-spatial feature extraction. The thesis moves a step ahead by extending the original functionalities of the BBO model and developing the BBO/EE technique which is adapted to both of the applications dealt with in the thesis. Finally, the thesis sets its goal to develop a novel optimization paradigm, the plate tectonics based optimization (PBO), based on plate tectonics concepts. The following subsection presents the approach adopted by us to address the key problems.

1.5.1. Performance analysis models

The research focuses on improving the algorithm of BBO by analyzing its performance governing factors for the terrain understanding application of geo-spatial feature extraction. The approach adopted is to study the behavior of BBO in terms of its classification efficiencies over individual land cover features on different multi-spectral satellite datasets. For establishment of the proposed relationship between the classification efficiency and the factors governing it, the classified images are taken from the biogeography based feature extractor results and the fact that the actual classification efficiencies that were provided by the extractor follow the equation governing the performance of BBO that is derived in the analytical study is

validated. Thus, the performance governing equation of BBO is established which states that the classification efficiency for a given land cover feature is proportional to the degree of similarity index and inversely to the degree of entropy. The BBO behavior is also studied on vegetation, water, urban, rocky, barren and snowy features on two different datasets that of 7-band resource-sat satellite image of size 472×546 of the Alwar area in Rajasthan and the 4-band landsat image of size 508×744 of the Patalganga region in Shivalik ranges (provided courtesy of DRDO). The above study establishes the fact that BBO is able to classify the homogeneous regions more efficiently. The thesis then attempts to establish the concept of information sharing in basic swarm intelligence techniques which is the key factor governing the heuristic function definition and leading towards optimal solutions. This establishes a theoretical framework for establishing the concept of information sharing in nature inspired swarm intelligence techniques for the terrain understanding application.

1.5.2. Hybrid Models

As hybrid models can improve the classification efficiency of nature inspired classifiers [(Freitas, A.A., 1997), (Pappula, L., 2012), (Johal, N. K., et. al., 2010), (Niu, Q., et. al., 2010), (Kaur, P., et. al., 2010)] by achieving a better kappa coefficient, thus, hybrid versions of BBO with ACO and PSO are developed and adapted for the targeted terrain understanding applications under this work. The designed hybrid classifiers have been applied to the 7-band satellite image of Alwar area in Rajasthan and the 4-band image of the Patalganga region in Shivalik ranges and the performance of the classification results is then evaluated by constructing an error matrix and using the accuracy assessment metric, the kappa coefficient, as described in section 1.2.

Two hybrid models are proposed in the thesis wherein the first model is based on combining the strengths of the ACO2/PSO technique with the characteristic strengths of BBO which turns this classifier intelligent thus developing a hybrid model for geo-spatial feature extraction. The hybrid ACO2/PSO/BBO classifier exploits the characteristic feature of the BBO technique to show a wide range of efficiencies in classifying different features of an image as is also reflected from the performance governing model of BBO. The classifier is based on the capability to identify the features efficiently classified by BBO and by ACO2/PSO separately based on the proposed performance governing model in turn based on a similarity pattern analysis of the training set data distribution graph and therefore the classifier is called as the hybrid bio-inspired pattern analysis based intelligent classifier.

The second model is based on integrating the modified biogeography based extractor with geo-sciences concepts and substituting the original BBO by this BBO-GS in the previously developed hybrid ACO2/PSO/BBO classifier. The hybrid ACO2/PSO/(BBO-GS) based feature

extractor is an improvement over the original hybrid bio-inspired classifier. The classifier is able to classify the homogeneous regions (the regions with the lower entropy), more efficiently than the other hybrid classifiers. This also validates the theory that was earlier established which states that the classification efficiency of biogeography based extractor on a given land cover feature is proportional to the degree of entropy for that feature. The above designed hybrid classifiers achieve very high kappa coefficients and prove to be great advancements over the existing classifiers.

A comparative study of the results of applying the proposed hybrid classifiers and the other recent computational intelligence based classifiers such as ACO, Hybrid PSO-ACO2, Hybrid ACO-BBO Classifier, Fuzzy sets, Rough-Fuzzy Tie up and the Semantic Web Based Classifiers when compared with the traditional probabilistic classifiers such as the Minimum Distance to Mean Classifier and the Maximum Likelihood Classifier on the benchmark image of Alwar has also been made.

1.5.3. Extended Models

In the process of developing hybrid models, the thesis identifies the factors governing the optimization process in the original BBO model and attempts to extend the original species abundance model of biogeography based optimization proposed by Simon by incorporating the factors of extinction and evolution in the calculation of the species count, for the characterization of a habitat with a high or a low HSI value. The approach is also based on including an additional dependency factor which signifies the interdependence of migrating species on each other; to be considered for the determination of immigration and emigration rates. Based on the proposed model of species abundance, the extended model of BBO is hence formulated: the Biogeography based optimization with Extinction and Evolution (BBO/EE) model with the concept of extended HSI function and incorporation of dependency factor in migration process. Hence, the proposed model uses the basic approach of streamlining the process of migration by improving the quality of the solutions preliminarily thus leading to better results than the original model. Since BBO/EE also considers the dependency among the candidate solutions, therefore it is also suitable for non-separable problems.

Six extended representative migration models of BBO are also presented. The first three are extended versions of the linear models, which mean that the immigration rate and the emigration rate are linear (or at least affine) functions of the number of species, and the other three are extended versions of the non-linear models. The performance of each of the extended models of BBO is demonstrated by running them on standard benchmark functions and it is found that the average convergence of each of the proposed extended species abundance

models is faster leading to better optimization results than the original species abundance model of BBO.

1.5.4. Adaptive Models

The extended models of optimization algorithms are hence adapted for the terrain understanding problem. The built extended BBO/EE model is adapted by modifying the BBO/EE parameters and integration with ACO and PSO for adapting it to the military application of battlefield situation awareness - the design of BBO/EE based Two-phase Recommender System Model for Prediction of Enemy Troop Deployment Strategies. The approach used in the design of the above application is to serve a dual purpose by predicting the deployment strategies of enemy troops in the battlefield and also finding the shortest and the best feasible path for attack on the enemy base station. The work introduces a new concept of efforts required in migration to a high HSI solution for optimization in BBO and hence also proposes an advanced optimization technique. The designed system is a milestone in the history of automated recommender system design for battlefield planning when compared with traditional decision making and other contemporary techniques.

The BBO/EE parameters are also modified to present a dynamic model of the BBO for land cover feature extraction based on extended HSI function- the BBO/EE based land cover feature extractor model. This work presents a land cover feature extraction technique based on the extended species abundance model of biogeography where the approach is to consider the HSI as a function of different combinations of SIVs depending upon the characteristics of the habitat under consideration and propose an extension to the classical BBO. Making use of the proposed hypotheses, the *HSI* of each of the habitats representing the image pixels is calculated using two different functions namely 'entropy' and 'deviation from mean metric' and hence maximize the classification efficiency achieved by adapting to dynamic changes in the HSI function definition and propose the extended model of the original biogeography based feature extraction technique which modeled the HSI function as static. The proposed algorithm has been successfully tested on two different multi-spectral satellite image datasets. The above extended model is also incorporated in the previously designed hybrid bio-inspired intelligent classifier and its performance is compared with the original hybrid classifier and twelve other classifiers on the 7-Band Alwar Image wherein it is observed that the proposed BBO/EE based hybrid classifier is able to achieve the kappa coefficient which is the highest amongst all the classifiers developed till date and far outperforms the standard BBO and the other optimization algorithms based extractors as well and hence proposes an advanced optimization algorithm.

1.5.5. New Generic models

Based on the study of the above models and the mechanisms used therein for developing extended optimization models of biogeography, the thesis finally attempts to invent a new generic technique inspired from the concepts of plate tectonics which is termed as the plate tectonics based optimization as a novel optimization paradigm. The theory of plate tectonics is a scientific theory that describes the large scale motions of Earth's lithosphere. This motivates the application of the concepts and mathematics of plate tectonics as the basis for the development of a new optimization paradigm. The thesis discusses the driving forces of continental drift / plate motion and their mathematical equations, and then discusses how these can be modeled to solve optimization problems. The proposed approach is based on the capability of incorporating lower prioritized decision variables iteratively and achieving a better solution with each iteration. Also, it incorporates the concept of dynamic adjustment of weights according to the conveyor-belt principle of plate tectonics.

The performance of the proposed optimization algorithm is analyzed on a set of 14 standard benchmark functions and the results are compared with the other state of the art evolutionary optimization methods. From the results, it is concluded that the proposed approach is the fastest among all the optimization algorithms analyzed, that it is the least computationally complex algorithm among all the other evolutionary optimization algorithms compared and that it removes the disadvantage of the population based optimization algorithms, one that is related to computational effort, i.e. the problem of creating infeasible solutions and can be successfully applied to practical problems especially when working in constantly changing environments. Thus, the proposed algorithm is found to outperform BBO and other contemporary evolutionary algorithms. Based on the above characteristic strengths of the approach, its applicability on the travelling salesman problem (which is similar to path planning application) in terrain understanding application is demonstrated on a 5-cityset and then extended to demonstrate it to standard TSP problem instances which can further be tested on path planning applications on real world satellite datasets. In the end, the performance is compared with twelve other recent evolutionary optimization techniques for the TSP problem for all the datasets taken.

1.6. Contribution of the thesis

An overview of the research work accomplished in the thesis and its targeted application areas is presented in figure 1.1 under the respective models. The main contributions of the thesis are summarized below:

- [1] Establishment of the concept of information sharing in swarm intelligence techniques which is the key factor governing the heuristic function definition and leading towards

optimal solutions [*Published in the International Journal of Computer Applications in the year 2011 and as a book chapter in the book on “Knowledge based processes” by IGI Global Publications in the year 2014 (Goel, L., et. al., 2011B), (Goel, L., et. al., 2014)]]*. The thesis establishes entropy and similarity index as the performance governing factors influencing the classification efficiency of biogeography based classifier. [*Published in the International Conference Proceedings of WICT, IEEE Publications, Mumbai in the year 2011 and the International Journal of Applied Evolutionary Computation, IGI Global Publications in the year 2013 [(Goel, L., et. al., 2011A), (Goel, L., et. al., 2013C)]]*].

- [2] Design of two new hybrid classifiers based on the performance governing model of BBO. The first is an integration of BBO with ACO2/PSO and second, is an integration of BBO-GS with ACO2/PSO. The first classifier is able to identify the features efficiently classified by BBO and by ACO2/PSO separately and is a hybrid bio-inspired pattern analysis based intelligent classifier. [*Published in the Applied Soft Computing Journal of Elsevier Publications in the year 2012 [Goel, L., et. al., 2012A]]*. The second classifier is able to classify the homogeneous regions of the image very efficiently and proves to be among the best known classifiers developed. [*Published as a book chapter in the Advances in Soft Computing series (AISC), Springer Publications in the year 2011 and the Applied Soft Computing Journal of Elsevier Publications in the year 2013 [(Goel, L., et. al., 2011C), (Goel, L., et. al., 2013A)]]*].
- [3] Design of a new optimization algorithm – BBO/EE and identification of the factors of extinction and evolution with the concept of different HSI functions for different habitats thus streamlining the process of migration by improving the quality of the solutions preliminarily and hence leading to better results than the original model. The above factors have contributed towards the sophistication of the existing model of biogeography. Also, the proposed model also considers the correlation (dependency) among the candidate solutions and hence it is suitable for non-separable problems and will be able to perform better than the other contemporary optimization algorithms in those cases where the knowledge about the problem structure is required for problem solving. [*Published in the IEEE Computer Science Digital Library during the International Conference IEEE CIMSIm, Malaysia in September, 2012 [Goel, L., et. al., 2012D] and revision under process in the Information Sciences Journal, Elsevier Publications in the year 2015]*].
- [4] Application of BBO/EE for the design of military application of enemy base station prediction and best feasible path finding targeted for two natural terrain scenarios - serving a dual purpose. The designed system can be of major use for the commanders in the

battlefield who have been using traditional decision making techniques of limited accuracy for predicting the destination. Using the above computation technique can help in enabling the commanders in the battlefield for intelligent preparation of the battlefield by automating the process of assessing the likely base stations of the enemy and the ways in which these can be attacked, given the environment and the terrain considerations. The above application is a milestone in the history of automated recommender system design for battlefield planning. *[Published in the International Conference Proceedings of HIS, IEEE Publications, Pune in December, 2012 and as a book chapter in the book titled “Case Studies in Intelligent Computing: Achievements and Trends” published by Taylor and Francis Publications in the year 2014 [(Goel, L., et. al., 2012C), (Goel, L., 2014)]]*.

- [5] Feature extraction using BBO/EE with dynamic HSI function. Making use of the proposed hypotheses, the *HSI* can be calculated for each of the habitats representing the image pixels using different functions and hence maximize the classification efficiency achieved by adapting to dynamic changes in the HSI function definition. This application when integrated with the existing hybrid bio-inspired classifier is able to achieve the highest classification efficiency amongst all the classifiers developed till date. *[Published as a book chapter in the Communications in Computer & Information Sciences (CCIS-LNCS), Springer - Verlag Berlin Heidelberg Publications in the year 2012 and the International Journal of Transactions on Computational Science, Springer - Verlag Berlin Heidelberg Publications in the year 2013 [(Goel, L., et. al., 2012B), (Goel, L., et. al., 2013B)]]*.
- [6] Design of a new algorithm based on plate tectonics called as the plate tectonics based optimization. This algorithm has introduced the novel concept of plate mobility index – weights with decision variables having tolerance and also the concept of dynamic adjustment of weights. This concept has been used for the development of an optimization algorithm called as the plate tectonics based optimization that utilizes an optimal set of decision variables until all the problem constraints are satisfied. The algorithm has shown improved results for standard TSP problem instances and proved to be better than other EAs on standard benchmark functions.
- [7] Extension in the taxonomy of Nature Inspired Computational Intelligence to include the techniques inspired from geo-sciences which had been left unexplored till date. *[Communicated to the Applied Soft Computing Journal of Elsevier Publications in the year 2014]*.

1.7. Outline of the Thesis

The rest of the thesis is organized as follows:

Chapter 2 deals with the analytical models of nature inspired computational intelligence. The mathematical framework for the behavioral analysis of the BBO technique over natural terrain features is presented by means of a combined tabular representation of the percentage of classification efficiency, the degree of entropy and the similarity index for each land cover feature for each of the multi-spectral bands taken on two different datasets.

Chapter 3 deals with the hybrid models of nature inspired computational intelligence. Based on the performance analysis model of BBO, the chapter presents a hybrid biologically inspired technique, called the ACO2/PSO/BBO technique that can be adapted according to the database of expert knowledge for a more focused satellite image classification. Also, the hybrid ACO2/PSO/(BBO-GS) technique is presented which is an improvement over the previous one.

Chapter 4 deals with the extended models of nature inspired computational intelligence. It proposes the modified species abundance model that was originally proposed by Dan Simon and calls it as the extended biogeography based optimization with extinction and evolution (BBO/EE) technique. Six extended migration models are also presented that were originally proposed by Haiping Ma as an extension to the work by Simon and also the performance of the proposed extended models are demonstrated on standard benchmark functions.

Chapter 5 deals with the adaptive models of nature inspired computational intelligence. Based on the modified BBO/EE model, a two - phase recommender system architecture is designed for intelligent preparation of the battlefield. The chapter also demonstrates the applicability of BBO/EE for land cover feature extraction of satellite images and compares the performance of the BBO/EE based hybrid feature extractor with the original hybrid classifier and twelve other optimization methods that have been adapted till date for feature extraction application.

Chapter 6 deals with the exploration of new generic optimization models of nature inspired computational intelligence. In this chapter, the entirely new geo-sciences based phenomenon of plate tectonics is studied and the concepts and mathematics of plate tectonics are used as the basis for the development of a new optimization paradigm. The chapter also compares and contrasts the approach with other evolutionary optimization methods by comparing their performances on a set of benchmark functions and also demonstrates its applicability on the travelling salesman problem first with a 5-city dataset for demonstration and then to five standard TSP problem instances.

Chapter 7 highlights the key findings of the research. It concludes the thesis with pointers for prospective future work. Additionally referred briefs are some of the prospective problems in the field that are unresolved as of now.

1.8. Observations

This chapter has given an overview of the whole work. The main objective and scopes are described. The chapter also provides a background of the works described in this thesis. An overview of the research work accomplished in each chapter of the thesis and its targeted application areas is provided on the title page of each chapter along with the accuracy assessment metrics used.

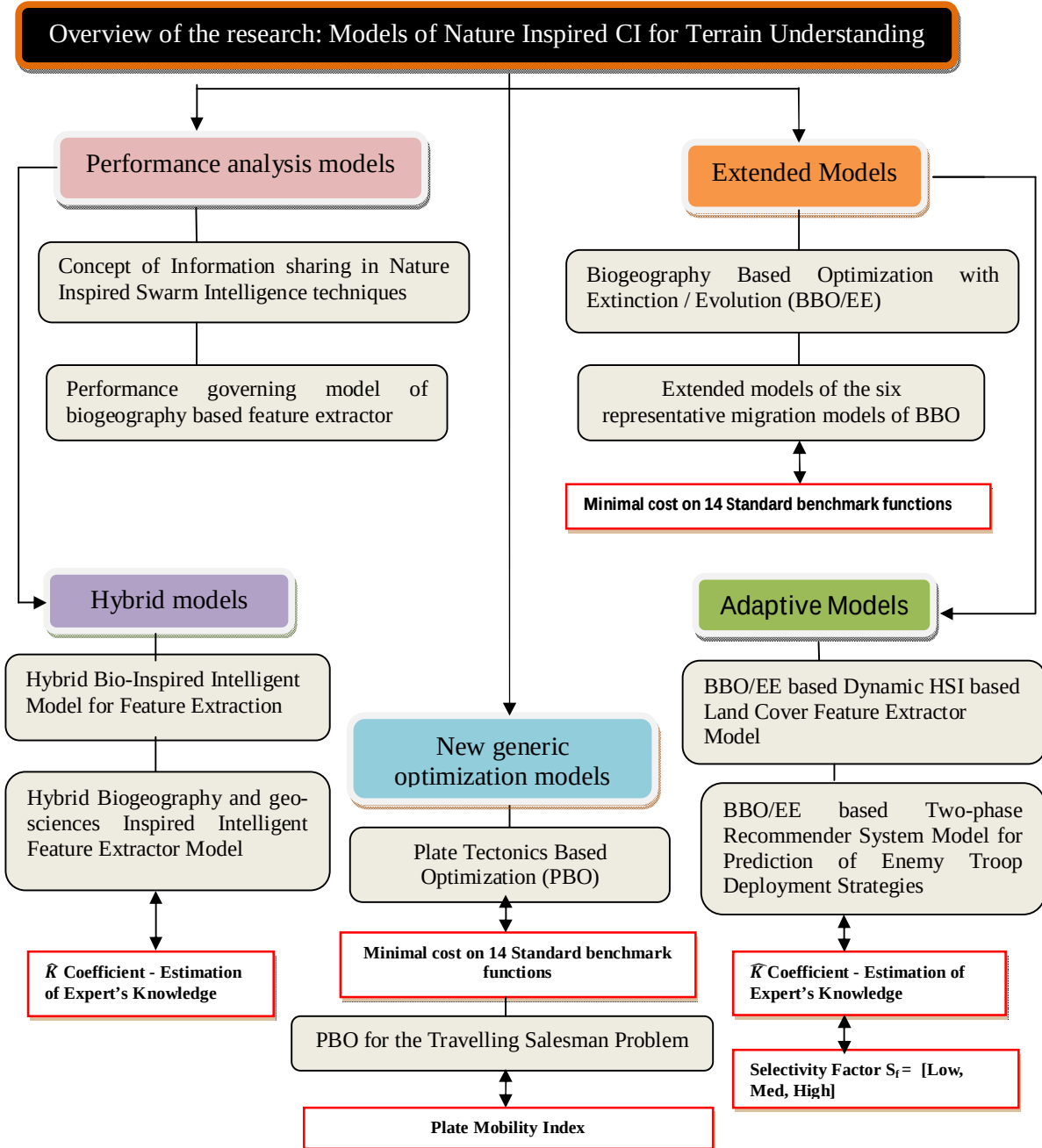
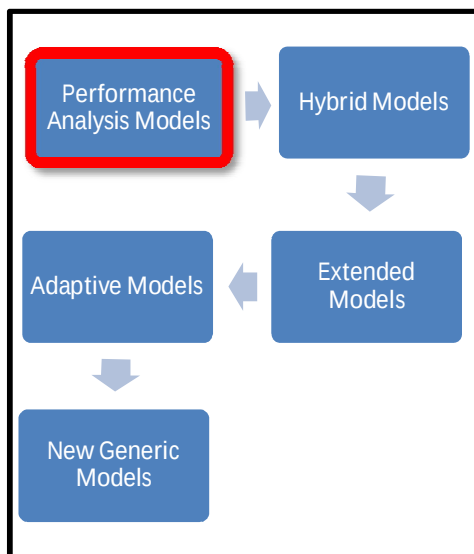
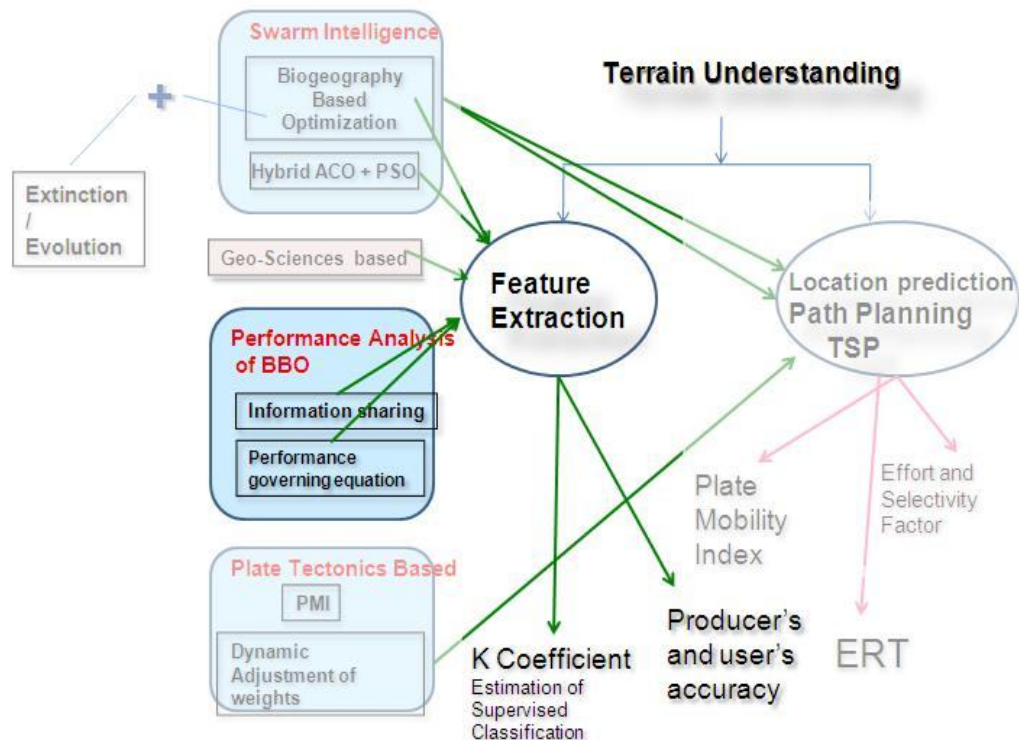


Figure 1.1: Framework of the research work in the thesis in nature inspired CI and its application areas undertaken. The respective accuracy assessment metrics used are given in the red boxes below each work.

Performance Analysis Model of Biogeography Based Land Cover Feature Extractor



Chapter 2

- 2.1. Biogeography based Optimization
- 2.2. Biogeography Based Framework for Land Cover Feature Extraction
- 2.3. Performance Analysis of Biogeography Based Land Cover Feature Extractor
- 2.4. Implementation, Results and Analysis
- 2.5. Information Sharing In Swarm Intelligence Techniques
- 2.6. Observations

Various swarm intelligence techniques such as the ACO [(Dorigo, M., et. al., 2004), (Dorigo, M., et. al., 2004), (Otero, F. E. B., et. al., 2008), (Parpinelli, R. S., et. al., 2002), (Xiaoping, L., et. al., 2008)], PSO [(Bratton, et. al., 2007), (Clerc, M., 2006), (Eberhart, R., et. al., 2001), (Wang, D., et. al., 2008)], hybrid ACO2/PSO optimization [(Omkar, S. N., et. al., 2007), (Holden, N., et. al., 2005), (Bansal, S., et. al., 2009), (Open Source Tool, SourceForgenet)] have been used for solving the problem of satellite image classification using the concepts of information sharing [Goel, L., et. al., 2011B]. Other computational intelligence techniques like those of semantic web based computing [Kumar, S., et. al., 2009], membrane computing [Gupta, D., et. al., 2011] and recently developed swarm intelligence technique of hybrid FPAB/BBO [Johal, N. K., et. al., 2010] have also been used for the purpose. However, in order to choose the best classifier for extracting land cover features, one should study the strengths and weaknesses of these algorithms and hence conducting a performance analysis of these algorithms is very important [(Hand, J., 1997), (Long, W. III, et. al., 2004), (Lillesand, T. M., et. al., 2007)].

From the study of the recent works where the original BBO algorithm has been adapted and applied for land-use / land-cover feature extraction from satellite images [(Goel, L., 2010B), [(Hand, J., 1997), (Panchal, V. K., et. al., 2009)], it can be concluded that the BBO technique is flexible to classify the desirable features more efficiently than the other features [(Goel, L., et. al., 2010C), (Panchal, V. K., et. al., 2009), (Panchal, V. K., et. al., 2011)]. In this chapter, the performance analysis model of the BBO technique over natural terrain features for satellite image classification is presented. This is derived from the study of the behavior of BBO for individual land cover features of vegetation, water, urban, rocky, barren and snowy regions on two different datasets of Alwar and Patalganga region and calculating their entropies and similarity indices. Thus, the key factors governing the performance of BBO are identified. These key factors establish the concept of information sharing in basic swarm intelligence techniques of ACO, PSO, BBO and BFO leading towards optimal solutions. [Goel, L., et. al., 2011B].

The organization of the chapter is as follows: Section 2.1 gives an overview of the BBO technique. Section 2.2 presents the adapted biogeography based framework for land cover feature extraction. Section 2.3 presents the proposed methodology for the behavioral analysis of the biogeography based feature extractor. Section 2.4 presents the implementation details and the analysis of BBO behavior based on the classification results on Alwar and Patalganga terrain. Section 2.5 attempts to establish the concept of information sharing in basic swarm intelligence techniques. Section 2.6 concludes the work and discusses the observations made.

2.1. Biogeography Based Optimization

This section gives an overview of how the concepts of biogeography are modeled into the BBO algorithm.

2.1.1. BBO concept

The BBO technique considers an island (habitat) with favorable conditions for migration as having a higher habitat suitability index (analogous to fitness in other EAs) which is dependent on factors called as the suitability index variables. Thus, the habitats with a high HSI are characterized by a larger number of species, lower species immigration rate, higher species emigration rate and a static species distribution and vice versa for low HSI habitats. The optimization in BBO is achieved through migration of species from low HSI habitat to a high HSI habitat since immigration of new species to low HSI habitats may raise the HSI of the habitat which is because the suitability of a habitat is proportional to its biological diversity.

Suppose that there is a problem and a population of candidate solutions that can be represented as vectors of integers. Each integer in the solution vector is considered to be an SIV. The good solutions are considered to be habitats with a high HSI, and vice versa for poor solutions. Figure 2.1 represents a model of species abundance in a single habitat with two candidate solutions S_1 and S_2 where S_1 represents a low HSI solution, while S_2 represents a high HSI solution. The immigration rate λ and the emigration rate μ are functions of the number of species in the habitat. The maximum possible immigration rate to the habitat is I , which occurs when there are zero species in the habitat at which point the emigration rate is zero. The maximum emigration rate is E , which occurs when the habitat contains the largest number of species, $S_{\max}$, that it can support at which point the immigration rate becomes zero. The equilibrium number of species, S_0 , is reached at the point when the immigration and emigration rates are equal. Thus, the immigration rate λ_1 for S_1 will be higher than the immigration rate λ_2 for S_2 . Similarly, the emigration rate μ_1 for S_1 will be lower than the emigration rate μ_2 for S_2 [Simon, D., 2008].

The emigration and immigration rates of each solution are used to probabilistically share information between habitats [Simon, D., 2008]. With probability P_{mod} , each solution is modified based on other solutions. If a given solution is selected to be modified, then its immigration rate λ is used to probabilistically decide whether or not to modify each suitability index variable in that solution. If a given SIV in a given solution S_i is selected to be modified, then the emigration rates μ of the other solutions is used to probabilistically decide which of the solutions should migrate a randomly selected SIV to solution S_i [Simon, D., 2008].

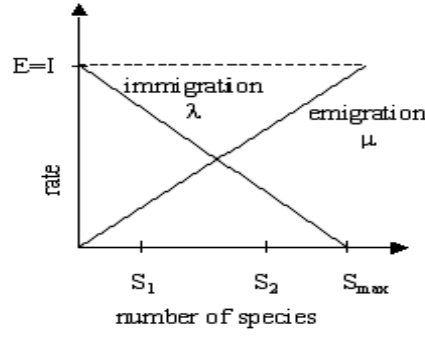


Figure 2.1: Illustration of two candidate solutions to some problem.

Cataclysmic events can drastically change the HSI of a natural habitat and can also cause a species count to differ from its equilibrium value. This is modeled in BBO as SIV mutation [Simon, D., 2008], and species count probabilities are used to determine mutation rates. Each population member has an associated probability, which indicates the likelihood that it was expected *a priori* to exist as a solution to the given problem. This mutation scheme tends to increase diversity among the population. The approach makes low HSI solutions likely to mutate, which gives them a chance of improving. The steady state value for the probability of the number of each species to exist is given as [Simon, D., 2008]:

$$P_k = \begin{cases} P_0 = \frac{1}{1 + \sum_{k=1}^n \frac{\lambda_0 \lambda_1 \cdots \lambda_{k-1}}{\mu_1 \mu_2 \cdots \mu_k}}, & k = 0 \\ P_k = \frac{\lambda_0 \lambda_1 \cdots \lambda_{k-1}}{\mu_1 \mu_2 \cdots \mu_k \left(1 + \sum_{k=1}^n \frac{\lambda_0 \lambda_1 \cdots \lambda_{k-1}}{\mu_1 \mu_2 \cdots \mu_k} \right)}, & 1 \leq k \leq n \end{cases} \quad (2.1)$$

The largest possible number of species that the habitat can support is n . It is necessary that $\mu_k \neq 0$ for all 'k' for this limiting probabilities to exist.

2.1.2. BBO Definitions and Algorithm

The definitions used in the BBO algorithm are reproduced below where R refers to the set of real numbers, Z refers to the set of integers, and Φ refers to the empty set:

- [1] A habitat $H \in SIV^m$ is a vector of m integers that represents a feasible solution to some problem [Simon, D., 2008].
- [2] A suitability index variable $SIV \in C$ is an integer that is allowed in a habitat. C (is subset of Z^q) is the set of all integers that are allowed in a habitat [Simon, D., 2008].
- [3] A Habitat Suitability Index $HSI: H \rightarrow R$ is a measure of the goodness of the solution that is represented by the habitat [Simon, D., 2008].
- [4] An ecosystem H^n is a group of n habitats. The size n of an ecosystem is constant [Simon, D., 2008].

- [5] Immigration rate λ (HSI): $R \rightarrow R$ is a monotonically non-increasing function of HSI. λ_i is proportional to the likelihood that SIVs from neighboring habitats will migrate into habitat H_i and for linear immigration and emigration model it can be given by the following formula [Simon, D., 2008]:

$$\lambda_k = I \left(1 - \frac{k}{n} \right) \quad (2.2)$$

- [6] Emigration rate μ (HSI): $R \rightarrow R$ is a monotonically non-decreasing function of HSI. μ_i is proportional to the likelihood that SIVs from habitat H_i will migrate into neighboring habitats and it can be given by the following formula [Simon, D., 2008]:

$$\mu_k = \frac{Ek}{n} \quad (2.3)$$

- [7] Habitat modification Ω (λ, μ): $H^n \rightarrow H$ is a probabilistic operator that adjusts habitat H based on the ecosystem H^n . The probability that H is modified is proportional to its immigration rate λ , and the probability that the source of the modification comes from H_j is proportional to the emigration rate μ_j [Simon, D., 2008]. The procedure is described by the following algorithm. Here, elitism can be implemented by setting $\lambda = 0$ for the p best habitats, where p is a user-selected elitism parameter.

Select H_i with probability $\propto \lambda_i$

If H_i is selected

For $j = 1$ to n

Select H_j with probability $\propto \mu_j$

If H_j is selected

Randomly select an SIV σ from H_j

Replace a random SIV in H_i with σ

End

End

End

- [8] *Mutation* M (λ, μ): $H \rightarrow H$ is a probabilistic operator that randomly modifies habitat SIVs based on the habitat's a priori probability of existence. A habitat's probability of existence is computed from λ and μ as shown in equation (2.1). For classic BBO, the mutation rate m is inversely proportional to the solution probability, which is defined by:

$$m = m_{\max} \left(1 - \frac{P}{P_{\max}} \right) \quad (2.4)$$

where $m_{\max}$ is a user-defined parameter. Mutation can be described as follows [Simon, D., 2008]:

Use λ_i and μ_i to compute the probability P_i
 Compute mutation probability m_i
 Select SIVs in H_i with probability based on m_i
 If H_i (SIV) is selected
 Replace H_i (SIV) with a randomly generated SIVs
 End

- [9] An ecosystem transition function $\Psi = (m, n, \lambda, \mu, \Omega, M): H^n \rightarrow H^n$ is a 6-tuple that modifies the ecosystem from one optimization iteration to the next. An ecosystem transition function can be written as follows [Simon, D., 2008].

$$\Psi = \lambda^n \circ \mu^n \circ \Omega^n \circ HSI^n \circ M^n \circ HSI^n \quad (2.5)$$

In other words, the ecosystem transition function begins by computing the immigration and emigration rates of each habitat. Then habitat modification is performed on each habitat, followed by an HSI recalculation. Finally mutation is performed, followed again by an HSI recalculation for each habitat.

- [10] A biogeography-based optimization algorithm BBO= (I, Ψ, T) is a 3-tuple that proposes a solution to an optimization problem. $I: \Phi \rightarrow \{H^n, HSI^n\}$ is a function that creates an initial ecosystem of habitats and computes each corresponding HSI. Ψ is the ecosystem transition function, and $T: H^n \rightarrow \{true, false\}$ is a termination criterion [Simon, D., 2008].

The basic BBO algorithm uses the following strategy:

$$I: \Phi \rightarrow \{H^n, HSI^n\}$$

While not T (where $T: H^n \rightarrow \{true, false\}$)

$$\Psi = \lambda^n \circ \mu^n \circ \Omega^n \circ HSI^n \circ M^n \circ HSI^n$$

End

The BBO algorithm [Simon, D., 2008] can now be described as follows:

- Initialize the BBO parameters. This means deriving a method of mapping problem solutions to SIVs and habitats, which is problem dependent. Also, the maximum species count S_{max} , the maximum migration rates E and I and an elitism parameter are initialized.
- Initialize a random set of habitats, each habitat corresponding to a potential solution to the given problem.
- For each habitat, map the HSI to the number of species S , the immigration rate λ , and the emigration rate μ .
- Probabilistically use immigration and emigration to modify each non-elite habitat as discussed above, and then re-compute each HSI.
- For each habitat, update the probability of its species count using (a). Then mutate each non-elite habitat based on its probability, and re-compute each HSI.

- f. Go to step (c) for the next iteration. This loop can be terminated after a predefined number of generations or after an acceptable problem solution has been found.

2.2. Biogeography Based Framework for Land Cover Feature Extraction

In this section, the details of the biogeography based land cover feature extraction process are given – the parameters and the definitions used, the assumptions made, the algorithm used and the classification [Panchal, V. K., et. al., 2009] which form the base to the behavioral analysis framework that is presented in the chapter.

2.2.1 Defining Parameters for the Algorithm

The section identifies some important parameters and their settings for the algorithm [Panchal, V. K., et. al., 2009]. For a better understanding, the definitions are discussed with reference to the satellite dataset of Alwar image for demonstration and pointers are included to the corresponding parameter settings for the second dataset of Patalganga region.

Assumption: Species S_i in this study refers to the image pixels and the image I , may be considered as a set of species.

$$I = \{S_i \mid \forall i \in [1 \dots (\text{Size of image } I)]\} \quad (2.6)$$

Definition 1: Eco-System: It is described as under [Panchal, V. K., et. al., 2009]:

- i. A *habitat* is a place where resides some species and that are likely to be migrated to other more suitable habitats. Let the total population of all kinds of species reside in N habitats, H_i .
- ii. The habitat H_i , is not of homogeneous composition of species. It contains different types of species.
- iii. A *Universal Habitat* Π is considered which hosts all the image species. Therefore,

$$\Pi = \bigcup_{i=1}^N H_i \quad (2.7)$$

- iv. *Feature Islands:* these are the habitat islands populated by the homogeneous species. These are nothing but the labeled dataset/training dataset generated by the expert. Five types of feature islands are considered here, namely: *water, vegetation, rocky, urban and barren*. For Patalganga dataset, only 3 feature islands that of vegetation, snow and rocky are considered.
- v. Therefore, the ecosystem H_6 is a group of 6 habitats (one *universal habitat* and five *feature islands*, Figure 2.2). In case of Patalganga, the ecosystem H_4 will be a group of 4 habitats.
- vi. The *eco-system* therefore is illustrated in Figure 2.2.

Definition 2: Suitability Index Variable (SIV): Each of the multispectral bands of image represents one Suitability Index Variable (SIV) of the habitat. Further, image in each band is a gray image; therefore, $SIV \in C$ is an integer and $C \subseteq [0, 255]$.

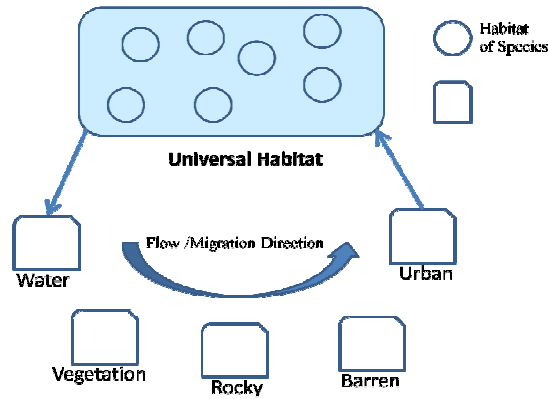


Figure 2.2: Eco-system – Initial state of Habitats

Definition 3: A habitat $H \in SIV^m$ where, m is $7 = \text{total image bands}$. For Patalganga image, $m = 4$ bands.

Definition 4: The image I is subjected to 20 simple partitions P_i . The image may be partitioned into any number as per the individual preference. For Patalganga region, the image is partitioned into $n = 15$ partitions.

$$I = \{P_1 \dots P_{20}\} \quad (2.8)$$

Each P_i is subjected to unsupervised clustering so that based on some criteria derived from the P_i statistics, new smaller clusters are generated. The rough sets based clustering [(Pawlak, Z., 1982), (Pawlak, Z., 1998), (Panchal, V. K., et. al., 2008)] is performed to P_i , and new, unsupervised, clusters are generated. These are now known as the new habitats H_i . The preference to the rough sets is given because of the following reasons [Ohm, A., et. al., 1997]:

- i. The concept of *equivalence class* [(Pawlak, Z., 1982), (Pawlak, Z., 1998), (Panchal, V. K., et. al., 2008)] automatically generates the clusters based on the criteria of indiscernibility.
- ii. This method does not need to know *a priori*, the *number* of clusters required.

For the indiscernibility criteria the attributes are the NIR and MIR bands. It has been observed that NIR and MIR bands contain a good amount of geospatial information [(Goel, L., et. al., 2010C), (Lillesand, T. M., et. al., 2007), (Panchal, V. K., et. al., 2008)]. The set of all equivalence class / elementary knowledge granules i.e. H_i are nothing but the partition of P_i .

$$P_i / R = \{[x]R : x \in P_i\}$$

$$\text{where the equivalence relation } R = \text{IND}(\{NIR, MIR\}) = \{(x, y) \in P_i\} \quad (2.9)$$

Therefore, H_i , may now be considered as *mixed species habitat* and therefore contains uncertainty [Ohm, A., et. al., 1997]. The removal of the uncertainty in the type of the species is carried out in this section.

$$\text{Let } J = |P_i / R|, \quad (2.10)$$

Then J gives the no. of knowledge granules generated from P_i . From the above discussion the following is inferred:

$$\Pi = \bigcup_i^{20} P_i \quad (2.11)$$

where, $P_i = \bigcup_j H_i$. For the sake of simplicity, HP is defined as a single unsupervised cluster of the P^{th} partition of the image I .

Definition 5: The original BBO algorithm [Simon, D., 2008] proposed the migration of SIV values from a high HSI habitat to a low HSI habitat. The shared features (SIV) remain in the high HSI solutions, while at the same time appearing as new features in the low HSI solutions. In this algorithm, rather than moving SIV, species altogether are moved from a universal habitat to feature islands. The species does not remain shared: it is removed from the universal habitat and migrated to feature habitat.

Definition 6: Habitat Suitability Index (HSI): The approach assumes the HSI value as the average magnitude of the absolute value of the chosen metric for each band of the image. This metric can be taken to be any measure for the individual bands such as the euclidean distance, deviation from mean metric, entropy, harmonic mean, geometric mean, arithmetic mean, etc. whichever validates the requirements of the chosen application. The stated application deals with relative values as the comparison measure and hence the HSI metric is defined as the one that will capture relatively similar values in different bands of the image representing the same set of pixels. Hence, the mathematical formulation for HSI metric termed as the ‘deviation from mean metric’ (DMM) is given by equation (2.12).

$$\text{DMM} = \left[\frac{\sum_{i=1}^n |h_i - \bar{h}|^p}{n} \right]^{\frac{1}{p}} \quad (2.12)$$

where h_i are the individual pixels’ DN values, $\bar{h}$ is the mean value, n is the no. of pixels in the respective band of the satellite image and p is the order which will depend upon the analysis of input data distribution.

Definition 7: Heuristic Function: The major steps that are followed are image clustering and heuristic method implementation. The 7-bands image is clustered into H_i , using rough set theory. The parameters of clustering are NIR and MIR bands. The resultant clusters or the habitats H_i , are species in universal habitat. These species are migrated to the suitable habitats i.e. feature islands. These habitats are actually the decisions, for the resultant different feature groups. The heuristic method decides which species are moved to which habitat using a fitness function.

As stated in definition 6 above, the HSI is taken as the average magnitude of the chosen metric for all of the multi-spectral bands of the image. (If the DMM is taken as the standard deviation (in case of $p = 2$), the arguments which suggest using statistical formulations for working with standard deviations do not hold good in the stated case, since these suggest finding the standard deviation of a resultant distribution which has no logical significance in the stated remote

sensing application. This is because, the different multi-spectral bands of the image have separate units and represent entirely different properties of the digital image, for example, the DEM band of the image represents the elevation (height) in m / km, the NIR and MIR bands represent the wavelength in μm / nm etc. and hence a combined distribution cannot be obtained. Thus, there will be no information contained in the parameter obtained from adding or averaging the individual band values and would result in loss of information because that would effectively represent the entire spectrum of individual bands of the image, dissipating the information contained in the individual parameters of the satellite image. It may also be noted that the average magnitude of standard deviation is not a property of pixel or image. It is used only for the basis of classification by comparing with the training set to which the same linear process is applied as done for the input image. Any higher order equation or process applied will distort the relation between the training set and the input image pixels for the purpose of comparison. When experimentally tested, the statistical formulations for finding standard deviation of a combined distribution fail to classify the input image accurately and result in a decrease in the classification efficiency (0.58 on Alwar dataset and 0.61 on Patalganga dataset) and a large no. of pixels remain unclassified until the end of the algorithm (almost 20% pixels remain unclassified) not allowing the algorithm to converge.

The above arguments aptly convince the reader about the inaccuracy and irrelevance of other statistical formulations for dealing with standard deviations in the stated application. Thus, after careful observations and analysis, it was found that the chosen HSI formulation aptly fits into the stated framework for any value of p .)

Hence, the HSI function as stated in definition 6 is used in this feature extraction application.

Thus, HSI_{F_i} of any of the feature island is defined as:

$$HSI_{F_i} = (|\omega_{1_i}| + |\omega_{2_i}| + |\omega_{3_i}| + |\omega_{4_i}| + \dots + |\omega_{m_i}|) / m \quad (2.13)$$

where $|\omega_{1_i}|, |\omega_{2_i}|, |\omega_{3_i}|, |\omega_{4_i}|, \dots, |\omega_{m_i}|$ are the absolute values of the HSI metric of the DN values of each of the m bands respectively of the i^{th} feature island.

Similarly, HSI_{C_j} of the species habitat C_j from H_i of the Universal Habitat Π is represented by:

$$HSI_{C_j} = (|\omega_{1_j}| + |\omega_{2_j}| + |\omega_{3_j}| + |\omega_{4_j}| + \dots + |\omega_{m_j}|) / m \quad (2.14)$$

Then the heuristic function Φ is defined as

$\begin{aligned} \Phi &= HSI_{C_j} - HSI_{F_i} \leq \delta \text{ Then } HSI_{C_j} \text{ may be thought of belonging to the class } HSI_{F_i}, \\ \Phi &= HSI_{C_j} - HSI_{F_i} \leq 0 \text{ Then } HSI_{C_j} \text{ is exactly the } HSI_{F_i} \text{ class,} \\ \Phi &= HSI_{C_j} - HSI_{F_i} \leq > \delta \text{ Then } HSI_{C_j} \text{ is of different type than } HSI_{F_i} \end{aligned}$

where, δ , is a threshold.

(2.15)

This function decides which value of mean of the chosen HSI metric has minimum difference from the original class. The recalculated HSI after the migration of species is compared with the original HSI of the habitat (that contained training pixels only). If the compared HSI is within the threshold i.e., it varies from -1 to +1 from the original HSI then it means that the habitat is suitable for that species and hence it can migrate to it. If the recalculated HSI is not within the threshold for any of the original habitat then the particular elementary class is migrated into the Universal habitat and considered as unclassified species because it contained a mixture of species. This unclassified class is again partitioned into elementary classes using rough set theory [Ohm, A., et. al., 1997]. The checking of all the habitats results in the completion of first generation and then the second generation begins for further refinement following the similar process. The maximum number of generations can correspond to the situation when no more unclassified class (species) is left in the Universal habitat.

Definition 8: Maximum immigration rate and Maximum emigration rate are same and equal to number of species in the habitat. The algorithm follows a linear curve ($E = I$). Number of species and thus the maximum immigration rate and maximum emigration rate can vary in each iteration. Maximum species count (S_{max}) and the maximum migration rates are relative quantities. That is, if they all change by the same percentage, then the behavior of BBO will not change [Simon, D., 2008].

Definition 9: Since mutation is not an essential feature of BBO [Simon, D., 2008], it is not required in the algorithm. Elitism, too, is an optional parameter [Simon, D., 2008], it has not been used in the algorithm.

2.2.2. Algorithm for biogeography based land cover feature extraction

The following biogeography based algorithm in figure 2.3 is used to extract land cover features from the satellite image [Panchal, V. K., et. al., 2009]. The input to the algorithm is the multi-spectral satellite image and the output is the extracted features from the image. The section also discusses some details of its implementation in the steps described below for which the input image is taken as the Alwar image and also mentions the corresponding settings for the Patalganga dataset. The detailed working of the above algorithm is described as follows [Panchal, V. K., et. al., 2009]:

- [1] A 7-band satellite image of Alwar region of size 472×546 is taken as an input image for biogeography based feature extraction. The image has 2,57,712 pixels and each pixel is of 25m by 25m spatial resolution. The image is divided into 'n' equal parts ($n = 20$ in this case however, this value can be varied) and each part corresponds to a class. Each pixel in a class has x and y coordinates along with the 7-band values of each pixel. For Patalganga

image, a 4-band image of size 508×744 is taken as input which has about 3,77,952 pixels divided into 15 partitions.

- [2] The input read data is further discretized by using semi-naïve discretization algorithm defined in ROSETTA [Ohm, A., et. al., 1997]. The no. of elementary classes produced depends upon the clustering of pixels into similarity classes (equivalence classes). The pixels which are similar (decided by the underlying discretization algorithm) are grouped into one elementary class. In the resulting data, DN value of each pixel is replaced by the discretized interval. Each row in the resulting data represents the x, y coordinates and the DN values in the red, green, NIR, MIR, RS1, RS2 and DEM bands respectively of each pixel in the image (for patalganga image, the bands are red, green, NIR and MIR bands). This discretized data is further partitioned on the basis of NIR and MIR band to obtain the final elementary classes. The band used for partitioning may affect the final results. In the case, the NIR and MIR bands are chosen for partitioning since the NIR and the MIR bands contain a good amount of geo-spatial information [(Goel, L., et. al., 2010C), (Lillesand, T. M., et. al., 2007), (Panchal, V. K., et. al., 2008)]. Each elementary class consists of the index of each pixel belonging to it defined in terms of pixel no. of the image containing 2,57,712 pixels. Hence, the pixel numbers for a pixel in a cluster will range from 1 to $(1/20 \times 257712)$ i.e. 1 to 12,886 for each of the 20 clusters. (For Patalganga image, these will range from 1 to $(1/15 \times 3,77,952)$ i.e. 1 to 25,197 for each of the 15 clusters). The pixel's DN values corresponding to these indexes are obtained before applying the biogeography based feature extraction algorithm. The elementary classes obtained by ROSETTA are put in the universal habitat and each elementary class is treated as a species [Ohm, A., et. al., 1997].

```

Input the multi-spectral satellite image.
Cluster the pixels of image randomly (using rough set theory) and consider each cluster as a species of  $H_u$ .
Take each of the land cover features as the feature habitats  $H_f$ .
Define HSI,  $S_{max}$ ,  $\lambda$ ,  $\mu$ .
Calculate HSI of each  $H_f$ 
  For each species in  $H_u$ 
    Select a species from  $H_u$  and migrate it to one of the  $H_f$ .
    Recalculate the HSI of the  $H_f$  after the migration of the species to it.
      If the recalculated HSI is within the threshold, then:
        Absorb the species in  $H_f$ .
      Else if any unconsidered  $H_f$  is left then:
        Migrate the species to that  $H_f$  and recalculate the HSI after migration.
        Else use rough set theory to discretize the species (as it contained mixed pixels) and make
        random clusters which are then considered as separate species.
    Add these new species to  $H_u$ .
  End If.
End For.

```

Figure 2.3: Biogeography based land cover feature extractor

The HSI metric of all the pixels in the elementary class is calculated on each band of the species habitat and also the feature habitats containing the training sets corresponding to each feature separately (which is a measure of HSI).

[3] Each member of the universal habitat is migrated to each of the feature habitat one by one.

The HSI of the habitat is recalculated after this migration and the recalculated HSI is compared with the original HSI of the habitat (that contained training pixels only). If the compared HSI is within the threshold of -1 to +1 from the original HSI, then the species get absorbed in the feature habitat. If the recalculated HSI is not within the threshold for any of the original habitat then the particular elementary class contained a mixture of species and it is further partitioned into elementary classes using Rosetta software. These classes are migrated into the Universal habitat and considered as unclassified species to be used for the next generation.

[4] The process completes when all the habitats have been checked which results in the completion of first generation and then the second generation starts for further refinement following the similar process. The maximum number of generations can correspond to the situation when no unclassified class (species) is left in the universal habitat.

2.3. Performance Analysis of Biogeography Based Feature Extractor

This section presents the proposed framework and the steps followed for the behavioral analysis of biogeography based feature extractor in order to draw useful conclusions regarding the applicability of BBO over different land cover features. The section describes the proposed methodology in detail.

2.3.1 Proposed Framework

The main aim is to identify the classification efficiency of the biogeography based feature extractor [Panchal, V. K., et. al., 2009] towards different natural terrain features so as to draw inference regarding the applicability of BBO for the extraction of given land cover features. This framework does not predict the exact efficiency of algorithm for each feature as the aim is not to calculate the classification accuracies from the determined factors, since that can be very well calculated from the error matrices but the aim is to compare the features on the basis of governing factors of BBO to get some apriori knowledge about the characteristic accuracy of BBO on each feature. This information will be helpful in identifying the features which can be better classified by BBO and thus help in building hybrid classifiers by applying BBO only on those features on which it shows very high classification efficiency and other algorithms on the rest of the features.

The section first establishes the factors which govern the performance of BBO for land cover feature extraction and then goes on to evaluate the values of each of the performance governing

factors for each land cover feature [(Goel, L., et. al., 2011A), (Goel, L., et. al., 2013C), (Panchal, V. K., et. al., 2011)]. For verification of the proposed relationship between the classification efficiency and the factors governing it, the classified images are taken from the biogeography based feature extractor results [Panchal, V. K., et. al., 2009] and the fact that the actual classification efficiencies that were provided by the extractor in the work [Panchal, V. K., et. al., 2009] follow the equation governing the performance of BBO that is derived in the analytical study is established. Figure 2.4 describes the validation of the performance governing framework of the biogeography based feature extractor.

2.3.2 Performance governing factors of BBO for land cover feature extraction

The work proposes that the classification efficiency of BBO for a given land cover feature is proportional to the degree of disorder of the DN values of the pixels comprising that land cover feature when viewed in any of the bands of the multi-spectral satellite image. More precisely, the classification efficiency of BBO on a terrain feature is inversely proportional to the entropy for that feature when viewed in any of the bands of the multi-spectral image

$$\eta \propto \frac{1}{E} \quad (2.16)$$

where ‘ η ’ is the classification efficiency of BBO for a given land cover feature and ‘E’ is the entropy of DN values of the pixels comprising the feature. Entropy ‘E’ is calculated as under:

$$\text{Entropy}(E) = - \sum_{i=1}^n P_i \log(P_i) \quad (2.17)$$

Where, ‘ P_i ’ is the probability of an input pixel falling in the i^{th} interval. This means that BBO is able to classify the homogeneous regions i.e. the regions with the lower entropy, more efficiently than the regions which show a greater degree of heterogeneity.

This means that when each P_i is subjected to rough sets based unsupervised clustering so that based on some similarity criteria derived from the P_i statistics, new smaller clusters called the equivalence classes are generated, the pixels representing the land cover features which exhibit a smaller degree of disorder in their DN values or which possess a smaller entropy value will

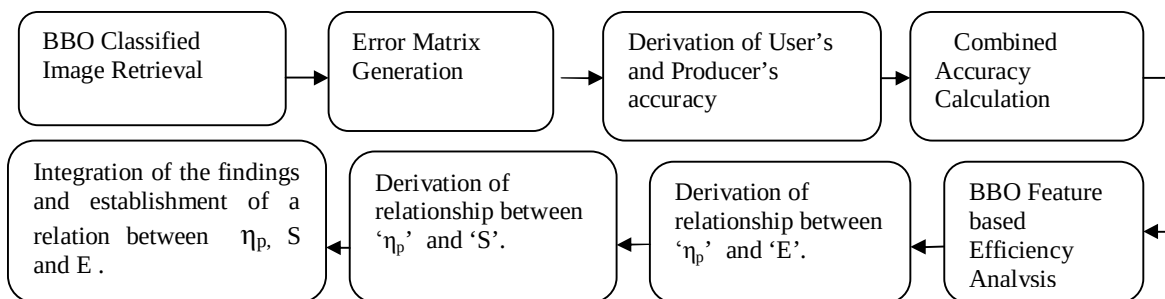


Figure 2.4: Proposed framework for the validation of the performance governing equation of biogeography based feature extractor.

lie approximately under the same similarity criteria which is derived from the P_i statistics. Hence, when discretized and partitioned, they will all group together into the same cluster or the equivalence class i.e. the entropy (E) is directly proportional to the percentage of mixed pixels in an equivalence class (m) i.e.

$$E \propto m \quad (2.18)$$

This means that smaller is the entropy of the feature 'j', lesser is the percentage of mixed pixels in the equivalence class which is classified as the feature 'j' and hence smaller will be the no. of clusters or the equivalence classes in which the pixels of the feature 'j' are spread thus leading to better unsupervised similarity clustering i.e.

Entropy (E) $\propto$ 1 / Quality of unsupervised similarity clustering (Q) i.e.

$$E \propto \frac{1}{Q} \quad (2.19)$$

Now, in BBO, for each i^{th} band where 'i' ranges from 1-7, the difference in the chosen HSI metric of the i^{th} band of the universal habitat and the i^{th} band of the feature habitat containing the expert generated training set of the image is calculated. If this difference is the minimum for the feature 'j' and also less than the pre-specified threshold value of, $-1 < t < +1$, then that particular equivalence class is classified as the feature 'j' else $j=1$ (unclassified). The process is repeated for each equivalence class until there is no equivalence class left in the universal habitat and the whole process is iterated till there is no unclassified equivalence class left. This means that the classification process of BBO is controlled by the difference in the HSI values (average magnitude of the chosen HSI metric for all the bands) of the rough sets generated equivalence classes and that of the expert generated training set of a particular land cover feature. It is to be noted that the classification accuracy of BBO for a particular feature depends upon the value of the minimum difference below threshold calculated for this feature. If this difference value is not below threshold, then the equivalence class is further broken down in further iterations until each of the subclasses thus generated have their minimum difference value below threshold and therefore can be classified into a particular land cover feature. In other words, the classification accuracy of BBO for a particular land cover feature not only depends upon the difference being minimum and below threshold but also it is inversely proportional to the value of this difference which was calculated as minimum and below threshold (for which the process may need to run iteratively until the threshold condition is satisfied) for classifying the equivalence class as belonging to this land cover feature. This is because the smaller is the value of this difference, the more is the degree of similarity between the equivalence class and the training set for the feature 'j' and hence the lesser are the chances of an erroneous classification leading to a greater degree of classification efficiency.

An input equivalence class EQ_i in the partition ' P_k ' is considered where ' i ' ranges from 1 to the no. of equivalence classes in the partition ' P_k ' say count (EQ_i) and ' k ' ranges from 1- no. of partitions in which the image is initially divided, say, count (P_k). The difference D_j between the HSI of EQ_i and the training set for each of the features ' j ' is calculated where j is in the range '1- n ' where ' n ' is the no. of land cover features that are to be extracted from the multi-spectral image. Then, if ' p ' be the feature for which the D_j is minimum and below threshold (may require iterative processing) where ' p ' is between 1 to ' n ' and EQ_i is classified as the feature ' p ', then the classification accuracy of BBO for the feature ' p ', μ_p , is inversely proportional to the value of the difference ' D_p ' in the HSI between the equivalence class and the training set for the feature ' p '.

$$\text{i.e. } \mu_p \propto \frac{1}{D_p}$$

$$\text{where, } D_p = \min_{j=1}^n D_j, \quad (2.20)$$

and D_j is calculated as explained in the previous section satisfying the first condition i.e. $D_j = \Phi = |HSI_{C_j} - HSI_{Fi}| \leq \delta$ (for definitions of ' i ' and ' j ' refer section 2.2.1).

Considering each EQ_i in every P_k ,

$$\mu_q \propto \frac{1}{\left[\sum_{k=1}^{\text{count}(P_k)} \sum_{u=1}^{\text{count}(EQ_i)} D_q \right]} \quad (2.21)$$

where ' q ' is the feature into which the input EQ_i is classified. Considering the equivalence classes EQ_u of the entire universal habitat that are classified as belonging to the feature ' p ', it can be said that the overall classification accuracy of BBO ' η ' on the feature ' p ' is inversely proportional to the average of the differences D_u in the HSI value between the equivalence classes EQ_u and the training set for the feature where u ranges from 1 to total no. of equivalence classes categorized into ' p ', say, count (EQ_p) i.e. for the feature ' p ',

$$\eta_p \propto \frac{1}{\frac{1}{u} \left[\sum_{u=1}^{\text{count}(EQ_p)} D_u \right]} \quad (2.22)$$

$$\text{where, } D_u = \left(\sum_{u=1}^{\text{count}(EQ_p)} \right) D_p$$

The above formulation of the classification accuracy of BBO is applicable on each of the land cover features. From equations (2.16) and (2.22),

$$\eta_p \propto \left[\frac{1}{\frac{1}{u} \left[\sum_{u=1}^{\text{count}(EQ_p)} D_u \right]} \right] \cdot \frac{1}{E} \quad (2.23)$$

Now, since right hand side of (2.22) represents the similarity criteria for BBO to classify a given equivalence class into the land cover feature ' p ', the above equation is simplified by replacing the right hand side of equation (2.23) by the similarity index, S ,

where,

$$S = \frac{1}{\frac{1}{u} \left[\sum_{u=1}^{\text{count}(\text{EQp})} D_u \right]} \quad (2.24)$$

Therefore, using (2.24), equation (2.23) is simplified to

$$\eta_p \propto \frac{S}{E} \quad (2.25)$$

From (2.25), it is concluded that the classification efficiency of BBO, η_p , on a given land cover feature 'p' can be assessed in terms of the similarity index 'S' and the entropy 'E'. It is to be noted that the classification accuracy, η_p , that has been referred to is the combined accuracy calculated by taking the product of the producer's and the user's accuracy of BBO on the feature 'p'. (A factor 'k' can be introduced to remove the proportionality relation however, the value of this constant will depend upon the dataset characteristics and it cannot be generalized keeping in view the limited datasets on which the relationship is validated. Thus, its analysis which includes finding out the exact relationship equation has been left for future work and thus, a constant has not been introduced here since the purpose of the analysis is just to point out the existence of some relationship between the above parameters which has been used for the development of hybrid models.)

2.3.3 Detailed Framework for studying BBO behavior on natural terrain features

For the purpose of analyzing the behavior of BBO, the training set of each land cover feature is analyzed and the entropy as well as the similarity index of each feature is calculated. Based upon these findings, conclusions are drawn regarding the performance of BBO on different features that are required to be extracted from the input image. A feature for which the calculated entropy is lowest (or similarity index is highest) will be best classified by BBO and vice-versa. The detailed framework of entropy and similarity index calculations is described below.

I. Classification efficiency in terms of 'Entropy (E)'

Here, the classification efficiency of BBO is assessed on different features by calculating the entropy the framework of which is presented in Figure 2.5. The reason why entropy is chosen as the parameter on which the performance of BBO is judged is that entropy is a good measure of the degree of disorder of the DN values of pixels comprising a feature. This means that the features which are homogeneous in their DN values distribution will be better classified by BBO than the others with a less homogeneous distribution. For the entropy calculation, the entropy of each classified feature is calculated in any of the multiple bands of the multi-spectral satellite image. For the purpose, each of the land cover features classified pixels are copied in an excel sheet and then the DN values in the band chosen are divided into 'n' intervals for each

land cover feature. Next, the no. of pixels of a particular land cover feature falling in each of the ‘n’ ranges obtained above are calculated and hence, the entropy of the particular land cover feature is calculated by using the following formula:

$$\text{Entropy}(E_i) = - \sum_{i=1}^n P_i \log (P_i) \quad (2.26)$$

since the DN range corresponding to the band chosen is divided into ‘n’ intervals and

$$P_i = \frac{n_i}{N} \quad (2.27)$$

where n_i is the no. of pixels in the i^{th} interval and N is the total no. of pixels classified into that particular land cover feature. The above procedure is followed for each land cover feature. The procedure is the same for the second band chosen for verification of the results. From the entropy calculations made above, an assessment of the classification efficiency for each feature is obtained since BBO will show a greater degree of classification efficiency on the land cover features with smaller entropy. Hence, the land cover features are arranged according to the performance of BBO on each of them.

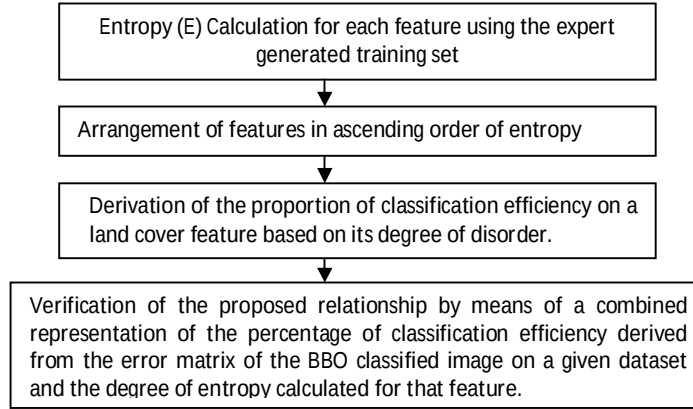


Figure 2.5: Detailed framework for the derivation of relationship between η_p and E .

II. Classification efficiency in terms of ‘Similarity Index (S)’

Next, the classification efficiency of BBO on different features is assessed by calculating the similarity index the framework of which is presented in figure 2.6. For the purpose, a table is constructed which calculates the HSI value of the training set for each feature in each of the bands of the multi-spectral satellite image. Next, the equivalence classes EQ_u of the entire universal habitat that are classified are considered as belonging to the feature ‘p’ and a table is constructed which calculates the average of the differences D_u in the HSI value of the equivalence classes EQ_u and the training set for the feature ‘p’ where ‘u’ ranges from 1 to total no. of equivalence classes categorized into feature ‘p’, say, count (EQ_p). The above procedure is followed for each of the land cover features.

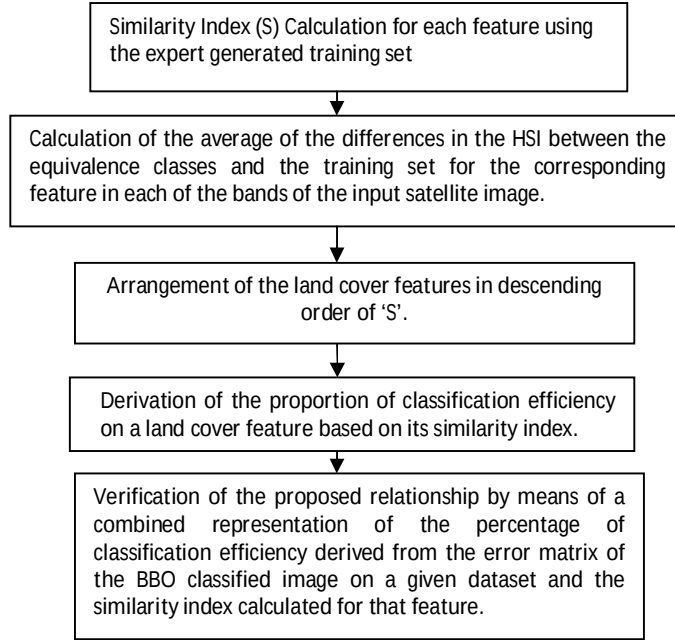


Figure 2.6: Detailed framework for the derivation of relationship between η_p and S .

From the index calculations made above, an assessment of the classification efficiency for each feature in each band is obtained by arranging the land cover features according to the performance of BBO on each of them. A similar procedure can be followed on the second dataset also for verification of the results. Hence, the validity of the proportionality relationship in equation (2.25) is proved.

The proposed relationship of the classification efficiency of BBO with the entropy and similarity index can be verified by using the error matrix of classification results of BBO that has been applied to some dataset. From the error matrix obtained, the accuracy measures from the user's and the producer's perspective is obtained and hence a combined table is derived which represents the overall efficiency of BBO for each land cover feature precisely and hence a combined representation of the percentage of classification efficiency and the degree of entropy and similarity index is obtained for each land cover feature in the chosen band arranged in ascending order of entropy (or descending order of similarity index) to reflect the fact that BBO shows a greater degree of classification efficiency on the land cover features with smaller degree of disorder and higher degree of similarity index.

2.4. Implementation, Results and Analysis

This section presents an analytical study of the BBO behavior based on the corresponding classification results in order to study the effect of entropy or the degree of disorder of the DN values of the pixels comprising a land cover feature as well as the similarity index with the classification efficiency of BBO on that feature. For the purpose, two different datasets of

Alwar and Patalganga are studied. Using the derived relation between the classification efficiency of BBO and its governing factors, the entropy and similarity index of each feature is calculated and thereby the classification efficiency of BBO towards each of the features which are required to be extracted from the given dataset is estimated. For verification, the classification results of BBO on the given dataset from [Panchal, V. K., et. al., 2009, (Panchal, V. K., et. al., 2011)] are used and hence the relationship that was established in equation (2.25) is validated.

2.4.1. Case study: 7-band resourcesat satellite image of size 472×576 pixels of Alwar Region in Rajasthan

This section discusses the results that are obtained on the Alwar image for studying the BBO behavior which establishes the relationship between the classification efficiency of BBO η_p and the similarity index 'S' and entropy 'E'.

A. Classification results

This section presents the feature extraction results of biogeography based feature extractor on the dataset of Alwar. The dataset, its characteristics and feature extraction results are presented. A multi-spectral, multi resolution and multi sensor image of Alwar area in Rajasthan with dimensions 472×576 is used. The satellite image for seven different bands is taken. The bands are Red, Green, NIR, MIR, RS1, RS2 and DEM. The Red, Green, NIR, and MIR band images are taken from LISS - III, sensor of Resourcesat, an Indian remote sensing satellite. RS1 and RS2 are the images from Canadian satellite Radarsat. Digital elevation model is derived by using images from RS1 and RS2. The 7-band satellite image of Alwar area in Rajasthan is shown in figure 2.7 is taken. Classified image of Alwar has been shown in figure 2.8 with a kappa coefficient of 0.6715 [Panchal, V. K., et. al., 2009, (Panchal, V. K., et. al., 2011)]. The color codes are red for urban, yellow for rocky, blue for water, green for vegetation and black for barren area.

B. Behavioral study

Once the classified image from BBO for the Alwar region is retrieved, the error matrix is

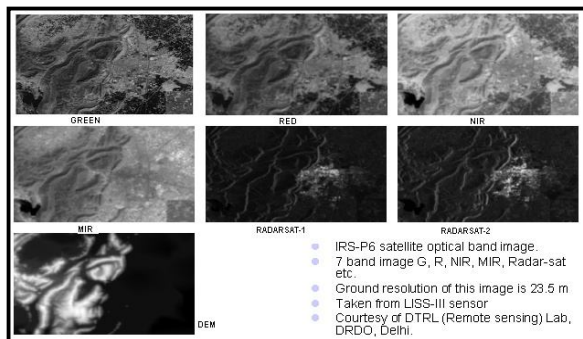


Figure 2.7: 7-band image of Alwar

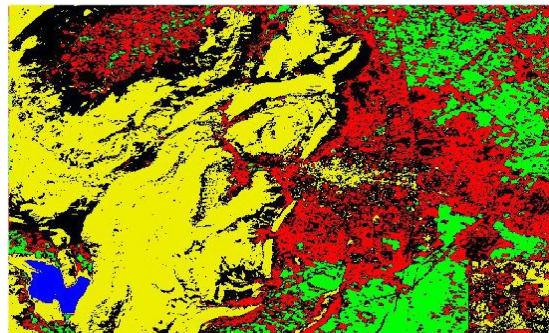


Figure 2.8: Biogeography based classification of Alwar image (with kappa coefficient = 0.6715)

obtained which is a representation of the classification accuracy of BBO over different land cover features in terms of the kappa coefficient which measures the proportion of agreement after chance agreements have been removed from considerations and hence removes the influence of noises in the classification efficiency calculation process. From the error matrix given in Table 2.1, the accuracy measures from the producer's and the user's perspective are obtained (Tables 2.2 and 2.3) and hence a combined table is derived (Table 2.4) which represents the overall efficiency of BBO for each land cover feature precisely. From the Table 2.4, it is observed that BBO displays a wide range of efficiencies in classifying different land cover features. The table reflects that the highest efficiency is observed on water pixels where BBO displays 100% efficiency, the second highest being the vegetation pixels with 83.2% efficiency, then the rocky (with 78.1% efficiency) and finally the barren (with 37.7% efficiency) and urban features (with 34.6% efficiency) respectively.

Table 2.1: Error Matrix for the Satellite image of Alwar region when BBO is applied

	Vegetation	Urban	Rocky	Water	Barren	Total
Vegetation	127	1	1	0	0	129
Urban	1	92	0	0	39	129
Rocky	1	15	175	0	3	194
Water	0	0	0	70	0	70
Barren	21	81	26	0	128	256
Total	150	189	202	70	170	781

Table 2.2: Producer's accuracy

Feature	Accuracy Calculation	Producer's Accuracy
Vegetation	127/150	84.6%
Urban	92/189	48.6%
Rocky	175/202	86.6%
Water	70/70	100%
Barren	128/170	75.3%

Table 2.3: User's accuracy

Feature	Accuracy Calculation	User's Accuracy
Vegetation	127/129	98.4%
Urban	92/129	71.3%
Rocky	175/194	90.2%
Water	70/70	100%
Barren	128/256	50%

Table 2.4: Combined accuracy of BBO for each land cover feature

Feature	Combined accuracy= (Producer's accuracy × User's accuracy)	Percentage accuracy
Vegetation	84.6 × 98.4	83.2%
Urban	48.6 × 71.3	34.6%
Rocky	86.6 × 90.2	78.1%
Water	100 × 100	100%
Barren	75.3 × 50	37.7%

I. Classification efficiency in terms of 'Entropy (E)'

Next, the entropy of each classified feature is calculated in any of the seven bands of the Alwar image. However, the Radarsat band is chosen since this band is the best for viewing the edges of a land cover feature and displays the variations in the reflectance values (DN ranges) more precisely. Also, in order to verify the results, the entropy is calculated in both the Radar-sat bands namely Radarsat-1 and Radarsat-2 bands respectively as shown in Table 2.5. For the

dataset of Alwar, the entropy values are calculated deciding upon a uniform interval count of 10 for all features after testing with smaller / greater interval counts both in order to maintain the tradeoff between the preciseness in entropy provided and the computational complexity of the entropy calculation module for the given dataset. Tables 2.6 and 2.7 below represent the calculation of entropy for each of the land cover features for both the Radarsat-2 and Radarsat-1 bands respectively. The histogram plot of the frequency of the pixels falling in each of the ‘n’ ranges for both the Radarsat-2 and Radarsat-1 bands is as shown in figures 2.9 and 2.10. Table 2.8 presents the BBO classification efficiency in both the RS-1 and the RS-2 bands in a combined form. From Table 2.8, it can be observed that the entropy of the water feature is the lowest in both the RS-1 and the RS-2 bands. Also, it can be seen that the entropy of the urban region is the highest among all the five land cover features in both the RS-1 and the RS-2 bands. From the table, the land cover features can be arranged in ascending order of their entropy values as water, vegetation, rocky, barren and urban features respectively. Also, it is found that BBO is able to classify the water feature with the maximum efficiency i.e. 100%. It is concluded that BBO is able to classify the feature which has the lowest entropy with maximum efficiency. The above finding is also reflected from figure 2.11 wherein it can be observed that the classification efficiency of BBO depends upon the entropy of the land cover feature.

Table 2.5: Entropy in RS-1 and RS-2 bands

Feature Entropy / Band in which the entropy is calculated	Radarsat-1 Band	Radarsat-2 Band
Vegetation Entropy	0.3185	0.3587
Urban Entropy	0.5004	0.6124
Rocky Entropy	0.3837	0.4799
Water Entropy	0.0742	0.1052
Barren Entropy	0.3923	0.5162

Table 2.6: Entropy calculation in Radarsat-2 band

	Vegetation (N=36197)		Urban (N=80653)		Rocky (N=87971)		Water (N=2544)		Barren (N=50347)	
K	N _i	$-\sum p_i \log p_i$	N _i	$-\sum p_i \log p_i$	N _i	$-\sum p_i \log p_i$	N _i	$-\sum p_i \log p_i$	N _i	$-\sum p_i \log p_i$
1	14506	0.1591	25738	0.1583	35798	0.1589	2419	0.0208	17030	0.1592
2	20444	0.2993	36937	0.3136	40544	0.3139	90	0.0721	25509	0.3088
3	1058	0.3441	8291	0.4152	8281	0.4105	18	0.0874	4652	0.4044
4	121	0.3524	3371	0.4728	2270	0.4515	7	0.0944	1580	0.4516
5	27	0.3547	2102	0.5141	736	0.4689	2	0.0968	798	0.4801
6	16	0.3562	1254	0.5422	223	0.4755	3	0.1003	388	0.4964
7	9	0.3571	910	0.5642	82	0.4783	0	0.1003	220	0.5067
8	7	0.3578	681	0.5817	24	0.4793	2	0.1027	107	0.5124
9	4	0.3583	569	0.5929	9	0.4797	2	0.1052	41	0.5149
10	4	0.3587	587	0.6124	3	0.4799	0	0.1052	19	0.5162

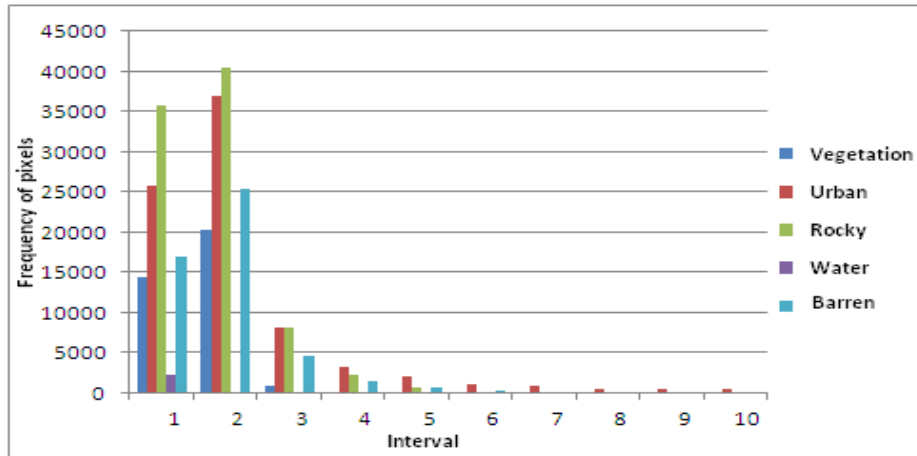


Figure 2.9: Histogram of the frequency of pixels in each interval for the land cover features from which the entropy corresponding to each feature is calculated in Radarsat-2 band.

Table 2.7: Entropy calculation in Radarsat-1 band

	Vegetation (N=36197)		Urban (N=80653)		Rocky (N=87971)		Water (N=2544)		Barren (N=50347)	
K	N_i	$-\sum p_i \log p_i$	N_i	$-\sum p_i \log p_i$	N_i	$-\sum p_i \log p_i$	N_i	$-\sum p_i \log p_i$	N_i	$-\sum p_i \log p_i$
1	22517	0.1282	47972	0.1342	59501	0.0970	2461	0.0000	34007	0.1151
2	13174	0.2880	21077	0.2865	21272	0.1149	64	0.0139	11924	0.2633
3	393	0.3093	5531	0.3663	4953	0.2639	8	0.0542	2921	0.3350
4	73	0.3148	2583	0.4142	1606	0.3343	2	0.0620	1008	0.3690
5	25	0.3170	1434	0.4453	496	0.3660	4	0.0645	334	0.3834
6	4	0.3174	798	0.4651	107	0.3787	1	0.0689	92	0.3884
7	5	0.3179	528	0.4794	30	0.3823	1	0.0702	39	0.3909
8	2	0.3182	298	0.4884	4	0.3834	1	0.0716	8	0.3915
9	1	0.3183	238	0.4959	1	0.3836	1	0.0729	6	0.3919
10	2	0.3185	132	0.5004	0	0.3837	0	0.0742	4	0.3923

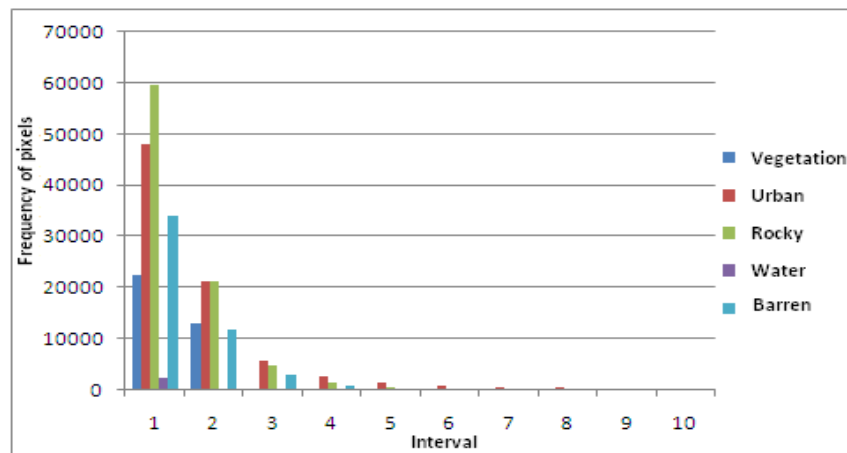


Figure 2.10: Histogram of the frequency of pixels in each interval for the land cover features from which the entropy corresponding to each feature is calculated in Radarsat-1 band.

Table 2.8: BBO Classification efficiency v/s Entropy in RS1 and RS2 bands for each land cover feature

Land cover feature	BBO Classification efficiency	Entropy in Radarsat-1 Band	Entropy in Radarsat-2 Band
Water	100%	0.0742	0.1052
Vegetation	83.2%	0.3185	0.3587
Rocky	78.1%	0.3837	0.4799
Barren	37.7%	0.3923	0.5162
Urban	34.6%	0.5004	0.6124

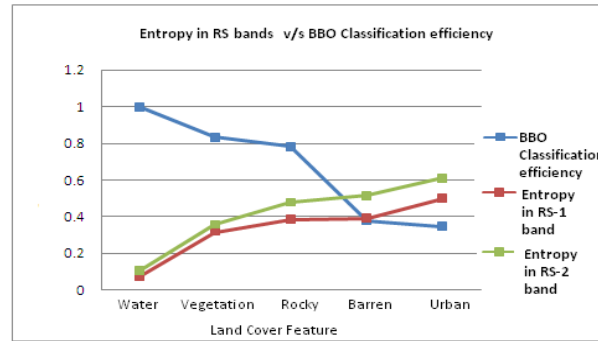


Figure 2.11: Combined graphical representation of the BBO classification efficiency v/s the entropy in RS-1 and RS-2 bands for each of the land cover features.

II. Classification efficiency in terms of 'Similarity Index (S)'

Next, it is analyzed why BBO behaves differently on land cover features with different similarity indices with the input equivalence class. For the purpose, a table is constructed which calculates the HSI values of the training set in each of the 7 bands of the 472×546 pixels multi-spectral satellite image. The HSI is found out for each of the land cover features as shown in Table 2.9.

Next, the rough set generated equivalence classes EQ_u of the entire universal habitat that are classified as belonging to the feature 'p' are considered and a table is constructed which calculates the average of the differences D_u in the HSI values between the equivalence classes EQ_u and the training set for the feature 'p' where u ranges from 1 to total no. of equivalence classes categorized into feature 'p', say, count (EQ_p). The above procedure is followed for each of the land cover features and hence Table 2.10 is obtained. Next, a table is constructed representing the calculation of similarity index 'S' for each of the land cover features as shown in Table 2.11. From the table, the land cover features are arranged in descending order of their

Table 2.9: Table of HSI values of the training set for each of the five land cover features in the Alwar image.

Land cover feature	HSI value (average magnitude of absolute value of chosen HSI metric 'deviation from mean metric' for the 7 bands) of the training set
Water	9.1
Vegetation	10.7
Rocky	14.2
Barren	18.6
Urban	34.1

similarity indices ‘S’. Comparing Tables 2.4 and 2.11 of combined classification accuracy of BBO and the similarity indices ‘S’ for each of the land cover features respectively, it is concluded that BBO is able to classify the land cover feature ‘water’ which has the highest similarity index 3.57 with the maximum efficiency i.e. 100%. The classification efficiency of BBO is the lowest for urban region i.e. 34.6% which also has the lowest similarity index 1.15.

Table 2.10: Average of differences D_u for each land cover feature for the Alwar image

Land cover feature	$\frac{1}{u} \left[\sum_{u=1}^{count(EQp)} D_u \right]$
Water	0.1026
Vegetation	0.3937
Rocky	0.4118
Barren	0.7459
Urban	0.8712

Table 2.11: Calculation of Similarity index ‘S’ from D_u for each feature

Land cover feature	$\frac{1}{u} \left[\sum_{u=1}^{count(EQp)} D_u \right]$	Similarity Index (S)= $\frac{1}{\frac{1}{u} \left[\sum_{u=1}^{count(EQp)} D_u \right]}$
Water	0.2805	3.57
Vegetation	0.3937	2.55
Rocky	0.4118	2.43
Barren	0.7459	1.34
Urban	0.8712	1.15

The similarity indices 2.55, 2.43 and 1.34 of the remaining land cover features of vegetation, rocky and barren features respectively are directly proportional to their respective classification efficiencies of 83.2%, 78% and 37.7% respectively. Hence, the greater is the similarity index ‘S’ of the land cover feature, the greater is the classification efficiency of BBO for that land cover feature.

Next, another table is constructed which is the combined representation of the percentage of classification efficiency and the normalized similarity index ‘ S_N ’ for each land cover feature arranged in ascending order to reflect the fact that BBO shows a greater degree of classification efficiency on the land cover features with greater degree of normalized similarity index ‘ S_N ’ as shown in Table 2.12. The above information is also represented through a graph plot in order to understand clearly the dependency relationships as shown in figure 2.12.

III. Classification efficiency in terms of ‘S’ and ‘E’

Finally, the findings are integrated and a relation is established between η_p , S and E by constructing a combined table representing the classification efficiency of BBO, the average entropy in the RS-1 and RS-2 bands chosen and the similarity index ‘S’ for each of the five land cover features of water, vegetation, urban, rocky and barren features as shown in Table

Table 2.12: Combined representation of η_p , S and S_N

Land cover feature	Classification efficiency of BBO in % (η_p)	Similarity Index (S)= $\frac{1}{\frac{1}{u} [\sum_{u=1}^{count(EQp)} Du]}$	Normalized Similarity Index (S_N)
Water	100%	3.57	1.00
Vegetation	83.2%	2.55	0.71
Rocky	78%	2.43	0.68
Barren	37.7%	1.34	0.37
Urban	34.6%	1.15	0.32

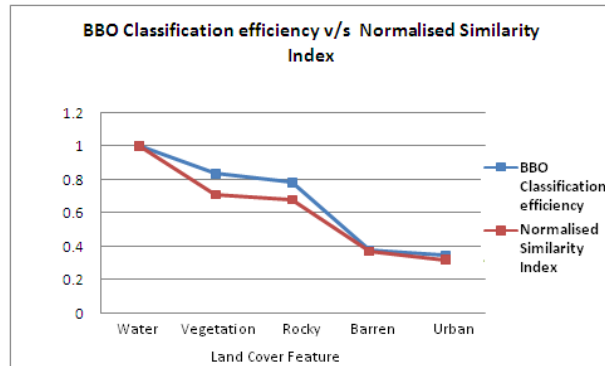


Figure 2.12: BBO Classification efficiency v/s Normalized Similarity Index for each land cover feature 2.13 and presented graphically by figure 2.13. From the figure, it can be observed that the classification efficiency of BBO ' η_p ', for a given land cover feature ' p ' is directly proportional to the similarity index ' S ' divided by the average entropy ' E ' for the feature ' p '. Hence the validity of equation (2.25) has been proved.

Table 2.13: Combined table of η_p , S and E for each land cover feature

Land cover feature	Classification efficiency of BBO in % (η_p)	Similarity Index (S)= $\frac{1}{\frac{1}{u} [\sum_{u=1}^{count(EQp)} Du]}$	Normalized Similarity Index (S_N)	Average of Entropy in RS-1 & RS-2 bands (E)	Normalized Entropy (E_N)	S_N/E_N
Water	100%	3.57	1.00	0.0897	0.159	6.289
Vegetation	83.2%	2.55	0.71	0.3386	0.6086	1.1666
Rocky	78%	2.43	0.68	0.4318	0.7761	0.8936
Barren	37.7%	1.34	0.37	0.4543	0.8165	0.4532
Urban	34.6%	1.15	0.32	0.5564	1	0.32

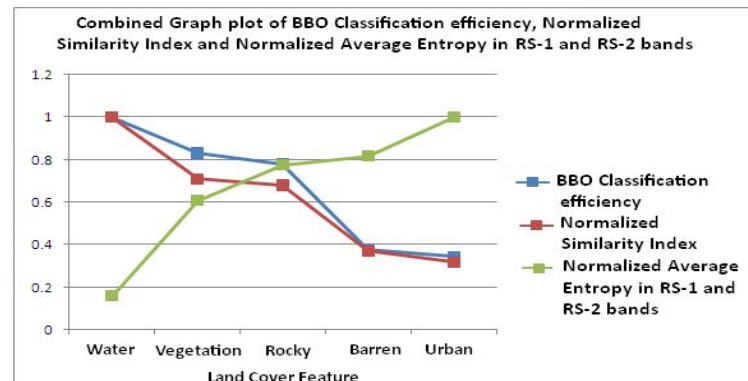


Figure 2.13: Combined graph plot of BBO Classification efficiency, Normalized Similarity Index and Normalized Average Entropy in RS-1 and RS-2 bands

2.4.2. Case study: 4-band Landsat-7 satellite image of size 508 × 744 pixels of Patalganga region in Shivaliks

This section discusses the results obtained on the Patalganga image for studying the BBO behavior.

A. Classification results

For Patalganga, area in Shivalik mountainous ranges is undertaken. 4 band images of Patalganga taken from Landsat-7 satellite with dimensions 508 X 744 have been used as shown in figure 2.14. The bands are Red, Green, NIR and MIR.

The classified image of Patalganga has been shown in figure 2.15 with kappa coefficient .72488 [(Panchal, V. K., et. al., 2009), (Panchal, V. K., et. al., 2011)]. The color codes are red for snow, yellow for rocky and green for vegetation area.

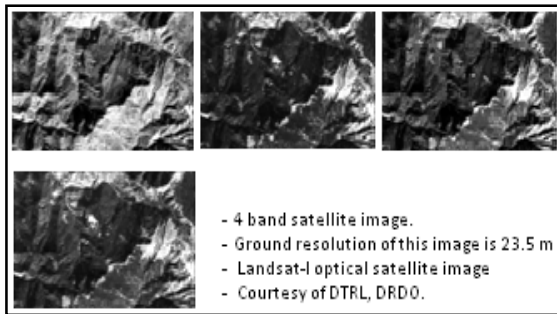


Figure 2.14: 4-Band Patalganga image

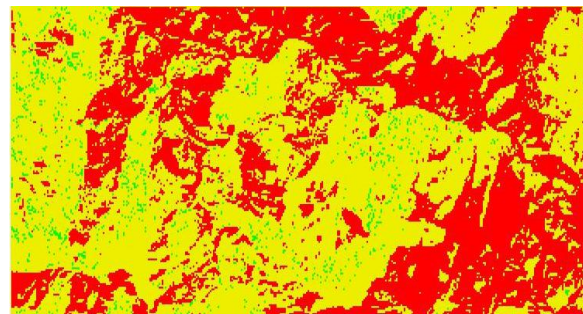


Figure 2.15: Biogeography Based Classification of Patalganga image (with kappa coefficient=.72488)

B. Behavioral Study

The error matrix is given in Table 2.14.

Table 2.14: Error Matrix for the Satellite image of Patalganga region when BBO is applied

	Vegetation	Snow	Rocky	Total
Vegetation	89	7	7	103
Snow	48	181	4	224
Rocky	24	5	189	227
Total	161	200	200	561

From the error matrix given in Table 2.14, the accuracy measures from the producer's and the user's perspective are obtained (Tables 2.15 and 2.16) and hence the combined table is derived (Table 2.17) which represents the overall efficiency of BBO for each land cover feature precisely.

Table 2.15: Producer's accuracy

Feature	Accuracy Calculation	Producer's Accuracy
Vegetation	89/161	55.3%
Snow	181/200	90.5%
Rocky	189/200	94.5%

Table 2.16: User's accuracy

Feature	Accuracy Calculation	User's Accuracy
Vegetation	89/103	86.4%
Snow	181/224	80.8%
Rocky	189/227	83.3%

Table 2.17: Combined accuracy of BBO for each land cover feature

Feature	Combined accuracy= (Producer's accuracy × User's accuracy)	Percentage accuracy
Vegetation	55.3 × 86.4	47.7%
Snow	90.5 × 80.8	73.1%
Rocky	94.5 × 83.3	78.7%

I. Classification efficiency in terms of 'Entropy (E)'

Next, the entropy of each classified feature is calculated in any of the four bands of the Patalganga image. For the illustration, the NIR and MIR bands are chosen first to observe and second to conclude and verify the results. Table 2.18 summarizes the entropy of each land cover feature for the NIR and MIR bands in a single table. Tables 2.19 and 2.20 below represent the calculation of entropy for each of the land cover features for both the NIR and MIR bands respectively. The histogram plot of the frequency of the pixels falling in each of the 'n' ranges for both the NIR and MIR bands is as shown in figures 2.16 and 2.17. Table 2.21 is a combined representation of the percentage of classification efficiency and the degree of entropy for each land cover feature in NIR band arranged in ascending order. For verification, the entropies were also calculated in NIR band and the results were found to be the same. From the Table 2.21, it can be observed that the entropy of the rocky feature is the lowest in both the NIR and the MIR bands. Also, it can be seen that the entropy of the vegetation region is the highest among all the three land cover features that were classified by BBO in both the NIR and the MIR bands. From the table, the land cover features are arranged in ascending order of their entropy values as rocky, snow and vegetation features respectively. Also, it is found that BBO is able to classify the rocky feature with the maximum efficiency i.e. 78.7%. Figure 2.18 is a combined graph plot of the BBO classification efficiency v/s the entropy in NIR and MIR bands for each of the land cover features.

Table 2.18: Feature Entropy in NIR and MIR bands

Feature Entropy / Band in which the entropy is calculated	NIR	MIR
Vegetation Entropy	0.8929	0.8210
Snow Entropy	0.7801	0.6922
Rocky Entropy	0.6199	0.5988

Table 2.19: Entropy calculation in NIR band

k	Vegetation (N=659)		Snow (N=2228)		Rocky (N=1927)	
	N_i	$-\sum p_i \log p_i$	N_i	$-\sum p_i \log p_i$	N_i	$-\sum p_i \log p_i$
1	40	0.0739	255	0.1077	36	0.0323
2	62	0.1704	523	0.2555	223	0.1407
3	68	0.2722	613	0.4097	365	0.2775
4	140	0.4151	394	0.5427	0	0.2775
5	170	0.5669	251	0.6496	415	0.4212
6	58	0.6598	108	0.7133	219	0.5285
7	30	0.7209	53	0.7519	0	0.5285
8	44	0.7994	19	0.7696	64	0.5776
9	32	0.8632	10	0.7801	52	0.6199
10	11	0.8929	0	0.7801	0	0.6199

Table 2.20: Entropy calculation in MIR band

k	Vegetation (N=659)		Snow (N=2228)		Rocky (N=1927)	
	N_i	$-\sum p_i \log p_i$	N_i	$-\sum p_i \log p_i$	N_i	$-\sum p_i \log p_i$
1	12	0.0317	35	0.0354	4	0.0056
2	87	0.1478	28	0.0637	7	0.0144
3	133	0.2880	18	0.0876	116	0.0879
4	130	0.4271	95	0.1045	391	0.2285
5	145	0.5718	329	0.1629	0	0.2285
6	68	0.6736	748	0.2856	437	0.3746
7	60	0.7683	772	0.4447	245	0.4885
8	16	0.8075	136	0.6042	94	0.5525
9	4	0.8210	14	0.6783	59	0.5988
10	0	0.8210	4	0.6922	0	0.5988

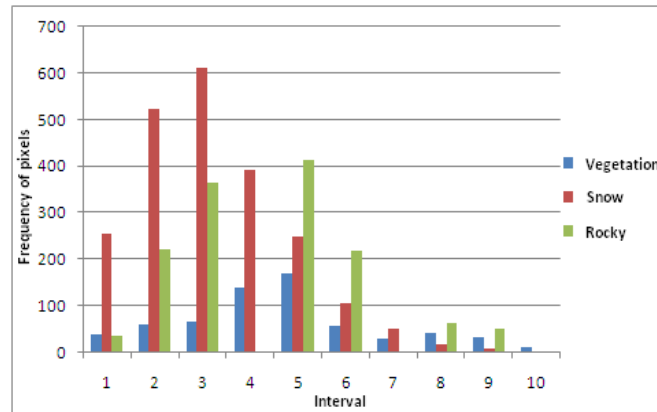


Figure 2.16: Histogram of the frequency of pixels in each interval for the land cover features from which the entropy corresponding to each feature is calculated in NIR band.

Table 2.21: BBO classification efficiency v/s the entropy in NIR and MIR bands for each of the land cover features.

Land cover feature	BBO Classification efficiency	Entropy in NIR Band	Entropy in MIR Band
Rocky	78.7%	0.6199	0.5988
Snow	73.1%	0.7801	0.6922
Vegetation	47.7%	0.8929	0.8210

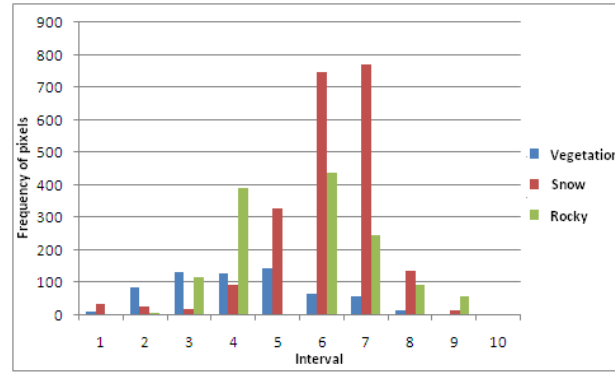


Figure 2.17: Histogram of the frequency of pixels in each interval for the land cover features from which the entropy corresponding to each feature is calculated in MIR band.

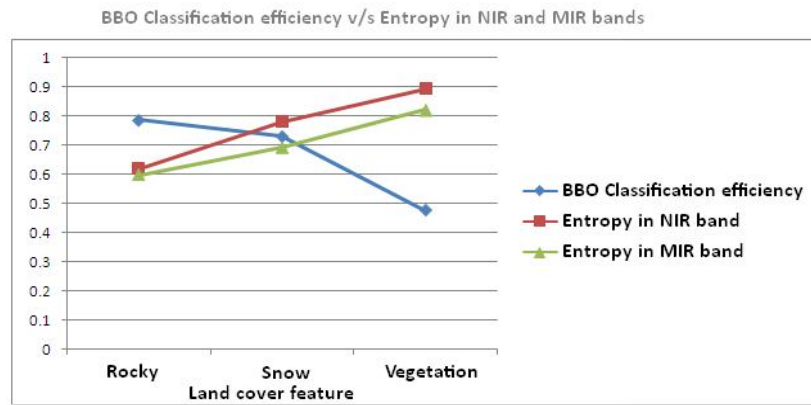


Figure 2.18: Combined graphical representation of the BBO classification efficiency v/s the entropy in NIR and MIR bands for each of the land cover features.

II. Classification efficiency in terms of 'Similarity Index (S)'

Next, BBO behavior is analyzed on different land cover features of the Patalganga image with different similarity indices. For the purpose, a table is constructed which calculates the HSI values of the training set in each of the 4 bands of the multi-spectral satellite image (Table 2.22) and hence calculates the average of differences D_u for each land cover feature for the Patalganga image as represented by Table 2.23. From the above tables, the similarity index is calculated for each feature (Table 2.24). Next, a table of the combined representation of the percentage of classification efficiency and the normalized similarity index ' S_N ' is constructed for each land cover feature arranged in ascending order as shown in Table 2.25.

Table 2.22: Table of HSI values of the training set for each of the three land cover features in the Patalganga image.

Land cover feature	HSI values (average magnitude of absolute values of the chosen HSI metric 'deviation from mean metric' for the 4 bands) of the training set
Rocky	12.48
Snow	11.06
Vegetation	0.882

Table 2.24: Calculation of Similarity index ‘S’ from D_u for each feature

Land cover feature	$\frac{1}{u} \left[\sum_{u=1}^{count(EQp)} D_u \right]$
Rocky	0.4700
Snow	0.4950
Vegetation	0.7180

Table 2.23: Average of differences D_u for each land cover feature for the Patalganga image

Land cover feature	$\frac{1}{u} \left[\sum_{u=1}^{count(EQp)} D_u \right]$	Similarity Index (S)= $\frac{1}{\frac{1}{u} \left[\sum_{u=1}^{count(EQp)} D_u \right]}$
Rocky	0.4700	2.13
Snow	0.4950	2.02
Vegetation	0.7180	1.39

From the table, it is concluded that BBO is able to classify the land cover feature ‘rocky’ which has the highest similarity index 2.13 with the maximum efficiency i.e. 78.7% in Patalganga. The classification efficiency of BBO is the lowest for vegetation region i.e. 47.7% which also has the lowest similarity index 1.39. The similarity index 2.02 of the remaining land cover feature ‘snow’ is also directly proportional to its respective classification efficiency of 73.1%. Hence, the greater is the similarity index ‘S’ of the land cover feature, the greater is the classification efficiency of BBO for that land cover feature. Figure 2.19 presents the results of Table 2.25 in the form of a graph plot.

Table 2.25: Combined representation of η_p , S and S_N

Land cover feature	Classification efficiency of BBO in % (η_p)	Similarity Index (S)= $\frac{1}{\frac{1}{u} \left[\sum_{u=1}^{count(EQp)} D_u \right]}$	Normalized Similarity Index (S_N)
Rocky	78.7%	2.13	1.00
Snow	73.1%	2.02	0.94
Vegetation	47.7%	1.39	0.65

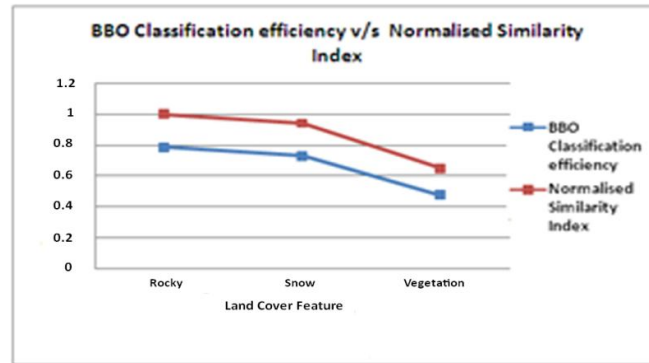


Figure 2.19: BBO Classification efficiency v/s Normalized Similarity Index for each land cover feature

III. Classification efficiency in terms of ‘S’ and ‘E’

Finally, the findings are integrated and a relation is established between η_p , S and E by constructing a combined table representing the classification efficiency of BBO, the average entropy in the NIR and MIR bands chosen and the similarity index ‘S’ for each of the three land cover features of vegetation, rocky and snow features as shown in Table 3.26 and

Table 2.26: Combined table of η_p , S and E for each land cover feature

Land cover feature	Classification efficiency of BBO in % (η_p)	Similarity Index (S)= $\frac{1}{\frac{1}{u} [\sum_{u=1}^{count(Eqp)} Du]}$	Normalized Similarity Index (S_N)	Average of Entropy in RS-1 & RS-2 bands (E)	Normalized Entropy (E_N)	S_N/E_N
Rocky	78.7%	2.13	1.00	0.6090	0.7106	1.407
Snow	73.1%	2.02	0.94	0.7360	0.8588	1.0945
Vegetation	47.7%	1.39	0.65	0.8570	1	0.65

graphically through figure 2.20. From the above table, it can be observed that the classification efficiency of BBO ' η_p ', for a given feature 'p' is proportional to the similarity index 'S' and inversely to the average entropy 'E' for 'p'. Hence the validity of equation (2.25) has been proved.

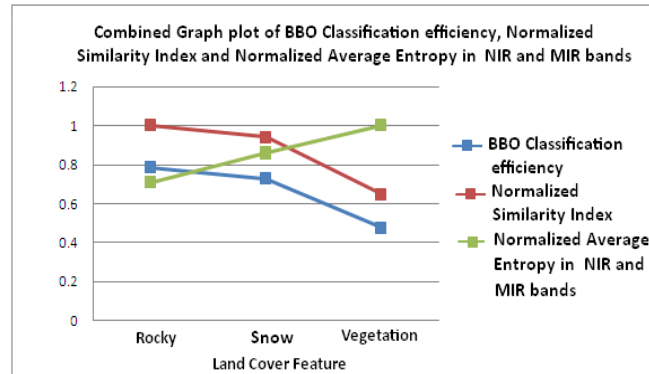


Figure 2.20: Combined graph plot of BBO Classification efficiency, Normalized Similarity Index and Average Entropy in NIR and MIR bands

2.5. Information Sharing In Swarm Intelligence Techniques: A Perspective Application for Land Cover Feature Extraction

As previously stated, swarm intelligence is based on the study of collective behavior in decentralized, self-organizing systems. It enables relatively simple agents to collectively perform complex tasks, which could not be performed by individual agents separately. Particles can interact either directly or indirectly (through the environment). The key to maintain global, self-organized behavior is social interaction i.e. information sharing between the system's individuals. Hence, information sharing is essential in swarm intelligence. This section highlights how the concept of information sharing in various swarm-based approaches is utilized for the elicitation of land cover features. The section provides a mathematical formulation of the concept of information sharing in each of the swarm intelligence techniques of Biogeography based optimization, Ant Colony Optimization, Particle Swarm optimization and Bee Colony Optimization which are the major constituents of the SI techniques that have

been used till date for extracting topographical facets over natural terrain. Particles involved in these techniques can either interact directly through waggle dancing (BCO), global-best positions (PSO) etc. or indirectly through the environment via pheromones (ACO), migration of SIV's between candidate solutions (BBO) etc. The basis of the increased intelligence is the shared information discovered individually and communicated to the swarm by different mechanisms of social interaction. In this way, intelligent solutions to problems naturally emerge from the self-organization and communication among simple individuals [(Goel, L., et. al., 2011B), (Goel, L., et. al., 2014)].

Figure 2.21 describes each of the SI techniques which are based on the information sharing concept along with the heuristic function which forms the base of the mathematical framework for adapting the respective SI technique to suit the purpose of geo-spatial feature extraction. Each of these heuristic equations and the adaptive framework for each of the SI techniques is described in detail in the respective sections below.

2.5.1. Information sharing in Biogeography Based Optimization (BBO) for feature extraction

This section discusses how the concept of information sharing finds its application in BBO has been used for elicitation of topographical facets. The major steps for image clustering and heuristic method (HSI) implementation, which form the pillars of the concept of adaptive application of information sharing for terrain features elicitation, are explained next [Goel, L., et. al., 2010C]. The m- bands image is clustered into H_i , using rough set theory. The parameters of clustering are NIR and MIR bands. The resultant clusters or the habitats H_i , are species in universal habitat.

These species are migrated to the suitable habitats i.e. feature islands. These habitats are actually the decisions, for the resultant different feature groups. The heuristic method decides which species are moved to which habitat using a fitness function [(Simon, D., 2008), (Ma, H., 2010)]. The fitness function is actually the information that is shared among all the habitats in order to decide the suitable habitat for migration of each species. Here average magnitude of absolute values of 'deviation from the mean metric' is used as the fitness function (HSI). This function for a specific class takes up the corresponding training sets and then calculates the required function values, which further help to decide the most suitable habitat according to the heuristic function Φ which was defined in equation (2.15) earlier. This function decides which value of mean of the chosen HSI metric has minimum difference from the original class. If this value is within the threshold then that class (species) will migrate to that habitat. The recalculated HSI after the migration of species is compared with the original HSI of the habitat (that contained training pixels only). If the compared HSI is within the threshold from the

original HSI then it means that the habitat is suitable for that species and hence it can migrate to it. If the recalculated HSI is not within the threshold for any of the original habitat then the particular elementary class contained a mixture of species and it is again partitioned into elementary classes.

2.5.2. Information sharing in Ant Colony Optimization (ACO) for feature extraction

This section discusses the adaptive strategy which is followed to use the concept of information sharing in ACO for elicitation of topographical facets. A widely adopted definition of information sharing through optimal clustering is a partitioning that the intra cluster similarity is minimized while the inter cluster similarity is maximized. For a given problem the pheromone can be set to be proportional to above criteria of the desired. In ACO, each image is assigned to a cluster and each ant is giving its own classification solution [(Bansal, S., et. al., 2009), (Piatrik, T., et. al., 2006)]. The algorithm starts by assigning a pheromone level τ and a heuristic information η to each image. The value for the pheromone level τ assigned to each image is initialized to 1 so that it does not have effect on the probability at the beginning. Heuristic information $\eta_{(X_i, C_j)}$ is obtained from the following equation (2.28):

$$\eta_{(X_i, C_j)} = \frac{B}{Sim(X_i, C_j)}, \quad (2.28)$$

where X_i represents the feature vector of i^{th} image and C_j is the feature vector representing j^{th} centroid of the cluster. $Sim(X_i, C_j)$ is the similarity between an image X_i and C_j . B is a constant where K is the number of clusters. Assume a number m of ants is chosen for clustering based on the K-Means approach. After all ants have done their classification, the assigned pheromone to this solution is incremented. In order to find global minimum, the pheromone value is updated according to quality of the solution. For updating the pheromone to each clustering the following formula in equation (2.29) is used:

$$\tau_{(X_i, C_j)}(t) = \rho \cdot \tau_{(X_i, C_j)}(t-1) + \sum_{a=1}^m \Delta \tau_{(X_i, C_j)}^a(t), \quad (2.29)$$

where ρ is the pheromone trail evaporation coefficient ($0 \leq \rho \leq 1$) which causes vanishing of the pheromones over the iterations. $\tau_{(X_i, C_j)}(t-1)$ represents the pheromone value from previous iteration. $\Delta \tau_{(X_i, C_j)}^a(t)$ in the equation above is a new amount of pheromones calculated from all m ants that assign image X_i to j^{th} cluster. This approach of marking solutions by pheromone which is based on the concept of information sharing is discussed as follows in equations (2.30-2.32):

$$InterSim(a, j) = \sum_{n \neq j}^K Sim(C_j, C_n), \quad (2.30)$$

$$InterSim(a) = \sum_{i \in S_j}^{N_j} \sum_{j=1}^K Sim(X_i, C_j), \quad (2.31)$$

$$\Delta\tau_{(X_i,Cj)}^a = \begin{cases} \frac{N \cdot InterSim(a,j)}{InterSim(a)} & \text{if } X_i \text{ belongs to } j^{th} \text{ cluster} \\ 0 & \text{otherwise} \end{cases} \quad (2.32)$$

$InterSim(a,j)$ represents the sum of the similarities obtained by ant a , between j^{th} centroid and the rest of centroids. $IntraSim(a)$ represents the sum of the similarities obtained by ant a , between each image and its centroid. N is the number of images in the dataset and this variable keeps the values of $InterSim(a,j)$ and $IntraSim(a)$ in the same order. It becomes clear that the pheromone increases when clusters get more apart and when each cluster has more similar images. Next, the classification performed by each ant is driven by the quality of previous solutions. This is repeated until the best solution for all ants is achieved [Patrik, T., et. al., 2006].

2.5.3. Information sharing in Bee Colony Optimization (BCO) for feature extraction

This section discusses the adaptive strategy which is followed to use the concept of information sharing in BCO through FPAB/BBO algorithm [(Johal, N. K., et. al., 2010), (Wong, L.P., et. al., 2008)] for elicitation of topographical facets. Initially, flowers are scattered randomly on 2D discrete grid, which is named as jungle. This jungle can be considered as a matrix of $M \times M$ cells. The matrix is toroidal which allows the bees to fly from one end to another easily and there is initially not more than one flower in each cell of jungle. The size of the jungle depends on the number of flowers. A jungle of $M \times M$ is used such $m^2 = 4n$ that where n is the total number of flowers (objects) to be clustered. Initially, the bees are randomly scattered throughout the jungle. The no. of bees used is $n/3$, where n is the total number of flowers to be clustered. Each garden defined as a collection of 2 or more flowers (or pollens) and for simplicity of algorithm each garden spatially located in a single cell. Consider a garden G with n_g flowers. The following parameters are defined:

The compatibility between two flowers in the garden

$$C(p_i, p_j) = 1 - D(p_i, p_j) \quad (2.33)$$

where 'D' is normalized Euclidean distance between the attributes of flowers p_i, p_j .

The center of garden G

$$O_{center}(G) = \frac{1}{n_g} \sum_{O_i \in G}^n O_i \quad (2.34)$$

The growth of the pollen in a garden

$$G(p_i) = C(p_i, O_{center}) \quad (2.35)$$

The flower with minimum growth in a garden G

$$p_{min} = \arg \min G(p_i) \quad (2.36)$$

where $p_i \in G$.

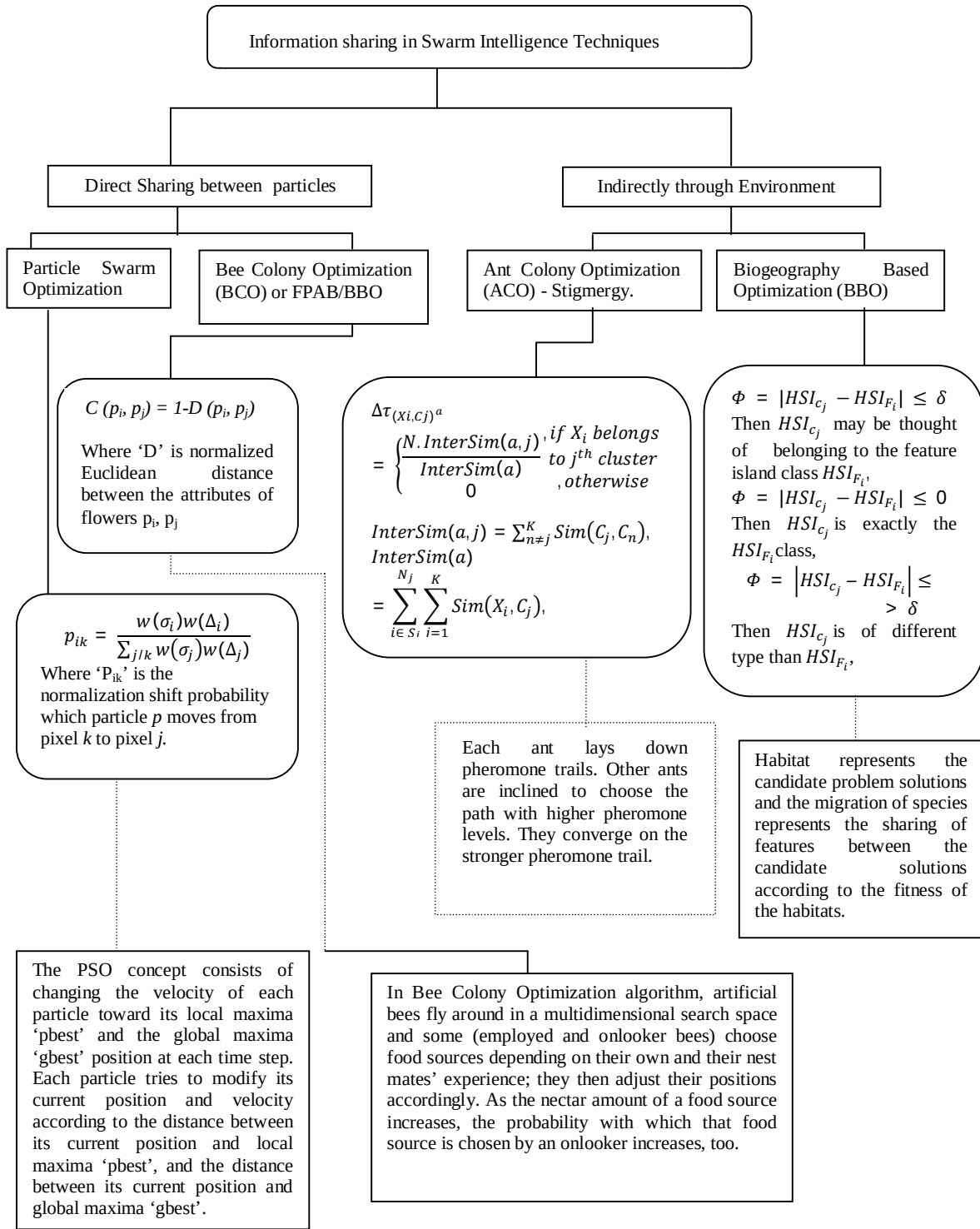


Figure 2.21: An overview of information sharing for land cover feature extraction in swarm-based approaches

The main FPAB algorithm [Johal, N. K., et. al., 2010] is written below:

1. Randomly scatter bees and flowers into the jungle (atmost one flower per cell)
2. Set the initial growth of all flowers to zero
3. Repeat 4, 5, 6 until stopping criteria
4. For each bee do

a) Fly the bee in jungle

b) If the bee does not carry any pollen then it possibly picks up a pollen else the bee possibly pollinates.

5. Next bee

6. Natural Selection

Initially, the bees are randomly scattered into the jungle and also randomly pick up or pollinate when they reach a new garden in the jungle. Here, the stopping criterion for the bees is the number of times through the repeat loop.

For extraction of natural terrain features, the procedure followed is explained next. Initially homogeneous clusters of image pixels are obtained using FPAB algorithm discussed above. Clusters obtained by honey bees are treated as species in BBO and are put to universal habitats. There are other feature habitats containing the training pixels of the corresponding feature produced by experts. For example, water habitat initially contains pixels of water and HSI metric (i.e. DMM) of all the training pixels of water, is calculated in each of seven bands. The HSI is calculated on these training pixels. Each species is taken from the universal habitat and is migrated to each feature habitat one at a time. The HSI of the habitat is recalculated after migrating the species to it. The difference between the original HSI and recalculated HSI is calculated. And if the difference is minimum in some particular feature habitat, species is migrated to that habitat.

2.5.4. Information sharing in Particle Swarm Optimization (PSO) for feature extraction

This section discusses the adaptive strategy which is followed to use the concept of information sharing in PSO for elicitation of topographical facets. Particle swarm classification model treats two dimensional remote sensing images as gravitational field. Each of pixels has different gravitation to particles locating in neighboring of the pixel. The gravitation is searching information in above-mentioned process, but it is updated continually by those particles which has reached the pixel according to classification rules. If the pixel belongs to their types to which particles belong, the gravitation of it to those particles locating in 8-neighborhood around the pixel will be increased by a certain number, otherwise the gravitation will be decreased gradually until to 0. The gravitation can be described using quantitative indicators, and is real number in $[0, 1]$. The larger the gravitation value is the stronger the gravitation is, otherwise the more weaken the gravitation is.

Particles explore next pixels with certain probability in each time-step, but weight value W of next pixels along with searching path are different. At the same time, there is only one particle in each of pixels because operation environment of particles is two-dimensional images. In order to keep consistency as far as possible with natural behavior of bird swarms, time will be

dispersed. Each of particles is permitted to move one and only one step in a time-step. In other words, particles must select one pixel to reach from 8 neighboring pixels. At certain time-step t , if the pixel in which the particle locates belongs to congener through distinguished by the particle, the particle will adjust gravitation of the pixel. Adjustment range is η . In the meanwhile, all of pixels' gravitations will be weakened along with the process of time. The weaken range is κ . According to above conditions, definition of normalization shift probability which particle p moves from pixel k to pixel j is given by equation (2.37):

$$p_{ik} = \frac{w(\sigma_i)w(\Delta_i)}{\sum_{j/k} w(\sigma_j)w(\Delta_j)} \quad (2.37)$$

In above formula, denominator is a summation formula which represents a weighted sum of gravitations belonging to all of pixels j from 8 neighboring pixels of pixel k . Δ_j represents direction change quantity of particle p at $t - 1$ time. Its value is one of five discrete W values. Each of particles will move stochastically with probability p_{ik} after establishing these parameter values. By this time, each of particles will explore next pixels according to above-mentioned mechanism [Wang, D., et. al., 2008].

2.5.5. Discussion

The section demonstrated the fact that the majority of the discovered swarm intelligence techniques are based on the concepts of information sharing. It has been observed that information sharing is an essential part in all swarm based approaches. The particles in a swarm try to solve a problem as a group by using the information contained by their peer mates. Information exchange in swarm-based approaches can take place either directly (as in PSO and BCO) or via the environment (indirectly as in BBO and ACO). The sharing of information enables swarm particles to be more efficient and to achieve goals that they could not achieve individually. It also helps to intensify the search in the promising regions of the solution space. The information that is shared between individuals in various swarm intelligence techniques can be represented using knowledge representation techniques, which further helps to optimize the problem more efficiently. In future, the concept of information sharing can be exploited further to adapt to other applications such as groundwater exploration through case based reasoning using any of the above SI techniques. Also, the above concept can also prove to be useful in military applications for predicting the deployment strategies of enemy troops.

2.6. Observations

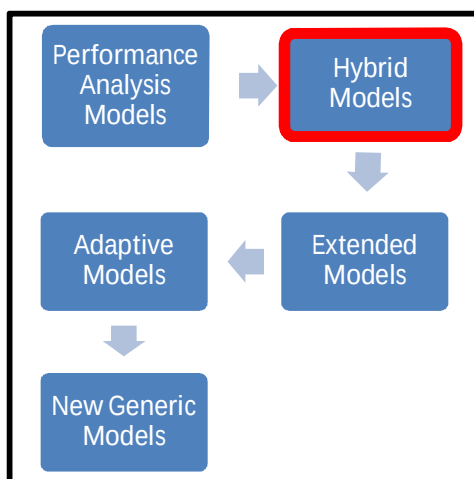
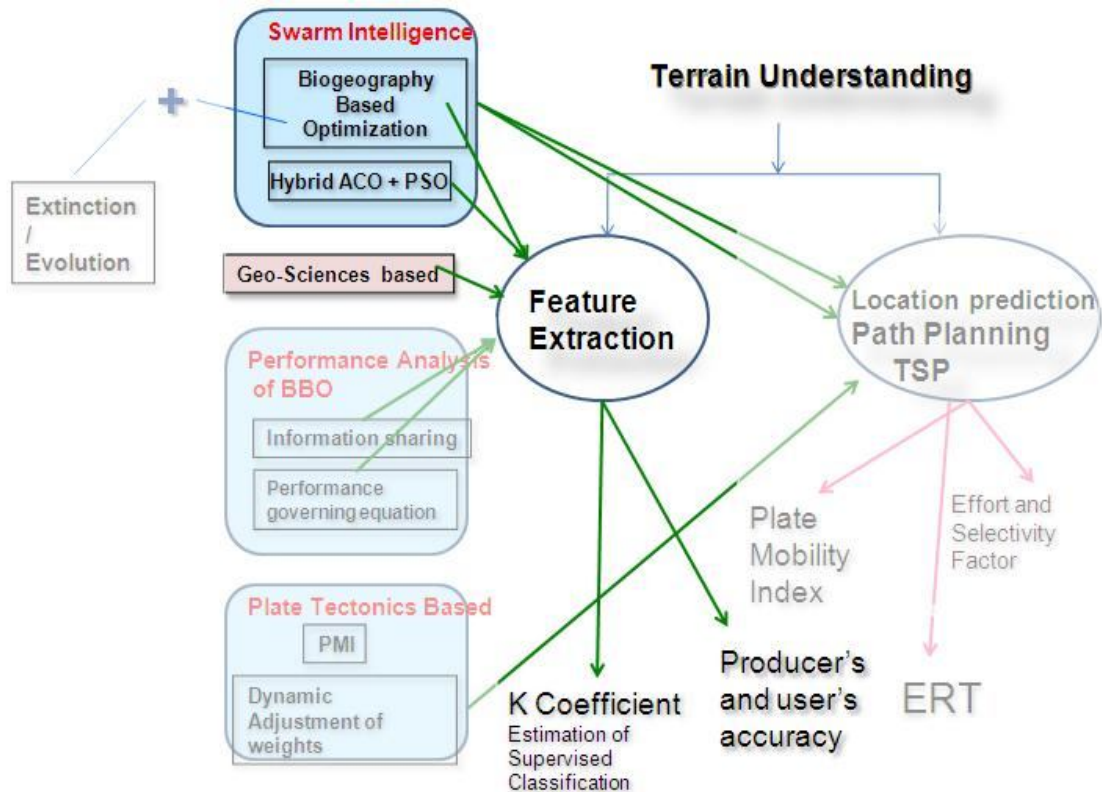
The main aim in this chapter was to draw inference regarding the applicability of BBO for the extraction of given land cover features by identifying the classification efficiency of the biogeography based feature extractor developed and described in [(Panchal, V. K., et. al.,

2009), (Panchal, V. K., et. al., 2011)] towards different natural terrain features. Based on these analytical results, it is concluded that which land cover features should be extracted using BBO. This finding is of prime importance since a prior analysis of the performance of BBO on different types of terrain under consideration will improve the analyst's decisive capabilities for the selection of the most appropriate technique for the feature extraction task in hand or improve the classification efficiency of artificially intelligent hybrid classifiers by applying the BBO technique on the extraction of those features on which it shows better classification efficiency than the other classifiers available as discussed in the chapter. The results indicate that BBO is able to classify the homogeneous regions i.e. the regions with the lowest entropy and hence highest similarity index, more efficiently than the regions which show a greater degree of heterogeneity. Analyzing the BBO behavior, it is observed that the classification efficiency of BBO on a land cover feature can be increased if the similarity index between the candidate equivalence classes classified into a given feature and the training set increases which in turn will decrease the entropy and hence increase the overall classification efficiency on that feature. The similarity index can be increased by replacing the similarity criteria or the HSI i.e. difference in the absolute values of 'deviation from mean metric' of the equivalence class and the training set, by artificial neural networks or any swarm intelligence technique which is more efficient in unsupervised clustering so that the similarity clusters generated exhibit a greater degree of similarity and hence would lead to a higher similarity index when compared with the training set used thus leading to a lower entropy and a higher overall classification efficiency on the given land cover feature.

The other aim was to study the swarm intelligence techniques from the perspective of how they utilize the information sharing concepts to reach towards an optimal solution. This analysis was observed while carrying out the behavioral study of BBO. Since BBO is a swarm intelligence technique, the concept was generalized to fit in other core SI techniques and it was found that most of the SI techniques effectively use the information sharing concepts either directly or indirectly to reach to a global solution. The chapter studies the four major techniques that form the pillars of swarm intelligence however, in future other techniques can also be analyzed in order to fully establish the concept of information sharing as a powerful paradigm leading towards optimal solutions.

The next chapter discusses the application of the analytical study of the biogeography based land cover feature extractor carried out in this chapter as a preprocessing step towards the building of artificially intelligent hybrid feature extractors (or classifiers).

Hybrid Models of Nature Inspired Computational Intelligence For Geo-Spatial Feature Extraction



Chapter 3

- 3.1. Hybrid Bio-Inspired Techniques for Land Cover Feature Extraction
- 3.2. Biogeography and Geo-Sciences Based Geo-spatial Feature Extraction
- 3.3. Observations

This chapter presents a framework for the hybrid feature extractors [(Holden, N., et. al., 2005), (Goel, L., 2010B), (Bansal, S., et. al., 2009)] where the performance analysis model describing the equation governing the performance of BBO [(Beyer, H., 2001), (Goel, L., et. al., 2010C)] is utilized in order to derive inference about which features can be best classified and hence should be extracted using BBO. The other features can be classified using hybrid ACO2/PSO for the development of hybrid intelligent classifiers that can lead to significant improvements in its classification capabilities.

The chapter is organized as follows: Section 3.1 presents a hybrid model called the ACO2/PSO/BBO technique which combines the strengths of the modified BBO technique for land cover feature extraction with the ACO2/PSO technique by adapting to the database of expert knowledge for a more focused satellite image classification [Goel, L., et. al., 2012A]. The classifier is intelligent to identify the features efficiently classified by BBO and by ACO2/PSO separately based on the proposed performance governing model in turn based on a similarity pattern analysis [Plaza, A., et. al., 2008]. The classifier has been applied to the satellite image of the Alwar area in Rajasthan. A comparative study of the results of applying the classifier and the other recent CI based classifiers as well as the traditional probabilistic ones is also presented on the Alwar image which is a benchmark image for testing the performance of a bio-inspired classifier on multi-spectral satellite images since this image is a complete image containing all the important land cover features that are required to be extracted.

Section 3.2 presents a novel hybrid model inspired from geo-sciences, the biogeography and geo-sciences based CI technique that is an extension of original biogeography based feature extraction algorithm using the concept of entropy inspired from the geo-sciences phenomenon of mantle convection and dynamics of the earth [(Goel, L., et. al., 2011C), (Goel, L., et. al., 2013A)]. This algorithm uses surface entropy in the relevant band of multi-spectral images as the basis of calculating the habitat suitability index which in turn forms the basis of identifying different terrain features in the satellite image. The work has been primarily developed for the purpose of finding the applications of geo-sciences in developing computationally intelligent models. The proposed feature extractor algorithm has been applied on the datasets of Alwar region in Rajasthan and Patalganga area in Shivalik ranges. The results indicate that the proposed geo-sciences based classifier is highly efficient in extracting land cover features. Further when integrated with hybrid bio-inspired intelligent classifier, it improves its classification efficiency and also outperforms the earlier probabilistic classifiers and the recent CI based classifiers such as membrane computing,

hybrid FPAB/BBO, extended non-linear BBO, etc. [(Goel, L., 2010B), (Goel, L., et. al., 2012A)]. The results conclude that the classifier based on the proposed model is one of the best known classifiers developed till date. The proposed model is flexible and can adapt itself to suit to a large number of classification problems including mixed pixel resolution, face recognition, pattern recognition etc. whereby entropy can be simply calculated in any other band or according to its standard definition and hence feature extraction can be made. Finally, Section 3.3 concludes the work and discusses the observations made.

3.1. Hybrid Bio-Inspired Techniques for Land Cover Feature Extraction

Out of the enormous swarm intelligence techniques available for image classification, it is essential to select the most appropriate technique for the feature extraction task in hand. By the motivation of the characteristic strengths of the BBO technique [(Simon, D., 2008), (Simon, D., et. al., 2009), (Currie, D. J., 2012)] to classify the desirable features more efficiently (as is reflected from the performance governing model of BBO in the previous chapter) and the capability for improvement in its solutions probabilistically with each iteration during the classification process, an algorithm which is a hybrid of the ACO, PSO and BBO techniques is formulated for building an intelligent hybrid classifier for satellite images.

The section first presents a brief review of the hybrid ACO2/PSO techniques and motivation towards the proposed methodology. The proposed framework of the hybrid ACO2/PSO/BBO algorithm - the dataset used, the proposed architecture, and the parameters used is hence presented. Next, the section presents the classification results of the Alwar image using ACO2/PSO/BBO technique and provides a comparison of its classification efficiency in terms of kappa coefficient with the BBO, the traditional probabilistic classifiers as well as the other recent CI based techniques.

3.1.1. A Brief Review of hybrid ACO2/PSO

The hybrid ACO-PSO given by Nicholas and Frietas uses sequential covering approach for rule extraction which directly deals with both the continuous and nominal attribute-values [(Holden, N., et. al., 2005), (Bansal, S., et. al., 2009), (Bratton, et. al, 2007)] and can be understood by figure 3.1. It is necessary to estimate the quality of every candidate rule [(Alpaydin, E., 2004), (Hand, J., 1997). A measure must be used in the training phase in an attempt to estimate how well a rule will perform in the testing phase. Given such a measure it becomes possible to optimize a rule's quality (the fitness function) in the training phase and this is the aim of the ACO2/PSO algorithm. In ACO/PSO the Quality measure used was Sensitivity * Specificity as given by equations (3.1) – (3.3) where TP, FN, FP and TN are,

```

Initially Rule Set is empty ( $\Phi$ )
For Each class of cases Trs = {All training cases}
    While (No. of uncovered training cases of class A > Maximum uncovered cases / class)
        Run the PSO/ACO algorithm for finding best nominal rule
        Run the standard PSO algorithm to add continuous terms to Rule, and return
        the best discovered rule BestRule
        Prune the discovered BestRule
        RuleSet = RuleSet U BestRule
        Trs = Trs – {training cases correctly covered by discovered rule}
    End of while loop
End of for loop
Order these rules in RuleSet by descending Quality

```

Figure 3.1: Hybrid ACO2/PSO algorithm

respectively, the number of true positives, false negatives, false positives and true negatives associated with the rule [(Hand, J., 1997), (Holden, N., et. al., 2005), (Bansal, S., et. al., 2009)].

$$\text{Sensitivity} * \text{Specificity} = ((\text{TP}) / ((\text{TP}) + (\text{FN}))) * ((\text{TN}) / ((\text{TN}) + (\text{FP}))) \quad (3.1)$$

Equation 1: Original Quality Measure

Later it is modified as equation (3.2) given below-

$$\text{Sensitivity} * \text{Precision} = ((\text{TP}) / ((\text{TP}) + (\text{FP}))) * ((\text{TP}) / ((\text{TP}) + (\text{FP}))) \quad (3.2)$$

Equation 2: Quality Measure on Minority Class

This is also modified with using Laplace correction as:

$$\text{Precision} = 1 + (\text{TP}) / (1 + b + (\text{TP}) + (\text{FP})) \quad (3.3)$$

Equation 3: New Quality Measure on Minority Class

where 'b' is the number of classes.

So, ACO/PSO attempted to optimize both the continuous and nominal attributes present in a rule antecedent at the same time, whereas ACO2/PSO takes the best nominal rule built by ACO2/PSO and then attempts to add continuous attributes using a standard PSO algorithm.

3.1.2. Land Cover Feature Extraction using ACO2/PSO technique

Here, the modified hybrid ACO2/PSO adapted to the problem of land cover feature extraction is described. A multi-spectral satellite image is taken as input for classification. By these multi-band images the training set is generated using ERDAS IMAGINE [(Long, W. III, et. al., 2004), (Lillesand, T. M., et. al., 2008)] which is provided in the form of digital numbers (intensity value pixel in a digital image). This dataset can be in any excel sheet, access sheet or in text file. The decision according to the DN values is also set in the table.

From these DN values the rules are extracted by applying the ACO2/PSO algorithm. To apply the modified hybrid ACO2/PSO, the open source tool for hybrid ACO2/PSO is used. Before applying these algorithms for rule-set generation the training set is changed into the unified attribute relation file format, i.e., .arff format. The rule sets are obtained from the open source tool for ACO2/PSO [Open Source Tool, Source Forgenet]. But this rule set does not provide the classified image. So these rules are coded in a MATLAB file. In MATLAB coding the multi-band images are read and then for each pixel of the image, the DN values are checked according to the rule set contained. If the pixel validates any rule then the decision of the rule is set to be the class of the pixel and the pixel is set to a specified color. On executing this MATLAB file, the final classified image is obtained. The algorithm for the ACO2/PSO image classification [(Holden, N., et. al., 2005), (Bansal, S., et. al., 2009)] is summarized in figure 3.2.

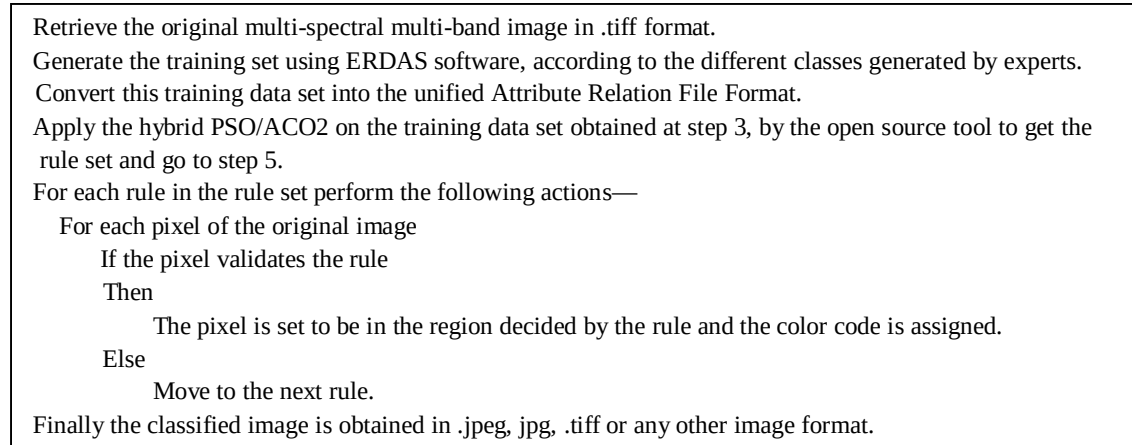


Figure 3.2: Modified algorithm for hybrid ACO2/PSO for land cover feature extraction

3.1.3. Proposed Framework for the hybrid ACO2/PSO/BBO based feature extraction

In this section, the proposed framework for the hybrid ACO2/PSO/BBO based land cover feature extraction is described. The section includes the architecture, mathematical formulation and also includes the detailed internal working of the hybrid land cover feature extractor.

A. Proposed Architecture

The proposed software architecture of the bio-inspired pattern analysis based intelligent classifier is given below in figure 3.3. The above layers of the software architecture is explained step by step below-

Input layer: A high resolution multi-spectral satellite image is taken as input. For illustration, the image used is the 7-band resourcesat satellite image of size 472 X 546 of the Alwar Region in Rajasthan.

Image Analyzer Layer: This layer is responsible for analysis of the input satellite image to exploit the data distribution of the training set and feed its output to the next layer i.e. selective cluster identifier layer.

- (i) **Training set Generation Layer:** This layer generates the training set using ERDAS software, according to the different classes by experts.
- (ii) **Data Distribution Graph Generation Layer:** Now, using this training set, the data distribution graph is plotted between the absolute values of the HSI metric i.e. ‘deviation from mean metric’ (equation (2.12) in chapter 2) of each land cover feature viz water, urban, rocky, vegetation and barren (plotted on the y-axis) for each of the 7-bands of the image i.e. red, green, NIR, MIR, RS1, RS2 and DEM (plotted as the x-axis) as shown in figure 3.4. On the x-axis, each of the 7-bands of the image are represented as integers from 4-10 for red, green, NIR, MIR, RS1, RS2 and DEM bands respectively. Integers 1-3 in the graph plot on the x-axis are not considered since these represent the other attributes of the image namely pixel no. and x and y coordinates which are not required at this stage.
- (iii) **Graph Pattern Similarity Analysis Layer:** Based on the observations from this graph, a similarity analysis is done to observe the features with the minimum difference value which will be the ones that will be classified with similar efficiency by the hybrid classifier. The group of features that will be more efficiently classified by BBO are identified since based on the proposed performance governing model of BBO which explores the adaptive nature of BBO algorithm for land cover feature extraction, an analysis of the performance of BBO towards different terrain features is made and inference about which land cover features can be better classified by using BBO technique is done. From the graph, it can be observed that the minimum difference between the absolute values of the ‘deviation from mean metric’ of the NIR and the MIR bands of the Alwar image is achieved in particularly two land cover features, those of water and urban area, both of which exhibit the same graph pattern in the NIR and the MIR bands i.e.

$$| \text{DMM for NIR band} \sim \text{DMM for MIR band} |_{\text{lowest}} = \{\text{water, urban}\}$$

Hence, it can be concluded that these are the two features that will be most efficiently classified by the hybrid algorithm which works in the NIR and MIR bands. This can also be verified by the performance analysis model of BBO generated from the data distribution pattern, wherein it was observed that BBO shows the highest efficiency on water and vegetation pixels which had the lowest entropy and highest similarity index values (see Table 2.13 of chapter 2).

Selective Cluster Identifier Layer: Based on the observation of the adaptive nature of BBO algorithm for land cover feature extraction, it is observed that BBO is able to classify some particular feature's pixels with greater efficiency than the other features based on the band that is selected for creation of equivalence classes in Rosetta. The functionality of this layer is described in detail below-

(i) Image Grid Division Layer: In this layer, the satellite image is divided into 'n' clusters, in this case $n = 20$. In the biogeography based classification layer, basically the aim is to identify to which cluster of the image BBO technique will show good classification efficiency and hence should be applied to.

(ii) Rough Set Theory Application Layer: For the purpose, rough set theory toolkit i.e. *Rosetta software* [Ohm, A., et. al., 1997] is used for discretizing each of the 20 clusters using the semi-naïve algorithm & then partition each of them based on the band which is able to classify the particular feature that needs to be extracted from the image. This is because BBO follows an adaptive strategy and will classify the desired feature most efficiently since it has its corresponding band selected in the partitioning step.

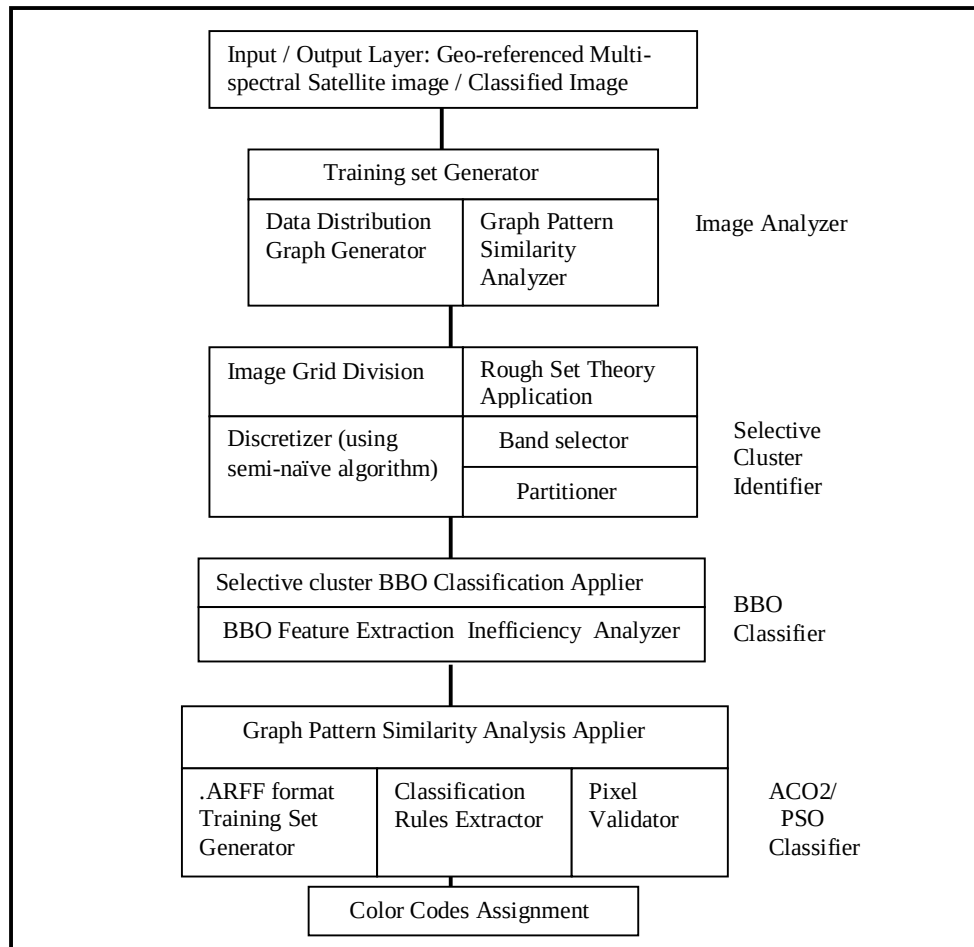


Figure 3.3: Software architecture of the proposed intelligent classifier

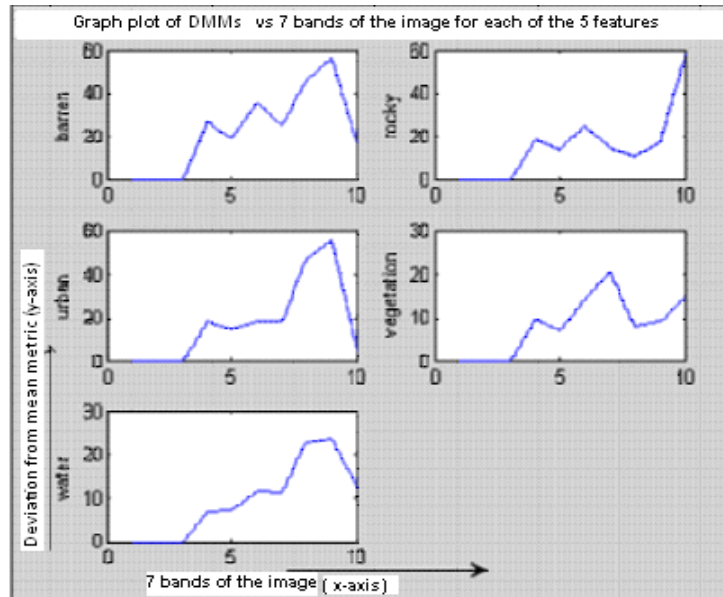


Figure 3.4: Graph plot of the DMIs of each land cover feature v/s each of the 7-bands in which the Alwar image is viewed.

(iii) Band selection Layer: Depending on the application, in other words depending on which feature needs to be extracted from the image most efficiently, the band for partitioning can be chosen. For example, if the barren area is needed to be extracted more efficiently, the green band is chosen and for rocky region extraction the MIR Band is chosen. The RS-1 and RS-2 bands are used to extract the urban area and also for extracting the edges of rocky region from the 7-band image. However, the drainages of rocky region are best viewed in the red band and water and vegetation pixels are best viewed in NIR and MIR Bands. For illustration, NIR and MIR bands of the 7-band image are chosen since the water pixels need to be extracted effectively in order to clearly identify the water body in the image and these are the bands in which the water feature is particularly more highlighted and best viewed.

(iv) Discretization and Partitioning Layer: Therefore, the NIR and the MIR bands are used for discretization and partitioning step in the semi-naïve algorithm used for creating rough set equivalence classes, thus creating equivalence classes for each of the clusters. This is what is termed as unsupervised classification. Each of these resultant classes are put in the universal habitat.

BBO Classifier Layer:

(i) Selective Cluster BBO Classification Applier Layer: Hence, the BBO technique is applied on those clusters of the satellite image which show the maximum classification efficiency which is due to the fact that these are the clusters which predominantly show the presence of the feature that is most efficiently classified by the BBO algorithm

(based on the observations from the data distribution graph and also the performance analysis model). The procedure for applying the biogeography based land cover feature extraction technique was described in section 2.2 of chapter 2.

In this illustration, BBO will be able to extract the water pixels most efficiently since the NIR and MIR bands are chosen for partitioning. And therefore, BBO is applied on the k^{th} cluster of the Alwar image since this is the cluster which gives the maximum classification efficiency because it predominantly shows the presence of water body in the image.

- (ii) **BBO Feature Extraction Inefficiency Analysis Layer:** However, it is also observed that BBO shows a wide range of efficiencies in classifying the rest of the features. In other words, BBO is not able to classify all the 5 features with good classification efficiency and shows poor performance on some of the remaining features. In fact, in the illustration, it shows the poorest performance in classifying the urban pixels. Hence it is required to proceed for the ACO2/PSO classification to improve the image classified by BBO.

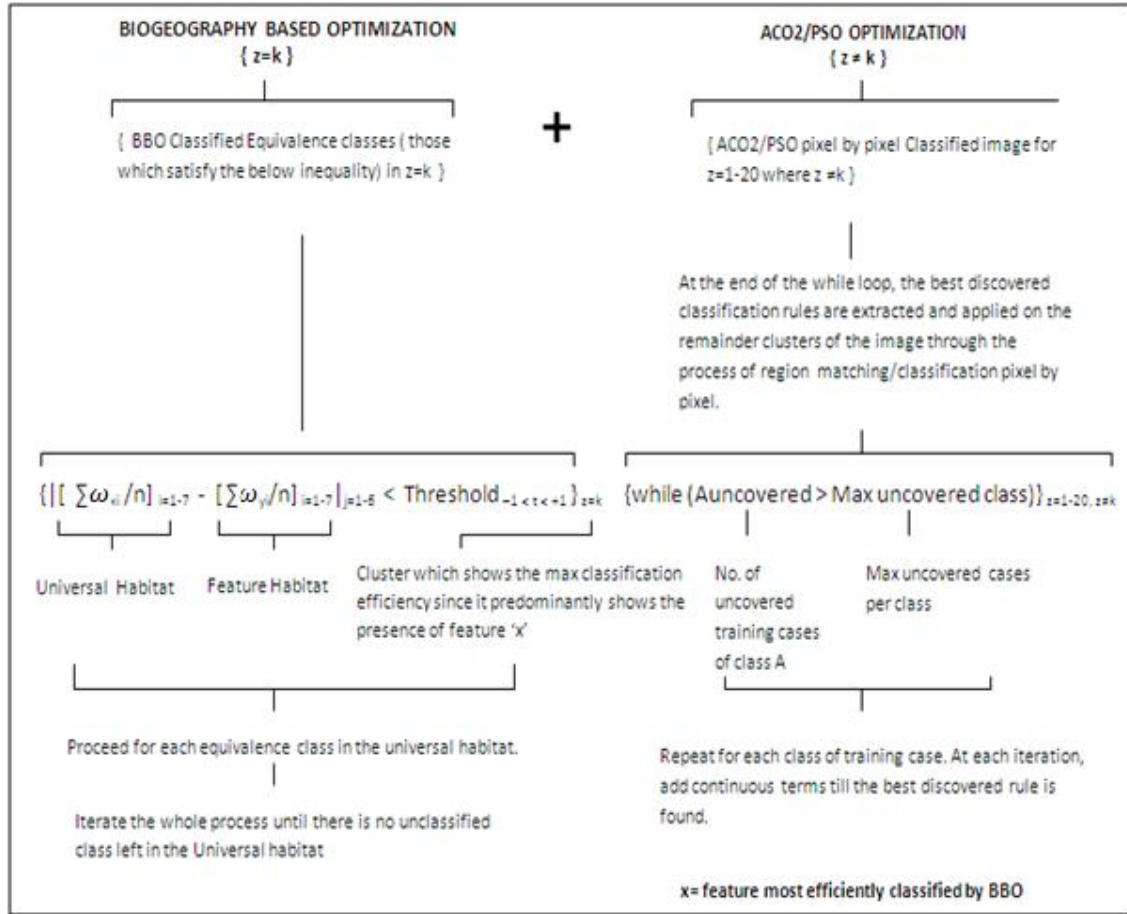
ACO2/PSO Classifier Layer:

- (i) **Graph Pattern Similarity Analysis Application Layer:** Now, it is known from the data distribution graph plotted in the Image Analyzer Layer that the hybrid algorithm will show similar efficiency (i.e. maximum) for the urban pixels too as for the water pixels since their graph pattern in the NIR and MIR bands is similar, therefore, ACO2/PSO Technique is then applied on the rest of the image. In this case, BBO performed better than ACO2/PSO only on the water pixels and hence BBO is applied on the k^{th} cluster. Although BBO gave good performance on vegetation pixels too (since in NIR and MIR bands these are the 2 features that are best viewed), however, the efficiency was no better than ACO2/PSO classification and hence BBO was not applied on the vegetation pixels.
- (ii) **Training set in .arff format Generator Layer:** In order to apply ACO2/PSO classifier, the training set for the 7-band Alwar image is taken in .arff format as input to generate the classification rules from it.
- (iii) **Classification Rules Extractor Layer:** For the purpose, the hybrid ACO2/PSO open source tool is used and then the extracted classification rules are applied on each of the remainder clusters of the image.
- (iv) **Pixel Validator Layer:** Each of the pixels of the image is checked for pixel validation on each pixel in the cluster & thus a refined classification of the image is obtained. The above procedure of applying ACO2/PSO generated rules for classifying the image is as described in section 3.1.2.

Color Codes Assignment Layer: Finally color codes are assigned for each pixel of the image corresponding to the classified feature of each pixel. The color codes for water, urban, vegetation, rocky and barren features are assigned as blue, red, green, yellow and black color respectively.

Output Layer: Final ACO2/PSO/BBO classified image is obtained in .jpeg, .tiff or any other image format.

B. Mathematical Formulation of the ACO2/PSO/BBO Algorithm



The working of the hybrid classifier is summarized in the form of an equation in figure 3.5.

Figure 3.5: Mathematical formulation of the ACO2/PSO/BBO classifier

where the universal habitat contains the rough set classified equivalence classes and the feature habitat consists of the expert generated training set of the original Alwar image in 7-bands.

And,

Z=input cluster of the 7-band image (z ranges from 1-20)

J= feature under consideration (j ranges from 1-6 , j=1 for unclassified feature)

X=feature most efficiently classified by BBO.

K=cluster which shows the maximum classification efficiency since it predominantly shows the presence of feature 'x'.

The above equation can be explained as follows-

- (i) For $z = k$, the algorithm proceeds in the following manner for the BBO optimizer -i.e. for each i^{th} band where 'i' ranges from 1-7, the difference in the 'deviation from mean metric' of the i^{th} band of the universal habitat and the i^{th} band of the feature habitat containing the expert generated training set of the image is calculated. If this difference is the minimum for the feature 'j' and also less than the pre-specified threshold value of $-1 < t < +1$, then that particular equivalence class is classified as the feature 'j' else $j=1$ (unclassified). The process is repeated for each equivalence class until there is no equivalence class left in the universal habitat and the whole process is iterated till there is no unclassified equivalence class left.
- (ii) For $z = 1-20$, where $z \neq k$, the ACO2/PSO optimization is used, wherein the training set for the 7-band Alwar image in .arff format is used as input to generate rules from it using the open source tool for each class of training case and on each iteration, continuous terms are added till the best discovered rule is found. The classification rules are then applied on the remainder of the clusters checking for pixel validation on each of them.

The aim of the above functional architecture is to provide a refined classified image which improves the classification results obtained on applying the original BBO algorithm on the multi-spectral image. This in turn leads to the improved flexible hybrid version of the BBO algorithm for satellite image classification which will classify the particular feature chosen by the band used in the unsupervised classification, most efficiently based on the expert knowledge [Goel, L., et. al., 2010C] and the band information contained in the training set of the particular area. Thus, the properties of the BBO technique are efficiently exploited to adapt itself to a more focused classification using the proposed equation governing its performance which upon integrating with the ACO2/PSO technique makes an advanced classifier [Goel, L., et. al., 2010C], which is termed as the hybrid bio-inspired pattern analysis based intelligent classifier.

3.1.4. Results and Discussion

Figure 3.6 shows the 16th cluster of the Alwar image after running multiple iterations of biogeography based land cover feature extraction algorithm on it with the threshold value taken as -1 to +1. The yellow color represents rocky area, green color represents vegetation area, black color represents barren area and red color represents the urban area. The white color in the image represents the unclassified pixels of image in the first iteration. It is analogous to those species that are not absorbed in any feature habitat and are put back to the Universal habitat. These species represent the unclassified pixels that can be further classified in the next iteration of algorithm. By taking the threshold value from -1 to +1, it is seen that no water body (represented in blue color) is extracted in the first iteration (figure

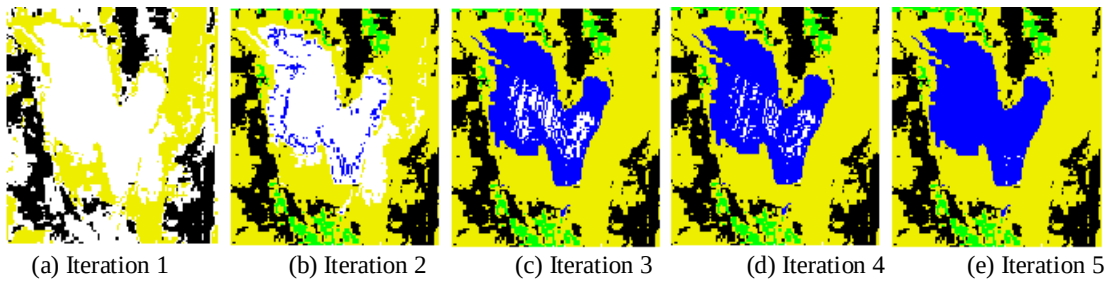


Figure 3.6: Water body extraction (in blue color) from the Alwar image with each iteration. White color portion represents the unclassified image.

3.6 (a)) and water pixels remain unclassified (represented in white color). With the increasing iteration (figure 3.6 (b), 3.6 (c), 3.6 (d), 3.6 (e)), water body is clearly extracted at the end of fifth iteration [Panchal, V. K., et. al., 2009]. After five iterations of the algorithm each species is absorbed in the corresponding habitat and the universal habitat becomes empty. This terminates the biogeography based feature extraction algorithm on the 16th cluster. Now from the performance analysis results shown in Table 2.13 (see chapter 2) for the Alwar dataset, displaying the values of entropy and similarity index for each of the land cover features which are required to be extracted from the image, it is observed that the water feature which shows the lowest entropy and highest similarity index value has been predicted by the efficiency analyzer as the feature which should be most efficiently classified by BBO. Hence, based on this analysis, it is observed that the algorithm is able to classify water pixels with the highest efficiency i.e. 99% efficiency and these are the pixels best viewed in the NIR and MIR bands in the BBO technique & hence, the BBO technique is applied on the 16th cluster of the satellite image of Alwar region ($z = 16$) since this is the cluster which gives the maximum classification efficiency since it predominantly shows the presence of water body in the Alwar image. However, BBO shows poor efficiency, in fact the poorest, in classifying the urban pixels as shown in figure 3.7. Here the encircled region in the BBO classified image shows that BBO wrongly classifies the urban pixels as barren ones which is also reflected from Table 3.1 where BBO classifies 91 urban pixels wrongly out of 190 total urban pixels.

Therefore, in order to classify the urban pixels efficiently, the ACO2/PSO technique [Dorigo, M., et. al., 2004] is then applied on the remainder of the clusters of the image ($z \neq 16$) by taking the training set for the 7-Band Alwar image in .arff format as input to generate rules from it using the open source tool and then applying them on the remainder of the clusters checking for pixel validation for each pixel in the cluster & thus obtaining a more refined classification of the image with an improved kappa coefficient of 0.9818 which is much better than the kappa coefficient of 0.6715 [Panchal, V. K., et. al., 2009] that is obtained on applying the original BBO algorithm on the 7-band image. This in turn leads to

the improved hybrid version of the BBO algorithm for satellite image classification where both the urban and the water features are classified with the highest efficiency i.e. almost 100% with no omission errors followed by rocky with only 1 omission error (column wise error) and thereafter barren and vegetation features, respectively. After applying the proposed algorithm to the Alwar image, the classified image is obtained in figure 3.8. From the figure, it is clearly shown that the proposed ACO2/PSO/BBO classifier is able to correctly classify the encircled region as urban which was wrongly classified by the simple BBO classifier. The yellow, black, blue, green, red color represents rocky, barren, water, vegetation, urban region respectively. As the threshold limit of HSI matching is lowered, the species do not get absorbed in the feature habitat and return to universal habitat. Those species are further discretized and classified in next iterations (generation). From the figures 3.7 & 3.8, it is evident that the hybrid ACO2/PSO/BBO technique produces a more refined image as compared to the BBO classified image.

3.1.5. Accuracy assessment of the proposed algorithm

Accuracy assessment is an important step in the classification process. The goal is to quantitatively determine how effectively pixels were grouped into the correct feature classes in the area under investigation.

A. Error Matrix

Now the classification accuracy of the proposed algorithm is calculated using the classification *error matrix* as shown in Table 3.3. The error matrix's interpretation along column suggests how many pixels are classified correctly by the algorithm. For example, in the first column, out of total 149 vegetation pixels, 142 pixels were correctly classified as vegetation, 2 vegetation pixels were classified wrongly as Barren, and 5 as Urban, a total of only 7 misclassified pixels. Also, the diagonal elements (diagonal elements indicate the no. of correctly classified pixels in that category) of Table 3.3 (Hybrid ACO2/PSO/BBO Classifier) vary drastically when compared to those in Table 3.1 (simple BBO Classifier).

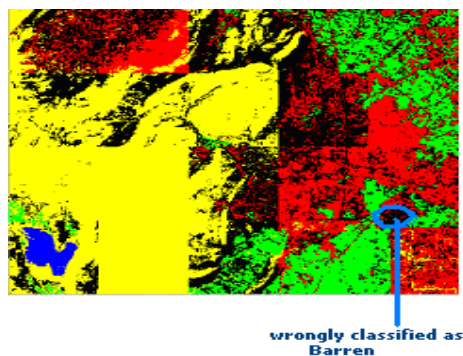


Figure 3.7: Classified image after applying BBO (Kappa Coefficient=0.6715)

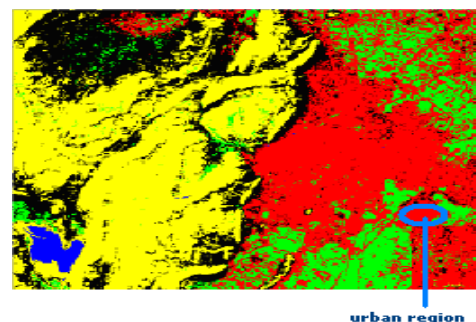


Figure 3.8: Hybrid ACO2/PSO/BBO classified image (Kappa Coefficient=0.98182)

For example, in Table 3.3, whole 190 out of 190 pixels were correctly classified as urban pixels whereas simple BBO Classifier in Table 3.1 could only classify 88 pixels correctly as urban pixels and it classified 91 pixels wrongly as barren ones. This indicates a drastic improvement in the accuracy of the results obtained by BBO through the use of the hybrid technique which is able to classify, in particular the urban and the water pixels, with almost 100% efficiency (with no omission errors), which was what was also reflected earlier, from the data distribution graph plotted. The error matrix for the hybrid ACO2/BBO classifier is also calculated and presented in Table 3.2 from which it is reflected that the proposed classifier is an improvement over the hybrid ACO2/BBO based intelligent classifier too.

Table 3.1: Error matrix when only BBO is applied (Kappa coefficient = 0.6715)

	Vegetation	Urban	Rocky	Water	Barren	Total
Vegetation	127	9	0	0	2	138
Urban	0	88	1	0	32	121
Rocky	6	2	176	1	17	202
Water	0	0	3	69	0	72
Barren	17	91	20	0	119	247
Total	150	190	200	70	170	780

Table 3.2: Error matrix when hybrid ACO2/BBO technique is applied. (Kappa Coefficient=0.96699)

	Vegetation	Urban	Rocky	Water	Barren	Total
Vegetation	145	0	0	0	0	145
Urban	2	188	0	0	9	199
Rocky	0	0	194	0	0	194
Water	1	0	3	70	0	74
Barren	1	2	2	0	157	162
Total	149	190	199	70	166	774

Table 3.3. Error matrix when hybrid ACO2/PSO/BBO technique is applied. (Kappa Coefficient=0.98182)

	Vegetation	Urban	Rocky	Water	Barren	Total
Vegetation	142	0	0	0	0	142
Urban	5	190	0	0	0	195
Rocky	0	0	198	0	3	201
Water	0	0	0	70	0	70
Barren	2	0	1	0	163	166
Total	149	190	199	70	166	774

B. Producer's Accuracy

Producer's accuracies (as shown in Table 3.4) indicates how well the training pixels of a given cover type are classified. The results in the Table 3.4 show that water and urban pixels have been extracted perfectly followed by rocky pixels which show 99% efficiency.

Table 3.4: Producer's Accuracy

Feature	Accuracy	Producer's
Vegetation	142/149	95.3%
Urban	190/190	100%
Rocky	198/199	99.5%
Water	70/70	100%
Barren	163/166	98.2%

C. User's Accuracy

User's accuracies (as shown in Table 3.5) indicate the probability that a pixel classified into a given category actually represents that category on the ground.

Table 3.5: User's Accuracy

Feature	Accuracy Calculation	User's Accuracy
Vegetation	142/142	100%
Urban	190/195	97%
Rocky	198/201	98.5%
Water	70/70	100%
Barren	163/166	98%

3.1.6. Classification comparison of Hybrid ACO2/PSO/BBO Classifier with the traditional probabilistic classifiers

Figure 3.9 compares the hybrid ACO2/PSO/BBO technique with the MDC & MLC. A comparison of the kappa coefficients is given in Table 3.6.

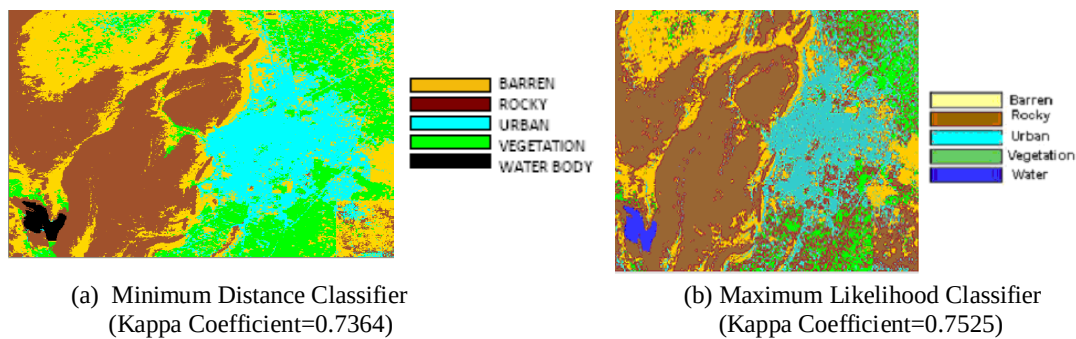


Figure 3.9: A comparison with the Traditional Probabilistic Classifiers

Table 3.6: A comparison of the Kappa coefficients of the Hybrid ACO2/PSO/BBO Classifier with the traditional classifiers.

Minimum Distance Classifier(MDC)	Maximum likelihood Classifier(MLC)	Biogeography Based Optimization (BBO)	Hybrid ACO2/PSO-BBO Classifier
0.7364	0.7525	0.6715	0.98182

3.1.7. Classification results of other computational intelligence based techniques used for satellite image classification

This section presents the results of classification of all the computational intelligence based classifiers developed, on the satellite image of Alwar region in Rajasthan, India.

From the discussion in section above, it is evident that the hybrid ACO2/PSO/BBO is a much efficient classifier as compared to the traditional classifiers such as the MDMC and MLC. However, this hybrid ACO2/PSO/BBO technique also produces comparable results with the other recent computational intelligence based classifiers. The satellite image classification results of the other recent CI based classifiers when applied on the Alwar region are shown below. Fig 3.10(a) shows the fuzzy classification of Alwar region which has a kappa –

coefficient of 0.9134 [Panchal, V. K., et. al., 2008]. Fig 3.10(b) presents the results of an integrated rough – fuzzy tie up approach which has a kappa coefficient of 0.9700 [Panchal, V. K., et. al., 2008]. Fig 3.10(c) applies the cAntMiner algorithm on the Alwar region which has a kappa coefficient of 0.964 [(Bansal, S., et. al., 2009)]. Fig 3.10 (d) shows the result of applying the hybrid ACO-BBO technique on the Alwar image which has a kappa-coefficient of 0.96699. Fig 3.10 (e) applies the hybrid ACO2/PSO classifier which has a kappa coefficient of 0.975 [Bansal, S., et. al., 2009]. Fig 3.10 (f) presents the results of the semantic web based classifier on the image with a kappa coefficient of 0.9881[Kumar, S., et. al., 2009].

It is also evident from the classification results of the various classifiers that the swarm intelligence based classifiers namely cAntMiner, hybrid ACO-BBO and hybrid ACO2/PSO are more efficient classifiers as compared to the human cognition based CI classifiers such as the rough sets, fuzzy sets and rough-fuzzy tie-up classifiers. This can be verified by considering a portion of the Alwar image classified by different image classification techniques. The portion that is considered for verifying the correctness of the above results is the area which is encircled. This area is chosen since it is already known that this area is an urban region from the fact that the proposed classifier classified this region with 100% producer's accuracy and 0% omission error which means that the proposed classifier has classified all the urban pixels correctly and no barren pixels have been misclassified as urban pixels and vice-versa as is reflected from Table 3.3.

Now, the results of classification of each of the classifiers are observed on the encircled region which is preliminarily known that it is an urban region. From the figures 3.10(a) and 3.10(b) below, it is reflected that the fuzzy and the rough-fuzzy tie-up classifiers are not able to extract the urban feature properly since the fuzzy classifier has classified the encircled region as barren which is actually an urban region. Similar is the case with the rough-fuzzy tie-up classifier too. Next, the results of the swarm intelligence based classifiers are observed. From the figure 3.10(c) which shows the results of the cAntMiner classification, it can be seen that the encircled region has been correctly classified as urban region. Figures 3.10(d) and 3.10(e) which show the results of hybrid ACO-BBO and hybrid ACO2/PSO classification also produce a correct classification of the encircled region. Figure 3.10(f) is not a swarm intelligence based classifier (added as a special instance of a computational intelligence based classifier), however, correctly classifies the encircled region.

Table 3.7 below compares the kappa coefficients of the recent CI based classifiers and the traditional probabilistic classifiers. From the table, it is clearly reflected that the presented CI based classifiers are much more refined & efficient than the probabilistic classifiers. It is also reflected that the swarm intelligence based classifiers are more efficient than the human cognition based classifier namely, rough sets and fuzzy sets. Simple BBO classifier is not a

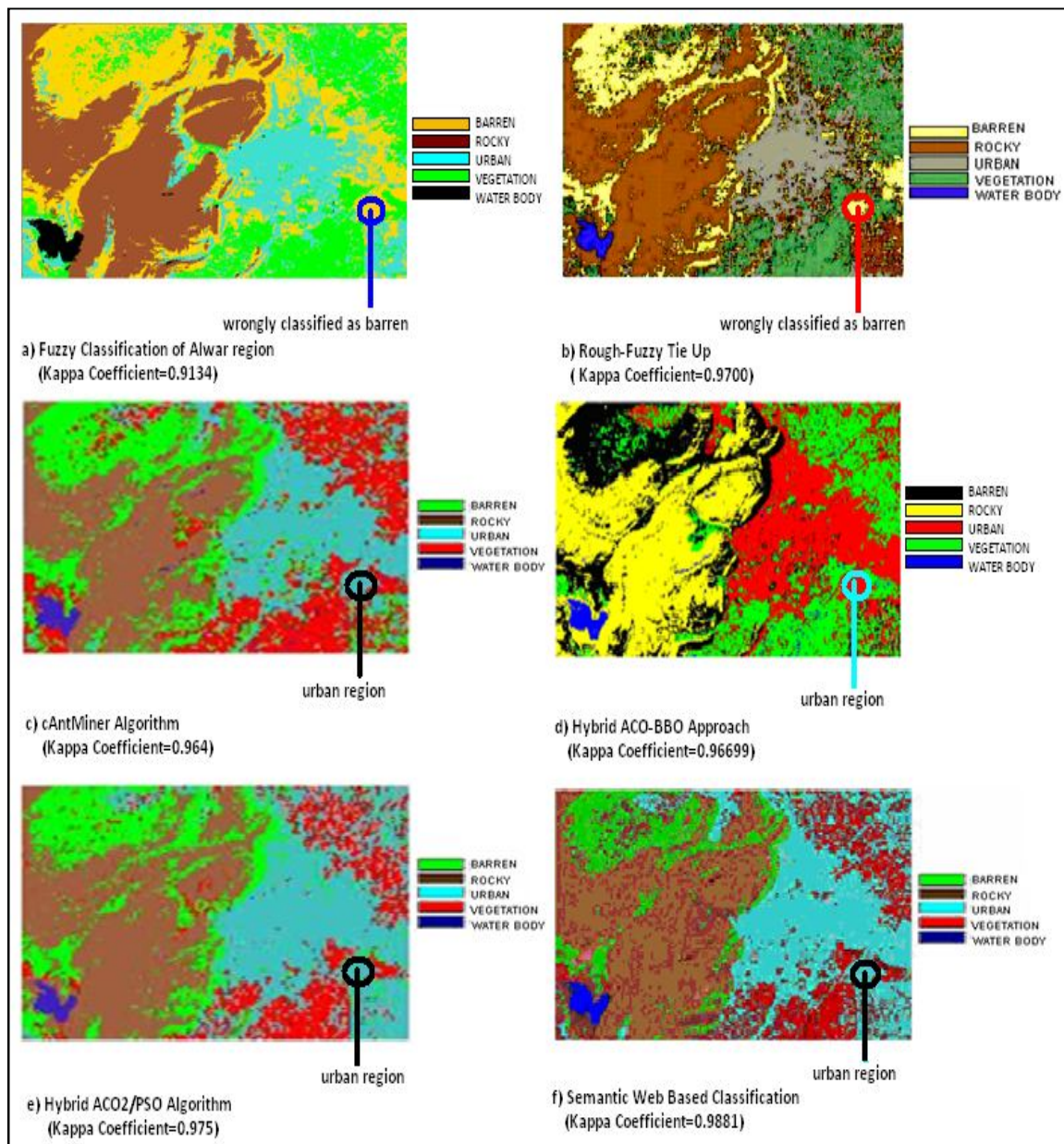


Figure 3.10: Classified images of Alwar region after applying various computational intelligence based techniques

Table 3.7: Kappa coefficient ($\hat{k}$) of recent computational intelligence based classifiers v/s probabilistic classifiers

Minimum Distance to Mean Classifier (MDMC)	Maximum Likelihood Classifier (MLC)	Fuzzy set	Rough-Fuzzy Tie up	cAnt-Miner	Hybrid ACO2/ PSO	Semantic Web Based Classifier	Biogeography Based Classifier	Hybrid ACO-BBO Classifier	Hybrid ACO2/ PSO/BBO Classifier
0.7364	0.7525	0.9134	0.9700	0.964	0.975	0.9881	0.6715	0.96699	0.98182

(Probabilistic classifiers)
Technology growth

(Other CI based classifiers)

completely swarm based classifier and its results are preliminary in nature since the approach is primarily developed for the purpose of demonstrating the application of BBO to a satellite image classification problem [Panchal, V.K., et. al., 2009] by modifying the original BBO

parameters and hence do not produce comparable results with the other swarm intelligence based classifiers demonstrated in the figure 3.10 above (its kappa coefficient being 0.6715). Also, it is reflected from the results of classification that the proposed hybrid classifier is the best known classifier from amongst all the present computational intelligence based classifiers, its performance being only slightly less than the semantic web based classifier.

3.2. Biogeography and Geo-Sciences Based Geo-spatial Feature Extraction

Enhanced versions of swarm intelligence techniques have been used for solving various engineering problems [(Peddle, D. R., et. al., 2001), (Zhou, G., et. al., 2008)]. For example, the improved PSO called as the immunity enhanced particle swarm optimization (IEPSO) for damage detection [Kang, F., et. al., 2012] has been developed wherein a damage detection method based on combined data of static and modal tests using particle swarm optimization (PSO) is discussed. To improve the performance of PSO, some immune properties such as selection, receptor editing and vaccination are introduced into the basic PSO and an improved PSO algorithm is formed. The BBO technique has been adapted and modified for economic load dispatch analysis [Bhattacharya, A., et. al., 2010] to solve both convex and non-convex economic load dispatch (ELD) problems of thermal plants involving constraints such as transmission losses, ramp rate limits, valve point loading, multi-fuel options and prohibited operating zones. The GA has been modified in the form of the real coded genetic algorithm for target sensing [Pappula, L., 2012] wherein an evolutionary technique of real coded genetic algorithm is applied to solve the system of linear equations.

This section demonstrates the evolution of a new computing paradigm [Alpaydin, E., 2004] for satellite image classification using the concept of entropy inspired from the geo-sciences phenomenon of mantle convection and dynamics of the earth and presenting the modified version of the original biogeography based land cover feature extraction technique [Eberhart, R., et. al., 2001]. The motivation behind the work was the absence of geo-sciences as a contributory domain to computational intelligence where domains like philosophy, psychology, bio-sciences, etc. have been used however, the above mentioned domain has not been explored. The developed hybrid (BBO-GS)/ACO2/PSO improves the hybrid bio-inspired classifier, the hybrid ACO2/PSO/BBO classifier presented in the previous section [(Goel, L., et. al., 2011A), (Goel, L., et. al., 2012A), (Goel, L., et. al., 2013C)] wherein the strengths of the ACO2/PSO technique were combined with the characteristic strengths of BBO. The proposed model uses surface entropy in DEM band as the measure for designing the fitness function. This entropy is the driving force leading to the formation of heterogeneous regions called as plates (texture analysis), similar to the convection forces in the mantle of the Earth. Hence, the classifier is able to classify the homogeneous regions (the regions with the lower entropy), more efficiently

than the other hybrid classifiers presented in the previous section [(Goel, L., et. al., 2010C), (Goel, L., et. al., 2012A)]. This also validates the model established in the last chapter [(Goel, L., et. al., 2011A), (Goel, L., et. al., 2013C)] which states that the classification efficiency of biogeography based extractor on a given land cover feature is proportional to the degree of entropy for that feature.

The section is organized into five sections, the first section illustrates the concepts of biogeography and geo-sciences. The second section describes the parameter settings, the hypotheses development and the proposed methodology of the biogeography and geo-sciences based feature extraction algorithm. The third section presents the implementation results and discussion for two different datasets. The fourth section presents the classification comparison of the results of the other recent computational intelligence based classifiers with the proposed classifier on a standard dataset to prove its effectiveness.

3.2.1 Basics of Geo-Sciences

This section presents the concepts of geo-sciences which can be adaptively applied for building the modified version of the original BBO for solving the problem of land cover feature extraction more efficiently [(Briggs, J. C., 1987), (Panchal, V. K., et. al., 2009)].

There are two underlying concepts which form the base of the geo-sciences phenomenon, the first being the dynamics of the movement of earth's plates and the second is the crust formation [(Currie, D. J., 2012), (Moresi, L., et. al., 2000), (Gornyi, V. I., 2002)]. It is considered that it is the entropy or the degree of disorder of the earth's mantle convection which is the driving force of the mechanism of plate dynamics. Figure 3.11 shows that the plate dynamics is considered as analogous to the clustering of similar pixels which is accomplished through texture analysis using rough sets in the proposed algorithm and resulting crust formation is considered as analogous to the presence of mixed pixels in an input plate. Hence, the underlying concepts of geo-sciences can be adapted to integrate with the original BBO technique for feature extraction in the satellite image. This work is an attempt to modify the original biogeography based feature extraction technique using an analogy between biogeography and geo-sciences phenomenon of earth's dynamics as shown in figure 3.12 [(Goel, L., et. al., 2011C), (Goel, L., et. al., 2013A)].

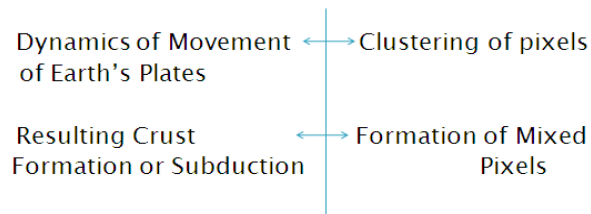


Figure 3.11: Adapting the geo-sciences phenomenon of earth's dynamics for land cover feature extraction

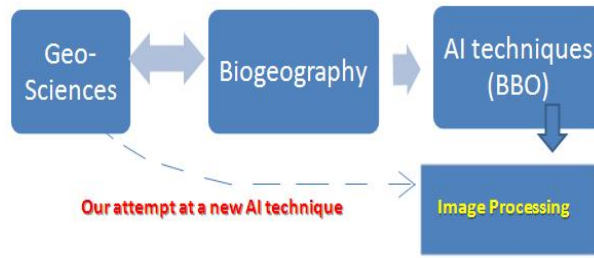


Figure 3.12: Attempt at a new AI technique

3.2.2. Proposed framework for the Biogeography and Geo-sciences Based Feature Extractor

This section discusses the parameter settings, the proposed methodology and the framework of biogeography based feature extraction technique inspired from the geo-sciences concepts.

A. Parameter settings and procedures

This section identifies some important parameter settings and procedures adopted [(Kang, F., et. al., 2011), (Kang, F., et. al., 2013A), (Kang, F., et. al., 2013B)] which comprise the preprocessing steps of the biogeography and geo-sciences based land cover feature extraction algorithm [(Goel, L., et. al., 2012A), (Bansal, S., et. al., 2009)]. For a better understanding of the definitions discussed, these are discussed with reference to the dataset of Alwar image for demonstration and pointers are included to the corresponding settings for the second dataset of Patalganga.

- Plates $plate(j)$ refer to the image 'I' consisting of a set of plates which correspond to the image pixels. The 7-band satellite image of Alwar region of size 472 x 546 is taken as an input image for biogeography and geo-sciences based feature extraction. The image has 2,57,712 pixels and each pixel is of 25m by 25m spatial resolution. For Patalganga image, a 4-band image of size 508 × 744 is taken as input which has about 3,77, 952 pixels.
- An eco-system is comprised of a universal habitat which hosts all the individual plates generated through texture analysis. The plates in the feature habitat are nothing but the labeled dataset/training dataset generated by the expert. Five types of feature plates are considered here for Alwar region, namely: water, vegetation, rocky, urban and barren. For Patalganga dataset, only 3 feature plates are considered that of vegetation, snow and rocky. Therefore, the ecosystem H_6 is a group of 6 habitats (one universal habitat and five Feature habitats). In case of Patalganga, the ecosystem H_4 will be a group of 4 habitats.
- Each of the multispectral bands of the image represents one SIV of the habitat. Further, image in each band is a gray image; therefore, $SIV \in C$ is an integer and $C \subseteq [0,255]$.
- A habitat $H \in SIV^m$ where, m is 7 = total image bands. For Patalganga image, m = 4 bands.
- The image is subjected to unsupervised clustering using the concept of rough sets thus generating equivalence classes or plates [Ohm, A., et. al., 1997]. The preference to the rough sets is given because the equivalence classes generate clusters based on the criteria of

indiscernibility and also, the number of clusters required are not needed to be known beforehand [(Ohm, A., et. al., 1997), (Pawlak, Z., 1982), (Pawlak, Z., 1998)]. In the resulting data, digital number value of each pixel is replaced by the discretized interval. Each row in the resulting data represents the x, y coordinates and the digital number values in the red, green, NIR, MIR, RS1, RS2 and DEM bands respectively of each pixel in the image (for patalganga image, the bands are red, green, NIR and MIR bands). This discretized data is further partitioned on the basis of NIR and MIR band which contain a good amount of geo-spatial information [(Lillesand, T. M., et. al., 2008), (Panchal, V. K., et. al., 2008)] to obtain the final classes.

- f. Inspired from the concepts of geo-sciences, the surface entropy of the plates calculated in the DEM band is used as the basis of calculating the habitat suitability index. For the calculation of entropy (ϕ), the procedure followed is described below [(Goel, L., et. al., 2011A), (Goel, L., et. al., 2013C)]. The pixels of the input plate in H_u for which the entropy has to be calculated, are copied in an excel sheet and then the digital number values in the DEM band are divided into 'm' intervals. Next, the no. of pixels of the entire plate $plate(j)$ falling in each of the 'm' ranges obtained above are calculated and hence, the entropy of the particular plate, $\phi(plate(j))$ is calculated by using the equation below.

$$Entropy(\phi(plate(j))) = - \sum_{k=1}^m P_k \log (P_k) \quad (3.4)$$

where, $P_k = m_k / N$ where m_k is the no. of pixels in the k^{th} interval and N is the total no. of pixels in the input plate $plate(j)$.

- g. Number of similarity classes 'c' formed are equal to the no. of land cover feature plates $feature(k)$ for each dataset. Since the maximum value of $k = 5$ and 3 for Alwar and Patalganga datasets respectively, c is assigned the values of 5 and 3. Thus, if 'count(sim(k))' is the no. of elements in the similarity class corresponding to feature k , and n is the total number of plates in H_u , then, the following equation holds,

$$\bigcup_{k=1}^d \bigcup_{l=1}^{count(sim(k))} plate(l) = \bigcup_{j=1}^n plate(j) \quad (3.5)$$

- h. A significance level (S_l) of 0.1 i.e. $-0.05 \leq S_l \leq +0.05$ is assigned. This is the value within which the plate similarity factor should lie so that the two plates can be categorized as similar plates leading to the formation of similarity classes. After careful observations, this value was chosen as the measure based upon which one can optimally categorize two plates as similar.
- i. The heuristic function makes use of a specific threshold value. A standard threshold value δ , where δ ranges from -0.05 to $+0.05$, is assigned for classification of the plates in each similarity class. The classification is done by comparing the HSI (plate similarity factor with

the feature plates) with this value defined for classifying an input plate into the feature 'k'. This number can be increased or decreased depending upon the precision in the classification results required. If the threshold value is decreased, a larger number of iterations will be required to completely classify the image, however, the results will be more accurate.

B. BBO-GS Methodology

This section presents the proposed hypotheses for classification based on biogeography and geo-sciences [(Williams, K. J., et. al., 2012), (Zimmermann, N. E., et. al., 2010), (Solomon, Z., et. al., 2008)]. The main steps of the biogeography and geo-sciences inspired feature extraction are presented in figure 3.13 and explained below. Alwar area in Rajasthan, illustrated in figure 3.14, is taken as a case study to develop the hypothesis, that may be generalized in future developments. The area is selected as it contains hilly track, water, agriculture land, urban zone, barren lands.

Step I. Remote Sensing images, the actual data on which work is done, is reflectance values of solar radiation of the surface objects [(Accad, A., et. al., 2009), (Peddle, D. R., et. al., 2001), (Ying, L., et. al., 2009)]. This phenomenon may be represented through Texture analysis.

- ✓ *Generate individual plates through texture analysis .*
- ✓ *In the initial state of the ecosystem, consider each of these individual plates as a member of the Universal habitat.*
- ✓ *Represent the surface entropy through the Digital Elevation Model (DEM).*
- ✓ *Calculate the surface entropy in the DEM band of each of the plates of the Universal habitat.*
- ✓ *Generate the similarity classes 'c' for each of the land cover feature plates 'k' by comparing the plate similarity factor with the significance level 'Sf' for each plate.*
- ✓ *Verify the formation of similarity classes by comparing the average entropy of the similarity classes 'c' and the entropy of each of the land cover features 'k'.*
- ✓ *Proceed for classification of the plates in each similarity class by comparing the HSI (plate similarity factor) with the standard threshold value ' δ ' defined for classifying an input plate into the feature 'k'.*
- ✓ *Initialize a new ecosystem for the classification of the remaining unclassified plates in the universal habitat and repeat the above steps.*

Figure 3.13: Summary of the steps followed for developing the hypotheses for biogeography and geo-sciences based feature extraction

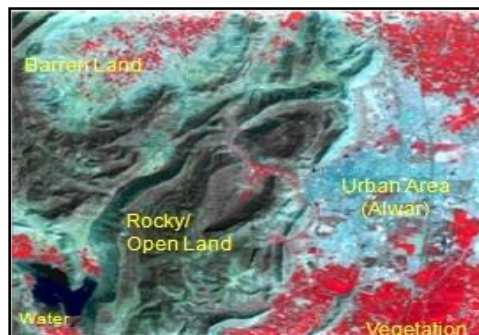


Figure 3.14: Surface topography demarcates the different land use types in the Alwar image.

Step II. Generate heterogeneous regions called plates through texture analysis of the region under consideration. Rough sets theory is used for texture analysis (unsupervised clustering).

Step III. In the initial state of the ecosystem required to initialize the geo-sciences and biogeography based classifier, each of the individual plates obtained through texture analysis are considered as a member of the universal habitat.

Step IV. The surface topography well represents visible perceptions of the plate concepts. Surface topography is an indirect manifestation of the plate tectonic mechanism. This may be represented nicely through Digital Elevation Model (DEM) or slope map or hill shading.

Step V. Based on this, different areas may be delineated. DEM & hill shading of Alwar region are shown below in figure 3.15.

Step VI. The major assumption is that it is the entropy which is the driving force of these regions, similar to the convection force in the mantle of the Earth.

Step VII. Hence, the surface entropy in the DEM band of each of the plates of the Universal habitat is used to generate the clusters of similar plates. For the purpose, the differences in the surface entropies in DEM band of each of the plates in the universal habitat are compared with the feature habitat plates by migration. This value is called the plate similarity factor. The feature habitat contains the plates representing the training data set of human expert's knowledge used for supervised clustering for each of the land cover features that need to be extracted. Hence, the input plate is categorized as similar to a feature habitat plate or homogeneous if their plate similarity factor lies below the significance level ' S_f ' else they are considered as heterogeneous plates. The significance level marks a threshold for identifying two plates as similar. Hence, a similarity class ' c ' (where ' c ' ranges from 1 to ' d ') is obtained containing the group of similar plates for each of the land cover feature plates ' k ' where ' k ' ranges from 1 to ' d '.

For verification, the average of entropy in the DEM band for the similarity class ' c ' corresponding to the land cover feature plate ' k ' containing the group of similar plates is calculated. Also, the entropy of the training data set in DEM band for each of the land cover feature plates ' k ' is calculated. Next, the difference in the average of entropy of the similarity class ' c ' with the entropy of each of the land cover feature plates ranging from 1 to ' d ' is calculated. This difference in entropy will be the minimum and within the significance level for the feature ' k ' to which the class ' c ' corresponds to.

Step VIII. Once similarity classes ' c ' for each of the land cover features ' k ' are obtained, the next step is to proceed for the classification of the plates by taking into account the standard threshold ' δ ' defined for classifying an input plate into the feature ' k '.

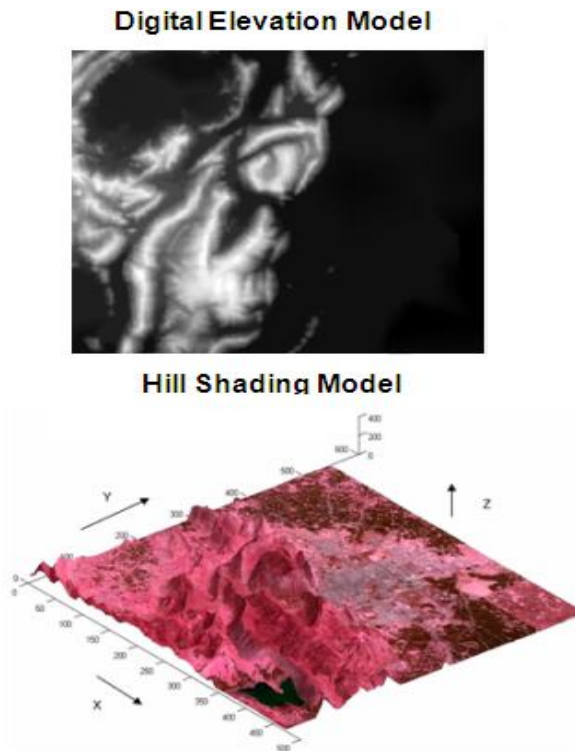


Figure 3.15: DEM & Hill Shading of Alwar

Each of the input plates in the similarity class ‘c’ are classified by migrating the input plate from the universal habitat to the feature habitat and calculating the HSI [(Simon, D., 2008), (Simon, D., et. al., 2009)] after migration. The HSI is taken as the plate similarity factor calculated by finding the difference in the entropy of the input plate and the entropy of the feature habitat or the land cover feature plate ‘k’. If the HSI calculated is within the threshold ‘ δ ’, then, the input plate in the similarity class ‘c’ is classified into the feature ‘k’ otherwise, the input plate is further split which remained in the universal habitat (since it could not be classified) through texture analysis. It can also be noted that using entropy as the basis of the HSI function will improve the classification accuracy of the original biogeography based land cover feature extractor which uses ‘deviation from mean metric’ as the basis of the HSI function as was demonstrated in the analytical study of the performance governing factors of biogeography based feature extractor that was carried out in chapter 2 [(Goel, L., et. al., 2011A), (Goel, L., et. al., 2013C)].

Step IX. Also, a new ecosystem is then initialized which follows a similar procedure for a more refined classification of the remaining unclassified plates in the universal habitat.

C. Functional architecture

The proposed functional framework for the biogeography and geo-sciences based feature extractor is presented in the form of algorithm in figure 3.16 below and explained next. The input to the algorithm is the multi-spectral satellite image.

- a. The algorithm begins by the initialization of the ecosystem which generates the input plates to initialize the biogeography and geo-sciences based classifier. The input plates are generated by means of texture analysis wherein the rough sets part of the biogeography based land cover feature extraction algorithm is used [Panchal, V. K., et. al., 2009]. The rough sets generator uses multiple bands of the image (which represent the texture of each of the land cover features in the image clearly, hence, clustering is based on texture analysis) during the discretization and partitioning step [(Goel, L., et. al., 2012A), (Panchal, V. K., et. al., 2009)] to generate the unsupervised clusters called the equivalence classes.
- b. In the initial state of the ecosystem required to initialize the biogeography and geo-sciences based classifier, each of these equivalence classes is considered as individual plates, $plate(j)$, each of which is considered as a member of the universal habitat, H_u , where j ranges from 1 to n where n is the total no. of members in H_u . Depending on the application, in other words depending on which feature that is required to be extracted from the image most efficiently, the band for partitioning can be chosen as described in [(Panchal, V. K., et. al., 2008), (Panchal, V. K., et. al., 2009)]. For the illustration, the bands chosen are the NIR and MIR bands of the 7- band image since the water pixels are required to be extracted effectively and the water body in the image needs to be clearly identified and these are the bands in which the water feature is particularly more highlighted and best viewed. Therefore, the NIR and the MIR bands are used for discretization and partitioning step in the semi-naïve algorithm used for creating rough set equivalence classes for each of the clusters. Each of these resultant classes are put in the universal habitat.
- c. Since it is known that the surface topography is an indirect manifestation of the geo-sciences phenomenon of plate tectonics, it can be represented nicely through the Digital Elevation Model (DEM) or *slope map* or *hill shading model*. Using entropy as the basis of the HSI function will improve the classification accuracy of the original biogeography based land cover feature extractor as was demonstrated in the analytical study of the performance governing factors of biogeography based feature extractor that was carried out in chapter 2 [(Goel, L., et. al., 2011A), (Goel, L., et. al., 2013C)].
For the calculation of entropy, the procedure followed is described in 3.2.2 (A) above. The procedure followed for calculation of the entropies of each of the land cover features is the same. Based on the results, each of the input plates of H_u is delineated into homogeneous and heterogeneous plates using the concept of *entropy*(φ), similar to the convection forces in the mantle of the earth.
- d. Next, the set of similarity classes c is generated where c ranges from 1 to d and d is the no. of similarity classes obtained (equal to the no. of land cover features).

```

While (no. of unclassified plates in  $H_u \neq \text{NULL}$  )
{
  a. Generate individual plates,  $\text{plate}(j)$ , through texture analysis.
  b. Consider each of these individual plates,  $\sum_{j=1}^n \text{plate}(j)$  as a member of the universal
     habitat,  $H_u$ . //  $n$  is the total no. of plates in  $H_u$ 
  c. Represent the surface entropy,  $\varphi(\text{plate}(j))$ , through DEM.
  d. Calculate  $\varphi(\text{plate}(j))$  in the DEM band of each of the plates of  $H_u$ .
  e. Generate the clusters of similar plates,  $\text{sim}(k)$  for each land cover feature called the
     similarity classes  $c$  by comparing the plate similarity factor  $S_f$ , with the significance
     level  $S_l$  for each plate in  $H_u$ .
     //plate similarity factor of two plates  $a$  and  $b$ ,  $S_f(a, b) = |\varphi(\text{plate}(a)) - \varphi(\text{plate}(b))|$ 
     For each plate  $j$  in  $H_u$ 
     Migrate  $\text{plate}(j)$  to each of the feature plates( $k$ ) in  $H_f$ 
     /*  $k$  ranges from 1 -  $d$  and  $d$  is the no. of land cover features to be extracted */
     If  $|(S_f(\text{plate}(j), \text{feature}(k)) \leq S_l|$ 
      $\text{similar plates}(k) = \{\text{similar plates}(k), \text{plate}(j)\};$ 
     End For
  f. Verify the similarity classes formed by using the following inequality:
     
$$\sum_{k=1}^d \frac{\text{count}(\text{sim}(k))}{\text{count}(\text{sim}(k))} (\varphi(\text{plate}(l)) - \varphi(\text{feature}(k)) < S_f$$

     for each  $c$  corresponding to feature  $k$ .
  g. // Proceed for final classification step.
     For each similarity class  $c$  in  $H_u$  corresponding to feature  $k$ 
     For each  $\text{plate}(l)$  in  $c$  //  $l$  ranges from 1 to  $\text{count}(\text{sim}(k))$ .
     
$$HSI = |\varphi(\text{plate}(l)) - \varphi(\text{feature}(k))|$$

     If  $(HSI < \delta)$ 
      $\text{plate}(l) = \text{feature}(k); H_u = H_u - \text{plate}(l);$ 
     End For
     End For
  h. Initialize a new ecosystem for classification of the remaining unclassified plates in  $H_u$ .
} //  $H_u$  is empty since no unclassified plates remain in  $H_u$ .

```

Figure 3.16: Biogeography and Geo-sciences based land Cover Feature Extraction

- i. All the input plates are kept in the universal habitat where the no. of species (or input plates) in the universal habitat is n . Let $\phi(\text{plate}(j))$ represent entropy of each of the input plates, $\text{plate}(j)$, of H_u in the DEM band.
- ii. A plate similarity factor, S_f is assigned to each plate based on the difference between the entropy in the DEM band of the input plate $\text{plate}(j)$ in the universal habitat H_u and the entropy in DEM band of the land cover feature plate $\text{feature}(k)$ in the feature habitat H_f to which the input plate is migrated to.
- iii. Feature habitat H_f is the training data set representing expert's knowledge and used for supervised clustering of these plates. A total of k land cover features are considered where k ranges from 1 to d which are considered as the feature habitat. Here, 5 feature habitats are available each for water, urban, rocky, barren and vegetation features, having members produced by experts.
- iv. Hence, plate similarity factor, ' S_f ' = $|\phi(\text{plate}(j)) - \phi(\text{feature}(k))|$. A significance level is assigned in order to infer if two plates are similar or different. A significance level (S_l) of 0.1 i.e. $-0.05 \leq S_l \leq +0.05$ is assigned.
- v. Next, the clusters of similar plates called the similarity classes c are generated

$$\bigcup_{l=1}^{\text{count}(\text{sim}(k))} \text{plate}(l)$$

where, $\text{sim}(k)$ is the set of plates similar to $\text{feature}(k)$ by comparing the plate similarity factor S_f , with the significance level S_l for each plate in H_u . For each $\text{plate}(j)$ in H_u , the $\text{plate}(j)$ is migrated to each of the feature plates $\text{feature}(k)$ in H_f where k ranges from 1 to d . Now, the plate similarity factor value after migration of $\text{plate}(j)$ to $\text{feature}(k)$ is calculated. If S_f or surface entropy $|\phi(\text{plate}(j)) - \phi(\text{feature}(k))| \leq S_l$, it represents an insignificant difference and thus a homogeneous region (similar plate) else a heterogeneous region (different plate).

- vi. Here, all the similar plates to $\text{feature}(k)$ are grouped in one similarity class leading to a set of similarity classes c .
- e. The similarity classes formed are hence verified.
 - i. For the purpose, the procedure starts by taking the average of the entropies in the DEM band for each of the similarity classes c corresponding to the land cover feature plate k containing the group of similar plates, according to

$$\text{Avg}_c = \frac{d}{k} \left(\frac{1}{\text{count}(\text{sim}(k))} \sum_{l=1}^{\text{count}(\text{sim}(k))} \phi(\text{plate}(l)) \right) \quad (3.6)$$

where ' $\text{count}(\text{sim}(k))$ ' is the no. of elements in the similarity class corresponding to feature k .

- ii. Next, the difference in the average entropy of the similarity class c corresponding to feature k with the entropy of each of the land cover feature plates k i.e.

$$\sum_{k=1}^d \sum_{i=1}^{count(sim(k))} \frac{1}{count(sim(k))} (\varphi(plate(l)) - \varphi(feature(k)) < S_f \quad (3.7)$$

for each c corresponding to feature k is calculated.

- iii. If this difference in entropy is the minimum and within the significance level for the feature k , the plate is verified to belong to the class c to which it corresponds to.

- f. Next, the algorithm proceeds with the final classification step taking into account the standard threshold value δ for classifying an input plate into the feature (k).

- i. Each of the input plates is classified in the similarity class c by migrating the input plate from the universal habitat to the feature habitat and calculating the *HSI* [(Simon, D., 2008), (Simon, D., et. al., 2009), (Currie, D. J., 2012)] after migration. δ is the standard value against which the HSI of all the input plates will be compared. HSI is taken as the plate similarity factor. For each of the input plates in the similarity class c corresponding to feature k , the difference HSI =

$$\sum_{k=1}^d \left| \left(\sum_{l=1}^{count(sim(k))} \varphi(plate(l)) - \varphi(feature(k)) \right) \right| \quad (3.8)$$

is calculated where, k is the feature to which the similarity class c is mapped to, and c is the similarity class to which plate(l) belongs to.

- ii. If ($HSI < \delta$), then the input plate plate(l) in the similarity class c is classified into the feature(k), else the input plate plate(l) (which remained in the universal habitat since it could not be classified) is further split through texture analysis as described above and the above procedure is repeated for the input plate l left in the universal habitat.

- g. The algorithm initializes a new ecosystem for the classification of the remaining unclassified plates in the universal habitat H_u and repeats the above procedure.

D. Mathematical formulation of biogeography and geo-sciences based feature extractor

Therefore, the working of the algorithm can be summarized in the form of the following equation where HSI is analogous to fitness function in conventional algorithms (figure 3.17).

$$\bigcup_{k=1}^d \bigcup_{l=1}^{count(sim(k))} \left(\min \left(\sum_{k=1}^d \left| \underbrace{\varphi(plate(l))}_{\text{Universal Habitat}} - \underbrace{\varphi(feature(k))}_{\text{Feature Habitat}} \right| \right) \right) < Threshold(\delta)_{-0.05 < t < +0.05}$$

Iterate the whole process until no unclassified plate is left in H_u

Figure 3.17: Mathematical formulation of the biogeography and geo-sciences based feature extraction algorithm

$$sim(k) = \{sim(k), plate(j)\},$$

If $| (S_f(plate(j), feature(k)) \leq S_t |$ and j ranges from 1 to the number of plates n in H_u .

For the biogeography and geo-sciences based feature extractor, the algorithm proceeds in the following manner i.e. for the DEM band, the difference in the surface entropy of each input plate of the similarity class of the universal habitat is calculated with each feature plate of the feature habitat containing the expert generated training set of each land cover feature of the image. If this difference is the minimum for the feature 'k' and also less than the pre-specified threshold value of $-0.05 < t < +0.05$, then that particular plate is classified as the feature 'k' else it is left as unclassified. The process is repeated for each feature's similarity class plate until there is no similarity class left in the universal habitat and the whole process is iterated till there is no unclassified plate left.

3.2.3. Results and Analysis

This section presents the results obtained after applying the biogeography and geo-sciences based land cover feature extractor on two different datasets for verification of the results. The datasets chosen are of Alwar region in Rajasthan and Patalganga region in Shivalik ranges. The results and analysis for each of the case studies are described in the sections below.

A. Case study: Alwar region in Rajasthan

This section demonstrates the applicability of the proposed classifier by applying it on the Alwar dataset.

Table 3.8 represents the calculations of the entropies in the DEM band for each plate (equivalence class) obtained through texture analysis on the Alwar image. The image is subjected to 'n' (here, n is taken as $n = 20$ however this number can be varied to any value depending upon the image size) simple partitions, for the sake of simplicity. The calculations are represented in a tabular form which summarizes the DEM entropies calculated for each of the equivalence class in each of the 'n' partitions. The plates are categorized into similarity classes for each land cover feature based on the entropies calculated in the DEM band with the significance level ' S_t ' of -0.05 to $+0.05$. Next, the algorithm proceeds for the classification of each of the input plate(j) by taking into account the standard threshold ' δ ' of -0.05 to $+0.05$ that is defined for classifying an input plate into the feature 'k'. The feature entropies are calculated preliminarily and summarized in Table 3.8. In this case, the plates get classified in the first iteration only, however, if the threshold is decreased, more iterations will be required to fully classify all the plates in the universal habitat as demonstrated later in this section. The above HSI calculations and the feature classification can also be deduced graphically by means of the graph shown in figure 3.18 which shows the feature entropy v/s the plate entropy plotted for each plate in the DEM band for each partition of the image

(referred to as class in figure 3.18). This graph plot presents four graphs one, plotting each input plate(j)'s entropy in DEM band in ' H_u ', second, plotting the entropy of the land cover feature 'k' to which the plate(j) belongs to, in the DEM band, and, third and fourth graphs are the ' $-\delta$ ' and ' $+\delta$ ' threshold graphs representing the area within which the feature entropy of feature plate 'k' should lie for the input plate(j) to be classified in the feature 'k'. The blue and the red lines in the graph represent the graph plot of the entropies of the input plates and the feature entropies respectively. The green and the purple lines represent the threshold interval ' $-\delta$ ' and ' $+\delta$ ' respectively in the graph plotted. For the plates to be classified into the corresponding mapped land cover features, they should lie within this interval. The classified image is obtained in figure 3.19 which clearly shows the extraction of land cover features of water, urban, rocky, vegetation and barren. The yellow color represents rocky area, green color represents vegetation area, black color represents barren area and red color represents the urban area. The kappa coefficient for the classified image is obtained as 0.7440 (explained in later sections) whereas the classified image obtained through the application of original biogeography based land cover feature extractor had a kappa coefficient of 0.6715 [(Goel, L., 2010B), (Goel, L., et. al., 2011B), (Panchal, V. K., et. al., 2009)]. Comparing the results, it can be seen that the encircled region has been correctly classified as urban region by the geo-sciences based classifier (which was wrongly classified by the original BBO as barren [(Accad, A., et. al., 2009), (Franklin, J., 1995), (Goel, L., et. al., 2012A)]).

Next, the effect of decreasing the threshold is discussed. The threshold is decreased to 0.025 to +0.025. The section also presents the refined results obtained after applying multiple iterations of the biogeography and geo-sciences based optimization technique for land cover feature extraction. For the demonstration simplicity, a part of the Alwar image is considered. Next, the algorithmic procedure for running multiple iterations is presented by demonstrating the incremental extraction of water body in the Alwar image. Table 3.9 presents the entropies calculated in the DEM band for each plate (equivalence class) obtained through texture analysis on the part of the image from which water body has to be extracted. The table summarizes the DEM entropies calculated for every iteration, however, the calculations in each iteration are described in detail next.

From Table 3.9, it can be observed that eight plates are obtained through texture analysis initially. These plates are then categorized into similarity classes for each land cover feature based on the entropies calculated in the DEM band with the significance level ' S_f ' of -0.05 to +0.05 as shown in Table 3.10. The similarity class formation is verified by calculating the difference in the average entropy of the similarity class in DEM band and the entropy of the

land cover feature to which this similarity class corresponds to and observing that this difference is within the significance level of 0.05 as shown in Table 3.11.

Next, the algorithm proceeds for the classification of each of the input plate(j) by taking into account the standard threshold ' δ ' of -0.025 to +0.025 that is defined for classifying an input plate into the feature 'k'. For each of the similarity class 'c', each of the input plate 'j' in the similarity class 'c' is processed by calculating the HSI value. If the HSI value is within ' δ ', then the input plate 'j' in the similarity class 'c' is classified into the feature 'k' else the input plate 'j' is further split through texture analysis and the above steps are repeated for the input plate 'j' in the next iteration. The above calculations are summarized in Table 3.12. The feature entropies were calculated preliminarily and summarized in Table 3.11. The above HSI calculations and the feature classification can also be deduced graphically by means of the graph shown in figure 3.20 which shows the feature entropy v/s the plate entropy plotted for each plate in the DEM band for each iteration. The color codes are as given in figure 3.20. The plates for which the entropies do not fall in the threshold interval are the plates which cannot be classified into the corresponding mapped land cover features and hence, are left unclassified (since these may contain mixed pixels [(Gupta, N., et. al., 2011B), (Sinha, S., et. al., 2012), (Varshney, T., et. al., 2014)] and hence, need to be further broken down into smaller plates) and considered for further iterations. Now, the plates which could not be classified are further split and the procedure followed for the first iteration is repeated. The calculations for the remaining iterations which lead to the classified image clearly showing the iterative extraction of water body in the Alwar image are shown in Table 3.13.

The classified image is obtained in figure 3.21 which clearly shows the incremental extraction of water body with each iteration. The white color in the image represents the unclassified pixels of image in the first iteration. It can be observed that no water body (represented in blue color) is extracted in the first iteration (figure 3.21 (a)) and water pixels remain unclassified (represented in white color). With the increasing iteration (figure 3.21 (b), 3.21 (c), 3.21 (d), 3.21 (e)), water body is clearly extracted at the end of fifth iteration along with the other land cover features present in the part of the Alwar image considered for water body extraction. Now the algorithm proceeds to calculate the classification accuracy of the proposed algorithm using the classification *error matrix* as shown in Table 3.14. For example, in the first column, out of total 150 vegetation pixels, 142 pixels were correctly classified as vegetation and only 3 and 5 pixels were wrongly classified as urban and barren respectively, a total of only 8 misclassified pixels which is much better than the original BBO

Table 3.8: HSI calculations and the corresponding classified feature for the first iteration

Similarity Class for 'Water' Feature Entropy=0.3486		Similarity Class for 'Rocky' Feature Entropy=0.9316		Similarity Class for 'Barren' Feature Entropy=0.7249		Similarity Class for 'Vegetation' Feature Entropy=0.4619		Similarity Class for 'Urban' Feature Entropy=0.5631	
Plate Entropy	HSI	Plate Entropy	HSI	Plate Entropy	HSI	Plate Entropy	HIS	Plate Entropy	HIS
0.3886	0.0400	0.9319	0.0003	0.7375	0.0126	0.4671	0.0052	0.5405	0.0226
		0.8944	0.0372	0.7040	0.0209	0.4474	0.0145	0.5708	0.0077
		0.9332	0.0016	0.7597	0.0348	0.4736	0.0117	0.5706	0.0075
		0.8996	0.032	0.7693	0.0444	0.4651	0.0032	0.6028	0.0397
		0.9038	0.0278	0.6938	0.0311	0.4155	0.0464	0.5552	0.0079
		0.9616	0.03	0.7701	0.0452	0.4932	0.0313	0.5663	0.0032
		0.9619	0.0303	0.7418	0.0169	0.4749	0.013	0.5132	0.0499
		0.9191	0.0125	0.7039	0.021	0.4348	0.0271	0.5979	0.0348
		0.9840	0.0524	0.7248	1E-04	0.4640	0.0021	0.5305	0.0326
		0.9784	0.0468	0.7428	0.0179	0.4829	0.021	0.6037	0.0406
		0.9747	0.0431	0.7000	0.0249	0.4513	0.0106	0.5364	0.0267
		0.8981	0.0335	0.7526	0.0277	0.4156	0.0463	0.5610	0.0021
		0.8867	0.0449	0.6945	0.0304	0.4168	0.0451	0.5913	0.0282
		0.8890	0.0426	0.7424	0.0175	0.5115	0.0496	0.5438	0.0193
		0.9422	0.0106	0.7164	0.0085	0.4485	0.0134	0.5493	0.0138
		0.9481	0.0165	0.7218	0.0031	0.4137	0.0482	0.5249	0.0382
		0.9330	0.0014	0.7673	0.0424	0.4123	0.0496	0.5408	0.0223
		0.8861	0.0455	0.7131	0.0118	0.4130	0.0489	0.5554	0.0077
		0.8967	0.0349	0.6762	0.0487	0.4255	0.0364	0.5623	0.0008
		0.8848	0.0468	0.6930	0.0319	0.4168	0.0451	0.5109	0.0522
		0.9086	0.023	0.6798	0.0451	0.4142	0.0477	0.5287	0.0344
		0.8852	0.0464	0.7590	0.0341	0.4722	0.0103	0.5573	0.0058
		0.9185	0.0131	0.6812	0.0437	0.4628	0.0009	0.5223	0.0408
		0.9459	0.0143	0.7699	0.045	0.4181	0.0438	0.5873	0.0242
		0.9541	0.0225	0.7743	0.0494	0.4325	0.0294	0.5139	0.0498
		0.9331	0.0015	0.7671	0.0422	0.4314	0.0305	0.5257	0.0374
		0.8913	0.0403	0.7663	0.0414	0.4750	0.0131	0.5543	0.0088
		0.9668	0.0352	0.7399	0.015	0.5109	0.049	0.5437	0.0194
		0.9096	0.0220	0.7255	0.0006	0.4402	0.0217		
		0.9194	0.0122	0.7847	0.0598	0.5119	0.05		
		0.9374	0.0058	0.7377	0.0128	0.4319	0.03		
		0.9532	0.0216	0.6814	0.0435	0.4728	0.0109		
		0.9371	0.0055	0.7473	0.0224	0.4318	0.0301		
		0.9767	0.0451	0.6884	0.0365	0.4365	0.0254		
		0.9309	0.0007	0.6863	0.0386	0.4107	0.0512		
		0.9098	0.0218	0.6911	0.0338	0.4944	0.0325		
		0.9613	0.0297	0.7659	0.041	0.4872	0.0253		
		0.8855	0.0461	0.7570	0.0321	0.4531	0.0088		
		0.9812	0.0496	0.7647	0.0398	0.4317	0.0302		
				0.7539	0.029	0.4873	0.0254		
				0.6749	0.05	0.4819	0.02		
				0.7052	0.0197				
				0.7497	0.0248				
				0.7068	0.0181				
				0.7000	0.0249				
				0.7378	0.0129				
				0.7245	0.0004				
				0.7034	0.0215				
				0.7378	0.0129				
				0.6872	0.0377				
				0.7231	0.0018				
				0.7621	0.0372				
				0.7437	0.0188				
				0.7637	0.0388				
				0.6918	0.0331				
				0.7694	0.0445				
				0.6931	0.0318				

Feature Entropy v/s Input Plate Entropy for each plate in Classes I-XX

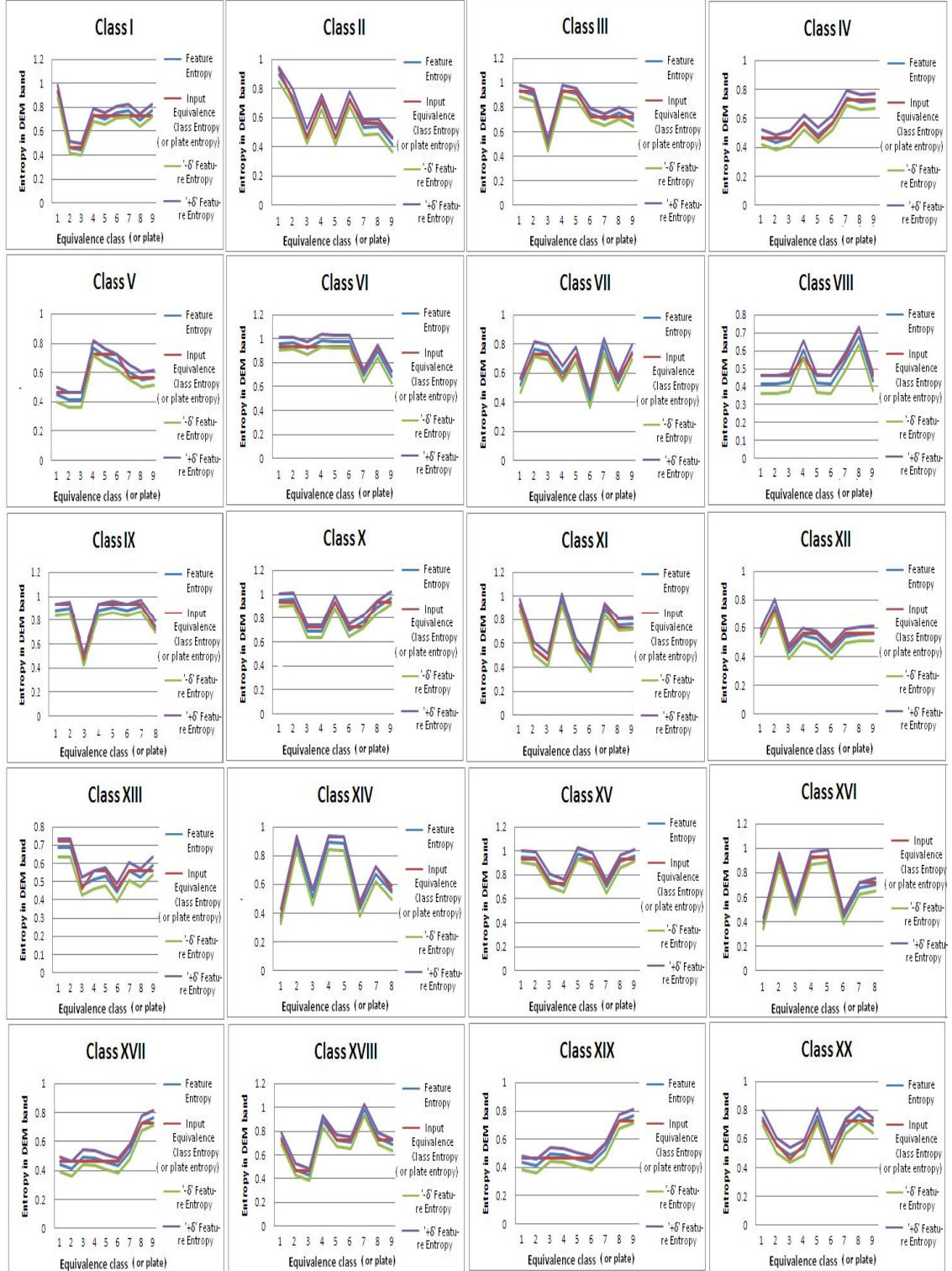


Figure 3.18: Feature Entropy v/s Plate Entropy for each plate in Classes I-XX

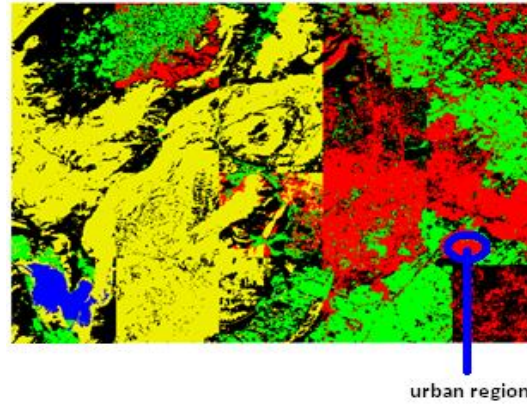


Figure 3.19: Final classified image of Alwar after applying biogeography and geo-sciences based land cover feature extraction. (Kappa Coefficient=0.7440)

classification which classified 127 vegetation pixels correctly and produced a total of 23 wrongly classified pixels (Table 2.1 of chapter 2).

To illustrate the computation of KHAT for the error matrix in Table 3.14,

$$\sum_{i=1}^r x_{ii} = 142 + 66 + 185 + 69 + 161 = 623$$

$$\sum_{i=1}^r x_{i+} \cdot x_{+i} = (150 \times 143) + (190 \times 81) + (200 \times 204) + (70 \times 69) + (170 \times 283) = 1,30,580$$

$$\begin{aligned} \hat{k}' \text{ (kappa)} &= (780 \times 623 - 130580) / (780 \times 780 - 130580) \\ &= 0.7440 \end{aligned}$$

Hence the KHAT statistics of the proposed biogeography and geo-sciences based land cover feature extractor is 0.7440.

Table 3.9: DEM entropies of each plate obtained in each iteration through texture analysis.

Iteration (k) / Entropy in DEM band for plate(i)	1	2	3	4	5
1	0.3886	0.3980	0.3633	0.3518	0.3246
2	0.9096	0.3820	0.3981	0.3618	
3	0.5119	0.3601	0.3674	0.3203	
4	0.9194	0.3687	0.3336		
5	0.9374	0.3901	0.3109		
6	0.4319	0.3697	0.3416		
7	0.6749	0.3616	0.3403		
8	0.7052	0.3516	0.3375		
9		0.3848	0.3640		
10		0.9167	0.3393		
11		0.4619	0.3530		
12		0.4668	0.3388		
13		0.9258	0.3433		
14		0.4839	0.3496		
15		0.4408	0.3679		
16		0.9557	0.3409		
17		0.4680	0.3428		
18		0.4692	0.3709		
19		0.9321	0.3382		
20		0.9565	0.3473		
21		0.4785	0.3589		
22		0.9548	0.3388		
23		0.9206	0.3405		
24		0.9513	0.3586		
25		0.9446	0.3410		
26		0.7444	0.3455		
27		0.9181	0.3336		
28		0.9390	0.3684		
29		0.9141	0.3659		
30		0.9127	0.9397		
31		0.9087	0.9304		
32		0.9157	0.9411		
33		0.9162	0.9142		
34		0.9559	0.9100		
35		0.9316	0.9106		
36		0.9387	0.9246		

Table 3.10: Categorization of plates obtained through texture analysis into similarity classes

Similarity class 'c'	Plates in 'c'
Water	1
Rocky	2,4,5
Barren	7,8
Vegetation	3,6

Table 3.11: Feature categorization step in order to give a class tag to the plates

Similarity class	Average entropy	Feature mapped
1	0.3886	Water
2	0.9221	Rocky
4	0.6999	Barren
5	0.4719	Vegetation

Table 3.12: HSI calculations and the corresponding classified feature for the first iteration

Similarity class 'c'	Feature Entropy	Plate Entropy	HIS	Classified Feature
1	0.3486	0.3886	0.0400	---
2	0.9316	0.9096	0.0220	Rocky
		0.9194	0.0122	Rocky
		0.9374	0.0058	Rocky
3	0.7249	0.6749	0.0500	---
		0.7052	0.0197	Barren
4	0.4619	0.4319	0.0300	---
		0.5119	0.0500	---

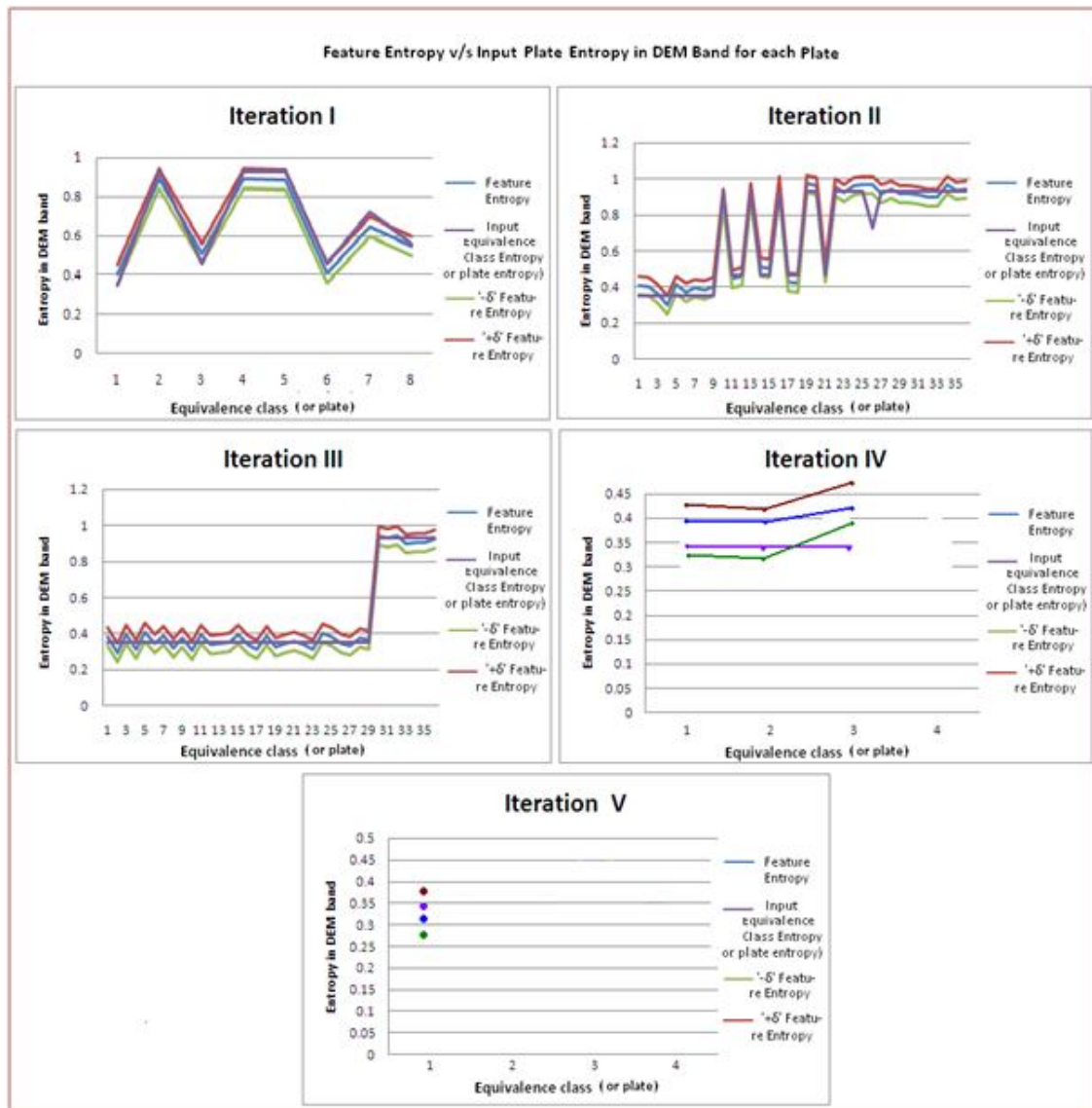


Figure 3.20: Feature Entropy v/s Input Plate Entropy in DEM band for each plate for Iterations II-V.

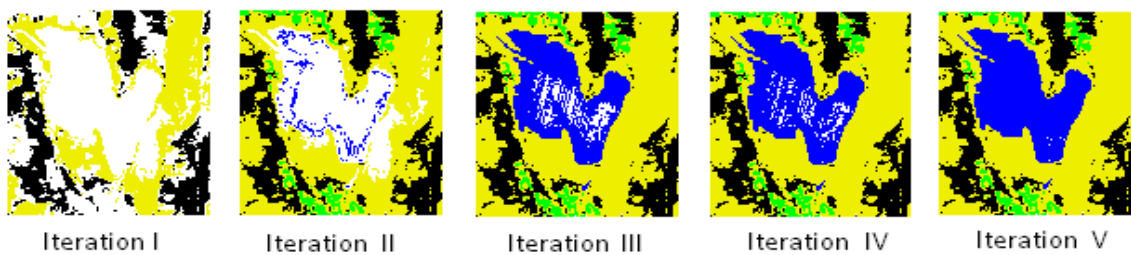


Figure 3.21: Water body extraction (in blue color) from the Alwar image with each iteration. White color portion represents the unclassified image.

Table 3.13: HSI calculations and feature classification for Iterations II-V

Iteration II						Iteration III					
Similarity class 'c'	Plates in 'c'	Feature Entropy	Plate Entropy	HSI	Classified feature	Similarity class 'c'	Plates in 'c'	Feature Entropy	Plate Entropy	HSI	Classified feature
1	1-9	0.3486	0.3980	0.0494	---	1	1-28	0.3486	0.3633	0.0147	Water
			0.3820	0.0334	---				0.3981	0.0495	---
			0.3601	0.0115	Water				0.3674	0.0188	Water
			0.3687	0.0201	Water				0.3336	0.015	Water
			0.3901	0.0415	---				0.3109	0.0377	---
			0.3697	0.0211	Water				0.3416	0.007	Water
			0.3616	0.013	Water				0.3403	0.0083	Water
			0.3516	0.003	Water				0.3375	0.0111	Water
			0.3848	0.0362	---				0.3640	0.0154	Water
2	10,13,16, 20,22,23, 25,27-36	0.9316	0.9167	0.0149	Rocky				0.3393	0.0093	Water
			0.9258	0.0058	Rocky				0.3530	0.0044	Water
			0.9557	0.0241	Rocky				0.3388	0.0098	Water
			0.9321	0.0005	Rocky				0.3433	0.0053	Water
			0.9565	0.0249	Rocky				0.3496	0.001	Water
			0.9548	0.0232	Rocky				0.3679	0.0193	Water
			0.9206	0.011	Rocky				0.3409	0.0077	Water
			0.9513	0.0197	Rocky				0.3428	0.0058	Water
			0.9446	0.013	Rocky				0.3709	0.0223	Water
			0.9181	0.0135	Rocky				0.3382	0.0104	Water
			0.9390	0.0074	Rocky				0.3473	0.0013	Water
			0.9141	0.0175	Rocky				0.3589	0.0103	Water
			0.9127	0.0189	Rocky				0.3388	0.0098	Water
			0.9087	0.0229	Rocky				0.3405	0.0081	Water
			0.9157	0.0159	Rocky				0.3586	0.01	Water
			0.9162	0.0154	Rocky				0.3410	0.0076	Water
			0.9559	0.0243	Rocky				0.3455	0.0031	Water
			0.9316	0	Rocky				0.3336	0.015	Water
			0.9387	0.0071	Rocky				0.3684	0.0198	Water
3	11,12,14, 17,18,21	0.4619	0.4619	0	Vegetation	2	29-36	0.9316	0.3659	0.0173	Water
			0.4668	0.0049	Vegetation				0.9397	0.0081	Rocky
			0.4839	0.022	Vegetation				0.9304	0.0012	Rocky
			0.4408	0.0211	Vegetation				0.9411	0.0095	Rocky
			0.4680	0.0061	Vegetation				0.9142	0.0174	Rocky
			0.4692	0.0073	Vegetation				0.9100	0.0216	Rocky
			0.4785	0.0166	Vegetation				0.9106	0.021	Rocky
4	26	0.7249	0.7444	0.0195	Barren				0.9246	0.007	Rocky

Iteration IV						Iteration V					
Similarity class 'c'	Plates 'c'	Feature Entropy	Plate Entropy	HSI	Classified Feature	Similarity class 'c'	Plates 'c'	Feature Entropy	Plate Entropy	HSI	Classified Feature
1	1,2,3	0.3486	0.3518	0.0032	Water	1	1	0.3486	0.3246	0.0240	Water
			0.3618	0.0132	Water						
			0.3203	0.0283	---						

Table 3.14: Error matrix of biogeography and geo-sciences based extractor on Alwar region. (Kappa Coefficient = 0.7440)

	Vegetation	Urban	Rocky	Water	Barren	Total
Vegetation	142	0	0	0	1	143
Urban	3	66	8	0	4	81
Rocky	0	14	185	1	4	204
Water	0	0	0	69	0	69
Barren	5	111	6	0	161	283
Total	150	190	200	70	170	780

B. Case study: Patalganga region in Shivalik ranges

This section demonstrates the applicability of the proposed classifier by applying it on the Patalganga image of Shivalik ranges. The feature entropies are calculated and the entropies in the DEM band for each plate (equivalence class) obtained through texture analysis are also calculated on the Patalganga image with the threshold ‘ δ ’ of -0.05 to +0.05 as shown in Table 3.15. The feature entropy v/s the plate entropy plotted for each plate in the DEM band for each partition of the image can also be plotted graphically (see figure 3.22). The classified image is obtained in figure 3.23 which clearly shows the extraction of land cover features of snow, vegetation and rocky area and has a kappa coefficient of 0.7870. The yellow color represents rocky area, green color represents vegetation area, and red color represents the snowy area. The original BBO classified image of Patalganga displayed a kappa coefficient of 0.7248 [Panchal, V. K., et. al., 2009]. The efficiency of the proposed classifier is reflected from the classification of the encircled region which has been correctly classified as rocky region in figure 3.23 (which was wrongly classified as snowy region by the original BBO based feature extractor as is reflected in the work [(Goel, L., et. al., 2011A), (Goel, L., et. al., 2013C)]). Table 3.16 displays the classification accuracy of the proposed algorithm by means of the classification error matrix. (Error matrix for the original biogeography based classifier is given in Table 2.14 of chapter 2 [(Goel, L., et. al., 2010B), (Goel, L., et. al., 2012A)]).

The KHAT statistic is computed as:

$$\sum_{i=1}^r x_{ii} = 197,630 + 121,008 + 16,412 = 335,050$$

$$\sum_{i=1}^r x_{i+} \cdot x_{+i} = (220,202 \cdot 215,960) + (143,580 \cdot 139,338) + (16,412 \cdot 16,412) = 6.78 \times 10^{10}$$

$$\begin{aligned} \hat{k} \text{ (kappa)} &= (375,952 \times 335,050 - 6.78 \times 10^{10}) / (375,952^2 - 6.78 \times 10^{10}) \\ &= 0.7870 \end{aligned}$$

Hence the KHAT statistics of the proposed biogeography and geo-sciences based land cover feature extractor is 0.7870.

Table 3.15: HSI calculations and the corresponding classified feature for the first iteration

Similarity Class for 'Vegetation' Feature Entropy=0.8570		Similarity Class for 'Snow' Feature Entropy=0.7361				Similarity Class for 'Rocky' Feature Entropy=0.6094			
Plate Entropy	HSI	Plate Entropy	HSI	Plate Entropy	HSI	Plate Entropy	HSI	Plate Entropy	HSI
0.8296	0.0274	0.6893	0.0468	0.7162	0.0199	0.5790	0.0304	0.5738	0.0356
0.8498	0.0072	0.7507	0.0146	0.7721	0.036	0.5959	0.0135	0.6262	0.0168
0.8730	0.016	0.7720	0.0359	0.7402	0.0041	0.6014	0.008	0.5660	0.0434
0.8578	0.0008	0.7192	0.0169	0.7847	0.0486	0.5627	0.0467	0.5680	0.0414
0.8439	0.0131	0.7244	0.0117	0.7674	0.0313	0.5996	0.0098	0.5712	0.0382
0.8261	0.0309	0.7654	0.0293	0.7831	0.047	0.5765	0.0329	0.6527	0.0433
0.8082	0.0488	0.7376	0.0015	0.7812	0.0451	0.5621	0.0473	0.5598	0.0496
		0.6854	0.0507	0.7851	0.049	0.6500	0.0406	0.6161	0.0067
		0.7073	0.0288	0.7581	0.022	0.6478	0.0384	0.5773	0.0321
		0.6960	0.0401	0.7072	0.0289	0.5984	0.011	0.6154	0.006
		0.7592	0.0231	0.7816	0.0455	0.6328	0.0234	0.6239	0.0145
		0.7691	0.033	0.7818	0.0457	0.6180	0.0086	0.5974	0.012
		0.6953	0.0408	0.7261	0.01	0.5672	0.0422	0.5851	0.0243
		0.7819	0.0458	0.7059	0.0302	0.5656	0.0438	0.6369	0.0275
		0.7597	0.0236	0.7227	0.0134	0.5856	0.0238	0.5597	0.0497
		0.7798	0.0437	0.7146	0.0215	0.5906	0.0188		
		0.7377	0.0016	0.7360	1E-04	0.5793	0.0301		
		0.7609	0.0248	0.7536	0.0175	0.6022	0.0072		
		0.6976	0.0385	0.7031	0.033	0.6062	0.0032		
		0.7807	0.0446	0.7245	0.0116	0.6101	0.0007		
		0.7859	0.0498			0.6467	0.0373		
		0.7063	0.0298			0.6000	0.0094		
		0.7628	0.0267			0.6031	0.0063		
		0.7263	0.0098			0.5971	0.0123		
		0.7774	0.0413			0.5614	0.048		
		0.7586	0.0225			0.6541	0.0447		
		0.7130	0.0231			0.5954	0.014		
		0.6863	0.0498			0.5682	0.0412		
		0.7528	0.0167			0.5849	0.0245		
		0.7069	0.0292			0.5829	0.0265		
		0.7771	0.041			0.5899	0.0195		
		0.7133	0.0228			0.5680	0.0414		
		0.7779	0.0418			0.5902	0.0192		
		0.7336	0.0025			0.5752	0.0342		
		0.7762	0.0401			0.5868	0.0226		
		0.7806	0.0445			0.6136	0.0042		
		0.7375	0.0014			0.6036	0.0058		
		0.7818	0.0457			0.5974	0.012		
		0.7507	0.0146			0.5632	0.0462		
		0.7837	0.0476			0.5855	0.0239		
		0.7256	0.0105			0.5839	0.0255		
		0.6985	0.0376			0.6405	0.0311		
		0.7838	0.0477			0.5944	0.015		
		0.7623	0.0262			0.5958	0.0136		
		0.7580	0.0219			0.5640	0.0454		
		0.7747	0.0386			0.5702	0.0392		
		0.7815	0.0454			0.5753	0.0341		
		0.7574	0.0213			0.5716	0.0378		
		0.7703	0.0342			0.5881	0.0213		
		0.7826	0.0465			0.5880	0.0214		
		0.7553	0.0192			0.6002	0.0092		
		0.7071	0.029			0.5637	0.0457		
		0.7043	0.0318			0.5614	0.048		
		0.6936	0.0425			0.6581	0.0487		

Feature Entropy v/s Input Plate Entropy for each plate in Classes I-XV

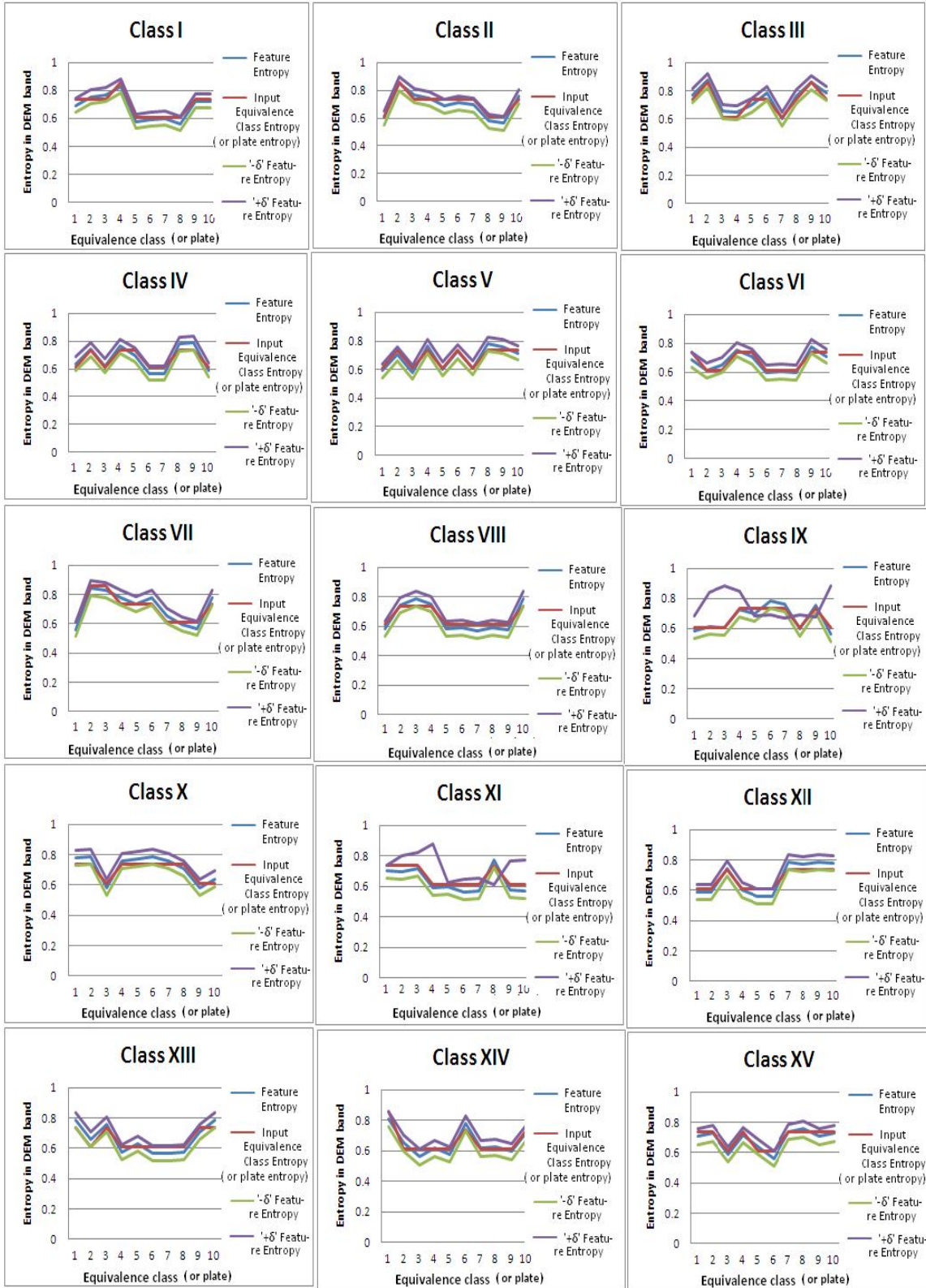


Figure 3.22: Feature Entropy v/s Plate Entropy for each plate in Classes 1-XV

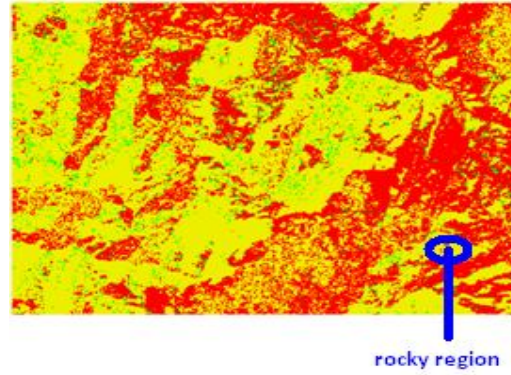


Figure 3.23: Final classified image of Patalganga region after applying biogeography and geo-sciences based land cover feature extraction. (Kappa Coefficient=0.7870)

Table 3.16: Error matrix of biogeography and geo-sciences based extractor on Patalganga region. (Kappa Coefficient = 0.7870)

	Rocky	Snow	Vegetation	Total
Rocky	197,630	22,572	0	220,202
Snow	18,330	121,008	0	139,338
Vegetation	0	0	16,412	16,412
Total	215,960	143,580	16,412	375,952

3.2.4. Comparison of BBO-GS based classifier with the other recent geo-spatial feature extraction techniques

This section demonstrates the effectiveness and applicability of the BBO-GS based classifier for improving the hybrid bio-inspired classifier [(Goel, L., 2010B), (Goel, L., et. al., 2012A)].

The choice of DEM band for evaluating the entropy in the designed BBO-GS based hybrid classifier is in accordance with the proposed algorithm which states that the texture analysis is done in the DEM band since this band well represents the surface topography however, the choice of the band for evaluating the entropy depends upon the application on which feature extraction is performed and the nature and characteristics of dataset. Thus, depending upon the features which are required to be extracted from the image, the band used for calculating the entropy values may be chosen. For optimal band selection for application of the proposed BBO-GS method, the reader is referred to the work [Goel, L., et. al., 2010C] which discusses the optimal choice of bands depending upon the features contained in the input image or which are required to be extracted from the image. The presented application takes as input satellite multi-spectral images of natural terrain for which the DEM band is a good indicator of identifying different land cover features however, for other multi-spectral datasets, other bands may be more appropriate.

The proposed biogeography and geo-sciences based feature extractor can be applied for the improvement of the hybrid classifier that was developed in the previous work [Goel, L., et. al., 2012A]. The section next presents the results of classification of the BBO-GS based hybrid

ACO2/PSO/(BBO-GS) classifier since the hybrid ACO2/PSO/BBO classifier is the best known classifier developed till date [(Franklin, J., 1995), (Johal, N. K., et. al., 2010)] and had outperformed the other recent evolutionary classifiers [(Dasgupta, D., et. al., 2011), (Beyer, H., 2001), (Back, T., 1996)]. Hence the result of substituting the original BBO with the proposed BBO-GS in the design of this hybrid bio-inspired intelligent classifier is studied. Also, the results are compared with this benchmark classifier and with only those feature extractors which have been developed thereafter for testing the effectiveness of the proposed evolutionary classifier [(Kang, F., et. al., 2011), (Kang, F., et. al., 2013A), (Kang, F., et. al., 2013B)] on the Alwar dataset.

The hybrid ACO2/PSO/BBO classifier with the highest kappa coefficient of 0.9818 [(Dorigo, M., et. al., 2004), (Goel, L., et. al., 2012A), (Clerc, M., 2006)] is shown in figure 3.24(d). Figure 3.24(a) presents the results of the extended biogeography based feature extractor (with non-linear immigration and emigration rates) on the image which has a kappa coefficient of 0.6912 [Gupta, S., et. al., 2011]. Figures 3.24(b) and (c) are the result of applying the membrane computing and the bacterial foraging algorithms (hybrid FPAB/BBO) and achieve accuracy of 0.6818 and 0.6790 respectively [(Gupta, D., et. al., 2011), (Johal, N. K., et. al., 2010)]. Figure 3.24(e) presents the results after applying the proposed hybrid ACO2/PSO/(BBO-GS) classifier which achieves the classification efficiency of 0.9870 and hence enhances the accuracy provided by the original hybrid ACO2/PSO/BBO classifier (which had a kappa coefficient of 0.9818) [(Goel, L., 2010B), (Goel, L., et. al., 2012A)]. *In fact, the hybrid ACO2/PSO/(BBO-GS) based on the proposed concept of using surface entropy in DEM band in BBO achieves the highest classification efficiency of 98.7% and hence outperforms all the above recent CI based classifiers developed and adapted till date for geo-spatial feature extraction.*

Table 3.17 below summarizes and compares the kappa coefficients of the proposed hybrid ACO2/PSO/(BBO-GS) classifier with these very recent feature extractors and the benchmark hybrid ACO2/PSO/BBO classifier that have been used for demonstrating the effectiveness of the proposed evolutionary classifier [(Kang, F., et. al., 2011), (Kang, F., et. al., 2013A), (Kang, F., et. al., 2013B)]. Now the classification accuracy of the proposed classifier is calculated using the classification *error matrix*. The error matrix for the original hybrid ACO2/PSO/BBO based classifier can be held from the previous section [Goel, L., et. al., 2012A] and that of the proposed hybrid ACO2/PSO/BBO-GS based classifier is shown in Table 3.18. From the tables, it is evident that the proposed hybrid ACO2/PSO/BBO-GS based classifier outperforms the original one. Hence the KHAT statistics of the proposed feature extractor is 0.9870. Tables 3.19 and 3.20 depict the producer's and the user's accuracies. The results in Table 3.19 show

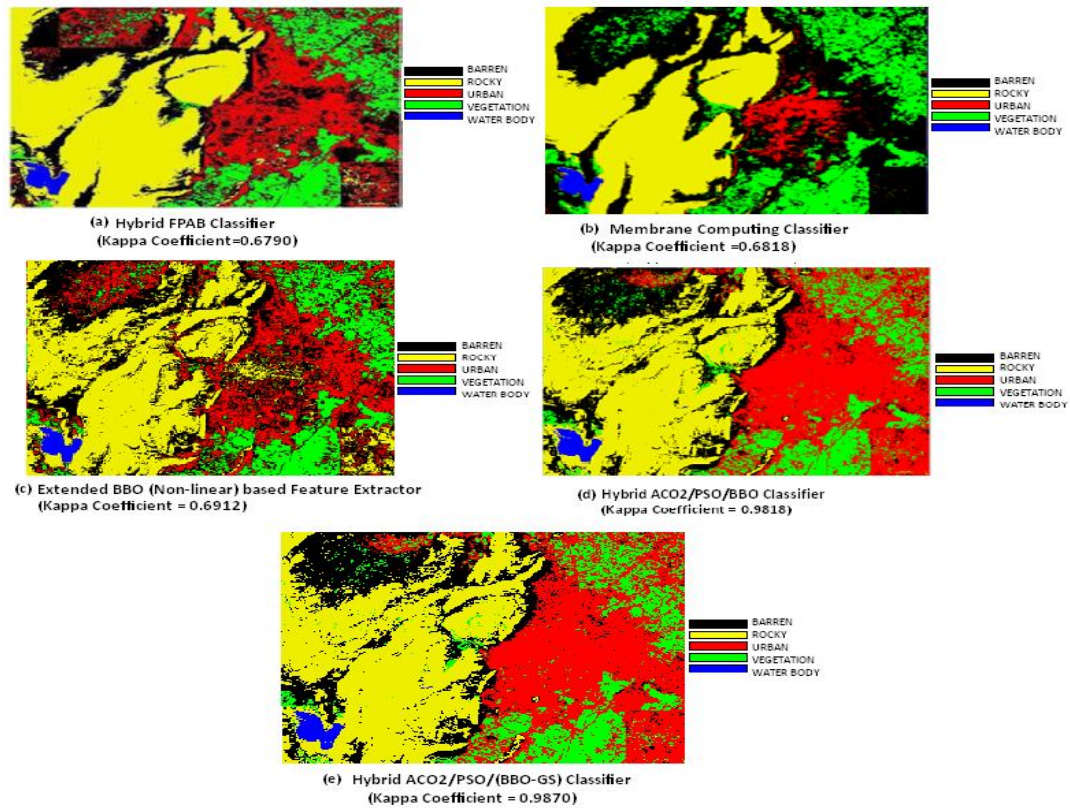


Figure 3.24: Comparison of feature extraction results of hybrid ACO2/PSO/BBO-GS based classifier with the other recent computational intelligence based classifiers on the dataset of Alwar region.

Table 3.17: Comparison of kappa coefficient of hybrid ACO2/PSO/(BBO-GS) based classifier with the recent CI based classifiers

Extended (Non-linear) biogeography based feature extractor	Membrane Computing based classifier	Hybrid FPAB Classifier	Hybrid ACO2/PSO/BO Classifier	Hybrid ACO2/PSO/(BBO-GS) based feature extractor
0.6912	0.6818	0.6790	0.98182	0.9870
(Recent feature extractors)			Original and proposed hybrid feature extractor	

Table 3.18: Error matrix of hybrid ACO2/PSO/(BBO-GS) on Alwar region. (Kappa Coefficient = 0.9870)

	Vegetation	Urban	Rocky	Water	Barren	Total
Vegetation	140	0	0	0	0	140
Urban	2	190	0	1	1	194
Rocky	0	0	198	0	2	200
Water	0	0	0	69	0	69
Barren	1	0	1	0	162	164
Total	143	190	199	70	165	767

Table 3.19: Producer's accuracy

Feature	Accuracy Calculation	Producer's Accuracy
Vegetation	140/143	97.9%
Urban	190/190	100.0%
Rocky	198/199	99.5%
Water	69/70	99.0%
Barren	162/165	98.2%

Table 3.20: User's accuracy

Feature	Accuracy Calculation	User's Accuracy
Vegetation	140/140	100.0%
Urban	190/194	97.9%
Rocky	198/200	99.0%
Water	69/69	100%
Barren	162/164	98.8%

that urban pixels have been extracted most efficiently followed by rocky and water pixels, respectively. From the Table 3.20, it can be observed that water and vegetation features show 100% accuracy of classification followed by rocky pixels.

3.3. Observations

Based on the performance governing analytical model, BBO is integrated with the nature inspired techniques of ACO and PSO to develop a hybrid model for land cover feature extraction. The hybrid classifier explored the adaptive nature of BBO and therefore is flexible enough to classify a particular land cover feature more efficiently than others based on the multi-spectral data and hence can be adapted according to the application. The proposed hybrid classifier improves the efficiency of the biogeography based classifier for a more efficient, focused and refined image classification. The hybrid classifier can be further improved by proposing certain modification to the algorithm so that the kappa coefficient can be improved further. Also, an unsupervised version of the biogeography based satellite image classification may be explored by including the context information of clusters used in generation of species granules and then evaluating the HSI. The current system is implemented using a simple heuristic technique namely 'deviation from mean metric'; the system performance can be increased by using other heuristic functions. Further, in this work, rough set theory has been used to obtain the elementary classes (or species). Since the band used in partitioning affect the final results, the rough set theory can be replaced by other swarm based clustering techniques, in this way the proposed feature extraction algorithm will become independent of the bands and hence, the algorithm will become completely 'swarm based'. The algorithm efficiency can be further improved by lowering the threshold value used in BBO algorithm thus leading to more iterations and refined results. Also, the image can be further divided into more clusters so that a more accurate comparison can be made and the decision about which of the two techniques to be applied on the particular cluster, can be further streamlined.

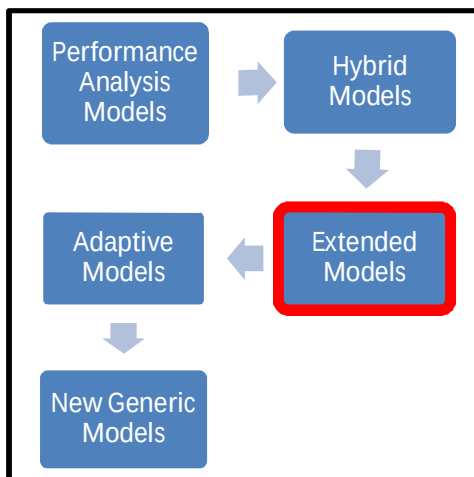
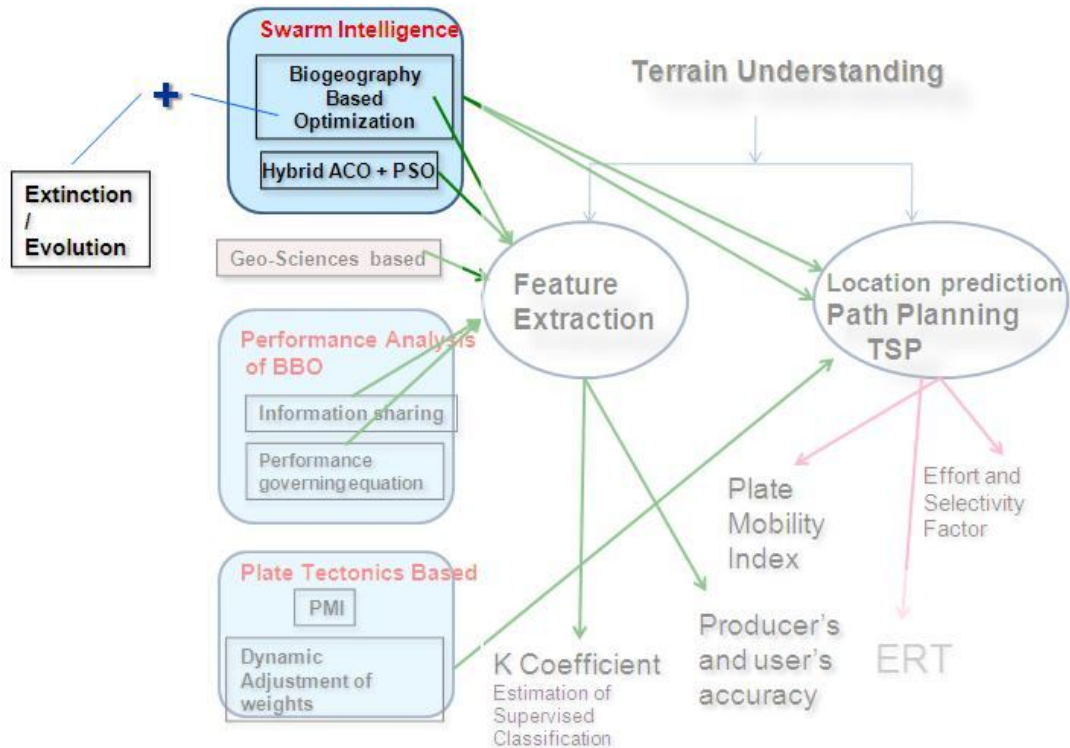
Based on the analytical model, a biogeography and geo-sciences based land cover feature extractor is designed considering the fact that it is the entropy which is the driving force leading to the formation of heterogeneous regions called as plates similar to the convection forces in the mantle of the earth and hence using entropy as the basis of the HSI function improves the classification accuracy of the original biogeography based land cover feature extractor as demonstrated in the analytical study. The main argument in the work was finding a novel mechanism of classification of images based on purely geo-sciences process. The proposed classifier has been applied to two different datasets of Alwar region in Rajasthan and Patalganga area in Shivalik ranges. The results indicate that the proposed geo-sciences based classifier is highly effective in extracting land cover features and when integrated with the hybrid bio-inspired intelligent classifier (that had demonstrated the best performance till date),

improves its classification efficiency further and hence outperforms the traditional probabilistic classifiers, the original BBO classifier as well as all the other recent computational intelligence based classifiers and hence proves to be the best known classifier developed till date. The proposed model is however, flexible and can be made applicable to a large number of classification problems wherein entropy can be calculated in any other band in case of applications where the dataset is terrain multi-spectral images for example, mixed pixel resolution [(Sinha, S., et. al., 2012), (Varshney, T., et. al., 2014)], design of hybrid classifiers, etc. For example, for the application of mixed pixel resolution in satellite multi-spectral imagery of natural terrain, the algorithm requires to calculate entropy in DEM band, however, for mixed pixel resolution in magnetic resonance imaging for disease diagnosis, the entropy is not calculated in DEM band. In case of applications where the dataset are normal images, for example, in case of face recognition application [Gupta, D., et. al., 2013A], pattern recognition, etc., the entropy function can be simply formulated according to its standard definition and evaluate the fitness function. In fact, the proposed BBO-GS based algorithm can be applied for various feature extraction applications by modifying the fitness function to use surface entropy (or surface entropy in the appropriate band in case of multi-spectral images) of the digital number values of the image pixels as the measure of comparison and classification of features in the input image dataset [Plaza, A., et. al., 2008]. The research can be further improved by dividing the image into more plates so that a more accurate comparison can be made and the classification efficiency can be further increased by more accurate categorization of mixed pixels [(Gupta, N., et. al., 2011B), (Varshney, T., et. al., 2014)].

Based on the behavioral analysis of BBO that was carried out in the previous chapter, this chapter was an attempt to work on to improve the analyst's decisive capabilities [(Aamodt, A., et. al., 1994), (Bargiela, A., et. al., 2003)] for the selection of the most appropriate technique for the terrain understanding problem by proposing hybrid optimization models [(Goel, L., et. al., 2010A), (Goel, L., 2010B), (Goel, L., et. al., 2012A), (Goel, L., et. al., 2013A)], which can refine the results obtained individually.

The next chapter deals with developing extended mathematical models inspired from natural phenomenon of biogeography in computational intelligence and hence extending the basic optimization functionalities of existing nature inspired technique of BBO. The chapter also analyzes their performance and compares them with the original model.

Extended Optimization Models of Nature Inspired Computational Intelligence



Chapter 4

- 4.1. Biogeography with Extinction and Evolution
- 4.2. Biogeography based Optimization with Extinction and Evolution
- 4.3. Extended Models of Species Abundance in Biogeography
- 4.4. Performance Analysis
- 4.5. Observations

This chapter deals with developing mathematical models inspired from natural phenomenon in computational intelligence and extending the basic functionalities of optimization of existing nature inspired optimization techniques, analyzing their performance and comparing them with the original model.

Motivated by the migration mechanisms of species, various extensions to BBO were proposed including the blended BBO algorithm [Ma, H., et. al., 2011], oppositional BBO [Eregezer, M., et. al., 2009], etc. BBO has become a popular EA by improving with additional amendments and extensions, for example, it is applied to practical engineering problems such as ECG diagnosis [Ovriiu, M., et. al., 2010], optimal power flow problem [Rarick, R., et. al., 2009], economic load dispatch [Bhattacharya, A., et. al., 2010], ground water detection [Kundra, H., et. al., 2010], satellite image classification [(Goel, L., 2010B), (Goel, L., et. al., 2012A), (Panchal, V. K., et. al., 2009)] and other remote sensing applications, etc.

Biogeography is modeled in terms of such factors as habitat area and immigration and emigration rates, and describes the evolution, extinction and migration of species. However, the extensions of BBO which have been proposed to date do not consider the factors of evolution and extinction of species in determining the number of species in a habitat at a time instant and hence, two additional factors, population evolution rate and population extinction rate, are introduced besides immigration and emigration for the characterization of a habitat with a high or a low HSI value [Goel, L., et. al., 2012D]. The mix of species and their evolution pattern have a significant effect on the physical characteristics of habitat and their contribution to HSI. For example, if a species for which a particular SIV was important is extinct now due to natural evolution or human activity or epidemic, then that particular SIV is no longer relevant to HSI calculation. The result of applying the extended form of BBO on an optimization problem is that the HSI which is a function of SIVs is now not only determined by migration but also by extinction and evolution. The extinction rate may be used to decide the less relevant SIVs which are likely to become extinct in near future and hence these may not be considered further for participation in migration for the improvement of the solutions. The evolution rate may be used to decide the significant SIVs which are active participants in improvement of the solution and hence these are more likely to be chosen for migration [Simon, D., 2008]. Thus, with the incorporations of factors of extinction and evolution, the SIVs which are less relevant to HSI calculations have been preliminarily considered extinct and more probable SIVs are considered evolved and are the ones that participate in migration, hence, the HSI of the solutions are determined by only the remaining relevant SIVs. Hence, the proposed model tries to streamline the process of migration by improving the quality of the solutions preliminarily thus leading to better results than the original model. An additional dependency factor is also introduced which signifies the interdependence of migrating species on each other such as the predator-prey

relationships; to be considered for the determination of immigration and emigration rates and hence further extend the original model of species abundance in a single habitat which further simplifies the HSI definition since dependent SIVs migrate together as a group. Hence the extended form of the biogeography based optimization technique, the BBO/EE algorithm is proposed. The performance of BBO/EE is demonstrated by running it on the same set of 14 standard benchmark functions that were originally used by Simon [Goel, L., et. al., 2012D]. The extended forms of each of the six representative migration models that were originally proposed by Haiping Ma as an extension to the work by Simon are also presented. The performance of each of the extended models of BBO is demonstrated by running them on same 14 standard benchmark functions and compared with the original species abundance model of BBO [Goel, L., et. al., 2012D].

The rest of the chapter is organized as follows. Section 4.1 describes the theoretical foundations of the extended form of the biogeography based optimization with extinction / evolution and proves new theorems about species count probabilities; second it extends the original species abundance model and the relationship between the species count probabilities and the species counts. Section 4.2 presents the proposed BBO/EE algorithm along with the set of relevant definitions. Section 4.3 extends the six representative migration models by Ma and presents the extended species abundance models. Section 4.4 presents the optimization results of the BBO/EE and its extended models by running on benchmark functions and comparing with original BBO and other EAs. Section 4.5 concludes the work and discusses the observations made.

4.1. Biogeography with Extinction and Evolution

This section presents the proposed species count model which forms the foundation of the development of the BBO/EE technique based on the new definition of the HSI function in the next section. The section has three subsections where the first section introduces the additional factors, besides migration, second section presents the concepts of biogeography with species population modeling based on the modified expression of species count on a habitat and the last section presents the proposed extended species abundance model based on the concepts introduced in the previous sections.

4.1.1. Factors of evolution and extinction

The extensions of BBO that have been proposed till date do not consider the factors of evolution and extinction of species in determining the number of species in a habitat at a time instant and only consider the immigration and the emigration rates as the deciding factor for the determination of the number of species at a time instant. Hence, the above two additional factors, are introduced besides immigration and emigration in the calculation of the species count, for the characterization of a habitat with a high or a low HSI value. Hence, the

population growth is determined by four factors, births (B), deaths (D), immigrants (I), and emigrants (E) using a formula expressed in equation (4.1) [Turchin, P., 2003].

$$\Delta P \equiv B - D + I - E \quad (4.1)$$

In other words, the population growth of a period on a habitat can be calculated in two parts, natural growth of population (B - D) and mechanical growth of population (I - E) [Turchin, P., 2003]. In the original as well as the extensions to BBO, the species count on a habitat is determined by solely the mechanical growth and hence the importance of factors categorized under natural growth namely, evolution and extinction rates, in the determination of species count, is discussed next in this section.

Population evolution rate is a governing factor in determining the number of species in a habitat at a time instant since the number of species in a habitat may increase due to several reasons besides immigration such as cross-breeding utilizing the biotic potential of the species, bringing in new species by natural causes such as wind, etc. or by human activities, plentiful resources, optimal environmental conditions, etc. The rate of evolution is characterized by the pattern shown in graph below in figure 4.1. The population evolution graph in figure 4.1 resembles a rectangular hyperbola since the product of evolution rate and the number of species always remains constant. Hence, the evolution curve can be represented by the mathematical equation (4.2) to represent the above relationship.

$$\alpha \cdot k = c \quad (4.2)$$

where α is the rate of evolution of species, k is the number of species at some stated interval of time and c is a constant which characterizes the evolution curve.

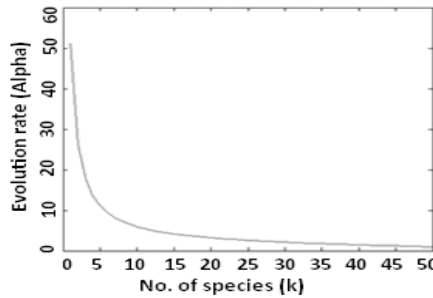


Figure 4.1: Graph representing the generalized behaviour of species evolution rate plotted against the species count on a single habitat.

Population extinction rate is a governing factor in determining the number of species in a habitat at a time instant since the number of species in a habitat may decrease due to several reasons besides emigration such as the vagaries of the physical environment, for example drought, freezes, hurricane, floods, forest fires, etc., intra-specific competition, inter-specific competition, reproductive competition, predation, parasitism, population cycles, the limited carrying capacity of the environment, etc. Based on the different models of species survivorship which characterize species with high infant mortality and birth rate, species with

low mortality rate, species with high infant mortality rate, inference can be drawn about the generalized species survivorship model with a theoretical population with half life of 18 years assuming that the generalized behavior of species is random mortality throughout life. Now, the species extinction curve is obtained as the inverse of the generalized species survivorship curve. Hence, the rate of extinction is characterized by the pattern shown in graph below in figure 4.2. It can be seen that the population survivorship curve which will be the reverse of the extinction rate curve is a parabola with the origin shifted to time $t = t_{max}$ and hence it can be represented by the equation below.

$$(x - t)^2 = 4by \quad (4.3)$$

$$\text{Replacing the values, } (t - t_{max})^2 = 4b \left(1 - \frac{\gamma}{n}\right) \Rightarrow \left(\frac{t - t_{max}}{t_{max}}\right)^2 = 1 - \frac{\gamma}{n} \quad (4.4)$$

(The above equation holds since, at, $t = 0$, $4b = (t_{max})^2$)

Differentiating with respect to t ,

$$2 \left(\frac{t - t_{max}}{t_{max}}\right) = -\frac{1}{n} \cdot \frac{d\gamma}{dt} \quad (4.5)$$

Now, $\beta = \frac{d\gamma}{dt}$ and hence, substituting in the above equation,

$$\beta = 2n \left(1 - \frac{t}{t_{max}}\right) \quad (4.6)$$

$$\text{Now, } \left(1 - \frac{t}{t_{max}}\right)^2 = 1 - \frac{\gamma}{n} = \frac{k}{n} \quad (4.7)$$

Hence, the extinction rate β is given by equation below.

$$\beta = 2\sqrt{nk} \quad (4.8)$$

where, β is the extinction rate of species, γ is the no. of species which have become extinct at time t , k is the number of species at any stated interval of time and n is the maximum species count at the initial stage of the habitat.

Hence, the significance of the additional factors governing the population of species at a time instant is established. In the next section, the equation proposed by Dan Simon [Simon, D., 2008] governing the population distribution of a habitat in biogeography is refined.

4.1.2. Biogeography with Extinction / Evolution

This section presents the concepts of extended biogeography to include additional factors in the species count definition and will be used for the development of BBO/EE model. The mathematical model of biogeography describes how species move from one habitat to another (migration), how new species emerge (evolution), and how species disappear (extinction). Each of the above aspects in the species count model is modeled in the form of differential equations for species count probabilities [(Ma, H., et. al., 2009), (Ma, H., et. al., 2010), (Simon, D., 2008), (Simon, D., 2011)].

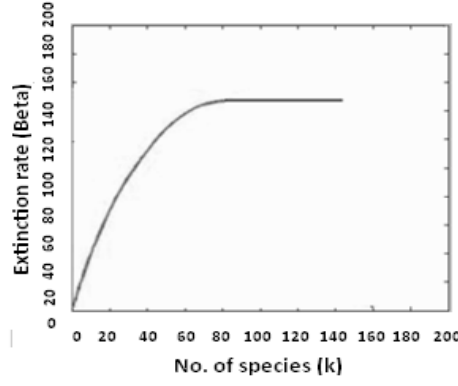


Figure 4.2: Graph representing the generalized species extinction rate curve plotted against the number of species. Consider a model of species counts in a single habitat, whose state at any time is represented by the number of species in the habitat at that time. Suppose that the largest possible number of species

$$\begin{aligned} \dot{P}_k &= \begin{cases} -(\lambda_0 + \alpha_0) P_0 + P_1(\mu_1 + \beta_1) & k = 0 \\ -P_k(\lambda_k + \alpha_k + \mu_k + \beta_k) + P_{k-1}(\lambda_{k-1} + \alpha_{k-1}) + P_{k+1}(\mu_{k+1} + \beta_{k+1}) & 1 \leq k \leq n-1 \\ -P_n(\mu_n + \alpha_n + \beta_n) + P_{n-1}(\lambda_{n-1} + \alpha_{n-1}) & k = n \end{cases} \end{aligned} \quad (4.9)$$

that the habitat can support is n . Whenever there are k species in the habitat, new arrivals enter the habitat at an immigration rate λ_k with the species evolution rate α_k , and the species leave the habitat at an emigration rate μ_k with the species extinction rate β_k . The probability P_k that the habitat contains exactly k species where P_k changes from time t to time $(t + \Delta t)$ is derived as follows in equation (4.10) [Ma, H., et. al., 2009]:

$$\begin{aligned} P_k(t + \Delta t) &= P_k(t)(1 - \lambda_k \Delta t - \mu_k \Delta t - \alpha_k \Delta t - \beta_k \Delta t) + P_{k-1}(t) \cdot \Delta t (\lambda_{k-1} + \alpha_{k-1}) + \\ &P_{k+1}(t) \cdot \Delta t (\mu_{k+1} + \beta_{k+1}) \end{aligned} \quad (4.10)$$

This equation (4.10) is true because in order to have k species at time $(t + \Delta t)$, one of the following conditions must hold: (1) There were k species at time t and no immigration, emigration, evolution or extinction of species occurred between time t and time $(t + \Delta t)$; or, (2) There were $(k-1)$ species at time t , and one specie evolved or immigrated; or, (3) There were $(k+1)$ species at time t , and one specie emigrated or became extinct. It is assumed that Δt is small enough so that the probability of more than one immigration, evolution, emigration or extinction can be ignored. Taking the limit of $\Delta t \rightarrow 0$ gives equation (4.10). It is to be noted that equation (4.10) is valid for $k = 0, \dots, n$ and $\mu_0 = 0, \beta_0 = 0, \lambda_n = 0$ and $\alpha_n = 0$. Define $P = [P_0 \dots P_n]^T$. Equation (4.11) is obtained as given below.

$$\dot{P} = AP \quad (4.11)$$

The equation for the steady state species count probability P_k is given in equation (4.13). If the species count probabilities are in steady state, then from $\dot{P} = AP$, it can be said that $AP = 0$, namely,

$$\begin{bmatrix} -(\lambda_0 + \alpha_0) & \mu_1 + \beta_1 & 0 & \dots & 0 \\ \lambda_0 + \alpha_0 & -(\lambda_1 + \alpha_1 + \mu_1 + \beta_1) & \mu_2 + \beta_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \lambda_{n-2} + \alpha_{n-2} & -(\lambda_{n-1} + \dots + \mu_{n-1}) & \mu_n + \beta_n \\ 0 & \dots & 0 & \lambda_{n-1} + \alpha_{n-1} & -(\mu_n + \beta_n) \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ \vdots \\ P_{n-1} \\ P_n \end{bmatrix} = 0 \quad (4.12)$$

From (4.12), by adding each equation to the equation preceding it,

$$\begin{aligned} (\mu_1 + \beta_1)P_1 &= (\lambda_0 + \alpha_0)P_0 \\ (\lambda_0 + \alpha_0)P_0 + (\mu_2 + \beta_2)P_2 &= (\lambda_1 + \alpha_1 + \beta_1 + \mu_1)P_1 \\ &\vdots \\ (\lambda_{n-2} + \alpha_{n-2})P_{n-2} + (\mu_n + \beta_n)P_n &= (\lambda_{n-1} + \alpha_{n-1} + \beta_{n-1} + \mu_{n-1})P_{n-1} \\ (\lambda_{n-1} + \alpha_{n-1})P_{n-1} &= (\mu_n + \beta_n)P_n \end{aligned}$$

Solving in terms of P_0 yields,

$$\begin{aligned} P_1 &= \left(\frac{\lambda_0 + \alpha_0}{\mu_1 + \beta_1} \right) P_0 \\ P_2 &= \frac{(\lambda_1 + \alpha_1)(\lambda_0 + \alpha_0)}{(\mu_2 + \beta_2)(\mu_1 + \beta_1)} P_0 \\ &\vdots \\ P_n &= \frac{(\lambda_{n-1} + \alpha_{n-1})(\lambda_{n-2} + \alpha_{n-2}) \dots (\lambda_0 + \alpha_0)}{(\mu_n + \beta_n)(\mu_{n-1} + \beta_{n-1}) \dots (\mu_1 + \beta_1)} P_0 \end{aligned}$$

By using the fact that $\sum_{k=0}^n P_k = 1$,

$$P_k = \begin{cases} P_0 = \frac{1}{1 + \sum_{k=1}^n \frac{(\lambda_{k-1} + \alpha_{k-1})(\lambda_{k-2} + \alpha_{k-2}) \dots (\lambda_0 + \alpha_0)}{(\mu_k + \beta_k)(\mu_{k-1} + \beta_{k-1}) \dots (\mu_1 + \beta_1)}} & k = 0 \\ P_n = \frac{(\lambda_{n-1} + \alpha_{n-1})(\lambda_{n-2} + \alpha_{n-2}) \dots (\lambda_0 + \alpha_0)}{(\mu_n + \beta_n)(\mu_{n-1} + \beta_{n-1}) \dots (\mu_1 + \beta_1)} P_0 & 1 \leq k \leq n \end{cases} \quad (4.13)$$

4.1.3. Species abundance model

In the Simon's model of species abundance, the immigration rate λ_k and the emigration rate μ_k are considered as linear functions of the number of species k and hence the Simon's species abundance model for a single habitat is linear for the original BBO model however, the inclusion of additional factors of evolution and extinction rates in the determination of species count make the immigration and the emigration curves complicated, contrary to that described

in the original BBO [Simon, D., 2008]. A dependency factor (φ) is also introduced with the emigration rate [Goel, L., et. al., 2012D]. The factor is an important factor in the determination of the emigration rate since species tend to resist emigration because of the dependency on other species until this dependence questions the survivability of the species themselves because of a deteriorating habitat suitability index. The HSI value of a habitat decreases with the deterioration of favorable factors (SIVs) such as rainfall, temperature, diversity of vegetation, topography, etc. Thus, the dependency factor (φ) signifies the interdependence of migrating species on each other one such example being the predator-prey relationships. The dependency factor is modeled as the ratio of the number of groups of dependent species (d) and the total number of species n for a non-zero value of d and 1 for $d = 0$ in equation (4.14) [Goel, L., et. al., 2012D]. Thus,

$$\varphi = \begin{cases} \frac{\text{Number of groups of dependent species } (d)}{n}, & d \neq 0 \\ 1, & d = 0 \end{cases} \quad (4.14)$$

Hence, an extended model to the original Simon's model is proposed where two factors are defined namely, the growth rate and the decline rates respectively where k is the number of species at the time instant t and c is a constant. The species abundance is modeled in terms of the above two factors rather than the immigration and the emigration rates where the species growth rate (σ_k) is defined as the function of immigration (λ_k) and evolution rates (α_k) and the decline rate (ρ_k) is a function of emigration (μ_k) and extinction rates (β_k) as represented by equations (4.15) and (4.16) respectively.

Figure 4.3 illustrates the extended model of species abundance in a single habitat. In figure 4.3, k_0 is the equilibrium number of species, which denotes that the immigration and the emigration rates are equal; I is the maximum possible immigration rate and E is the maximum possible emigration rate. For the graph plots, the following parameters are used: habitat (population) size $n = 50$, maximum migration rates $E = 1$ and $I = 1.2$, maximum evolution rate as $\alpha_{max} = 1$, maximum extinction rate as $\beta_{max} = 1$, $c = 50$. Substituting the value of λ_k from Simon's model [Simon, D., 2008] and α_k from (4.2), the species growth rate is defined by equation (4.15).

$$\text{Species growth rate} = \sigma_k = (\lambda_k + \alpha_k) = I \left(1 - \frac{k}{n}\right) + \frac{c}{k} \quad (4.15)$$

To account for the interdependent species' movement (emigration), the dependency factor φ is added to the mathematical formulation of the decline rate σ_k . Now, considering the dependency factor and hence substituting the value of μ_k from Simon's model [Simon, D., 2008] and β_k from (4.8), species decline rate is defined by equation (4.16). Thus,

$$\text{Species Decline Rate} = \text{Emigration Rate (with the effect of dependency factor)} + \text{Extinction Rate} = \rho_k = (\mu_k + \beta_k) = \frac{Ek\varphi}{n} + 2\sqrt{nk} \quad (4.16)$$

Figures 4.3 (a) and (b) present the species abundance model in a single habitat. Figure 4.3 (a) demonstrate the Simon's model of species abundance which is a linear function of the immigration (λ_k) and emigration rates (μ_k). Maximum values of immigration and emigration are E and I . The equilibrium point is k_0 at which the species immigration rate and the emigration rates are equal. Figure 4.3 (b) presents the proposed model of species abundance where the species abundance is a function of species growth and species decline rate. Species decline rate is a function of the function of emigration rate μ_k and the extinction rate β_k and is equal to $(\mu_k + \beta_k)$. Species growth rate is the function of immigration rate λ_k and the evolution rate α_k and is equal to $(\lambda_k + \alpha_k)$. The equilibrium point is k_0 at which the species growth rate and the decline rates are equal. The maximum value of the growth rate curve and decline rate curves are achieved at $I + \alpha_{max}$ and $E + \beta_{max}$ respectively.

Hence, it can be seen how the inclusion of the two additional factors which characterize the population growth rates and the decline rates [(Turchin, P., 2003), (Yao, X., et. al., 1999)] have affected the species abundance model. For the proposed model, the probability P_k is an even function with respect to its midpoint, as shown in figure 4.4(b) which is what is also reflected through the probability graph plot of the Simon's model as shown in figure 4.4(a). By looking at the equilibrium point on the probability curve of figure 4.4(b), it can be seen that there is high probability near the equilibrium point k_0 .

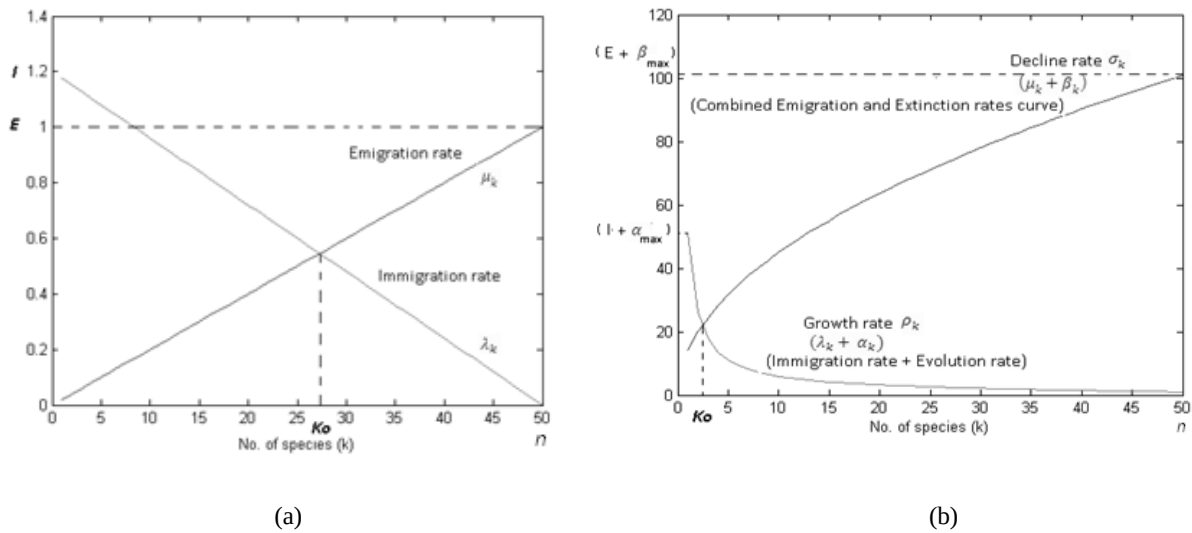


Figure 4.3: Species abundance model in a single habitat (a) Simon's model. (b) Proposed model.

Thus, with the incorporation of other factors, the equilibrium point k_0 is reached as shown in figure 4.4(b) ($k_0 \approx 27$). It is observed that the width of the probability graph is lesser than in the original BBO model which means that the difference between the low HSI and high HSI solutions is smaller here. This is because with the incorporations of factors of extinction and evolution, the SIVs which are less relevant to HSI calculations have been preliminarily considered extinct and more probable SIVs are considered evolved and are the ones that participate in migration, hence, the HSI of the solutions are determined by only the remaining relevant SIVs which reduce the difference in the two solutions. Hence, the proposed model tries to streamline the process of migration by improving the quality of the solutions preliminarily thus leading to better results than the original model. Also, the inclusion of the concept of dependency in migration of species in the determination of true migration rates further simplifies the HSI definition since the related (or dependent) SIVs migrate together as a group. Hence, convergence will be faster in the proposed model. It is this property of the proposed extended model which will be exploited to develop the extended BBO with far better optimization capabilities than the original BBO as well as the other evolutionary optimization algorithms. The point k_0 and its corresponding probability $P(k_0)$ can be found by equating the growth rates and the decline rates (for simplicity, we assume $\varphi = 1$):

$$\begin{aligned}
(\lambda_k + \alpha_k) &= (\mu_k + \beta_k) \\
\Rightarrow I \left(1 - \frac{k}{n}\right) + \frac{c}{k} &= \frac{kE}{n} + 2\sqrt{nk} \\
\Rightarrow \left(I \left(1 - \frac{k}{n}\right) + \frac{c}{k} - \frac{kE}{n}\right)^2 &= 4nk \\
\Rightarrow k^2 \left(\frac{I^2 + E^2 + 2IE}{n^2}\right) + k \left(2Ic - \frac{2I^2}{n} - \frac{2IE}{n} - 4n\right) + \left(I^2 - \frac{2Ic}{n} - \frac{2cE}{n}\right) + \frac{c}{k^2} &= 0
\end{aligned}$$

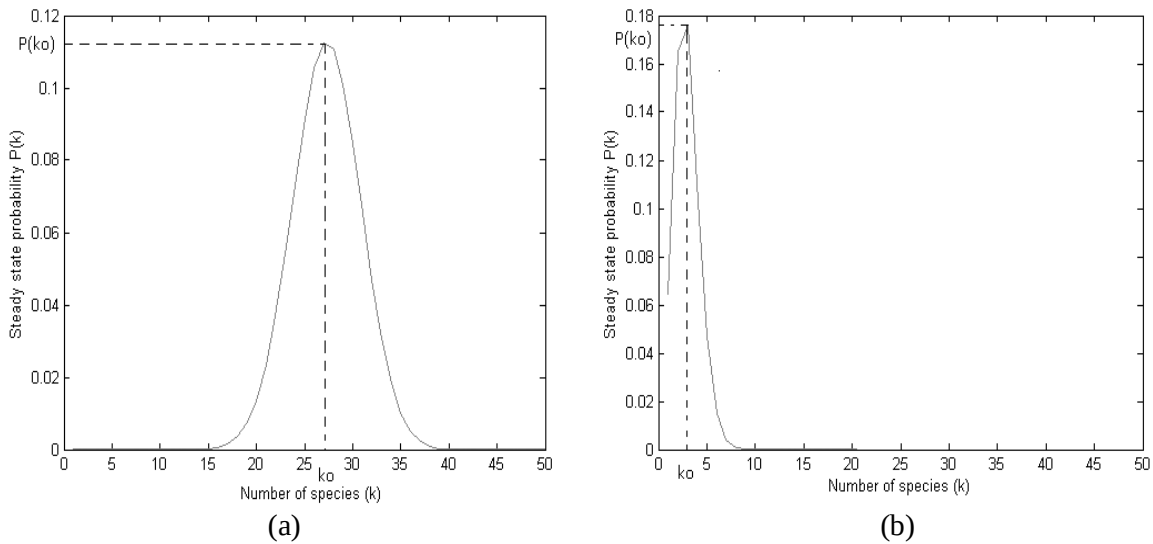


Figure 4.4: Steady state probability P_k as a function of species count k , where k_0 is the equilibrium number of species, and P_{k_0} is its corresponding probability (a) Simon's model. (b) Proposed model.

$$\Rightarrow k^4 \left(\frac{E+I}{n} \right)^2 + k^3 \left(2Ic - \frac{2I^2}{n} - \frac{2IE}{n} - 4n \right) + k^2 \left(I^2 - \frac{2Ic}{n} - \frac{2cE}{n} \right) + c = 0$$

Differentiating with respect to k and taking k common,

$$\Rightarrow 4k^2 \left(\frac{E+I}{n} \right)^2 + 3k \left(2Ic - \frac{2I^2}{n} - \frac{2IE}{n} - 4n \right) + 2 \left(I^2 - \frac{2Ic}{n} - \frac{2cE}{n} \right) = 0$$

Multiplying by k^2 on both sides,

So, $k_1 = 0$

and

$$k_2, k_3 = \frac{-3 \left(2Ic - \frac{2I^2}{n} - \frac{2IE}{n} - 4n \right) \pm \sqrt{9 \left(2Ic - \frac{2I^2}{n} - \frac{2IE}{n} - 4n \right)^2 - 32 \left(\frac{E+I}{n} \right)^2 \left(I^2 - \frac{2Ic}{n} - \frac{2cE}{n} \right)^2}}{8 \left(\frac{I+E}{n} \right)^2} \quad (4.17)$$

Now, substituting the values of the constants $E = 1$, $I = 1$, $n = 50$, $c = 50$, the values of k_2 and k_3 can be calculated ($k_2 = 31032.67$ and $k_3 = -0.025$). Now, the value of k_0 will lie between the interval $k_1 \leq k_0 \leq k_2$ or between the interval $k_2 \leq k_0 \leq k_3$. From the value of k_0 (which can be deduced from the graph in figure 4.4 (b) as approximately ≈ 3.716), it can be verified that k_0 lies between k_2 and k_3 . Also, for k_0 to lie in the interval between k_2 and k_3 , the product of $f(k_2)$ and $f(k_3)$ should come out to be negative.

i.e. $f(k_2) \cdot f(k_3) < 0$

The values of $f(k_2)$ and $f(k_3)$ are $f(k_2) = -\infty$, $f(k_3) = 2500$ the product of which comes out to be negative, hence it is verified k_0 lies between k_2 and k_3 which is what is reflected from the graph in figure 4.4(b). Having found k_0 , the corresponding probability P_{k_0} can be found using the equation (4.18).

$$\begin{aligned} P_{k_0} &= \frac{(\lambda_{k_0-1} + \alpha_{k_0-1})(\lambda_{k_0-2} + \alpha_{k_0-2}) \dots (\lambda_0 + \alpha_0)}{(\mu_{k_0} + \beta_{k_0})(\mu_{k_0-1} + \beta_{k_0-1}) \dots (\mu_1 + \beta_1)} \cdot \frac{1}{1 + \sum_{k=1}^n \frac{(\lambda_{k_0-1} + \alpha_{k_0-1})(\lambda_{k_0-2} + \alpha_{k_0-2}) \dots (\lambda_0 + \alpha_0)}{(\mu_{k_0} + \beta_{k_0})(\mu_{k_0-1} + \beta_{k_0-1}) \dots (\mu_1 + \beta_1)}} \\ &= \frac{\left(I \left(1 - \frac{k_0-1}{n} \right) + \frac{c}{k_0-1} \right) \cdot \left(I \left(1 - \frac{k_0-2}{n} \right) + \frac{c}{k_0-2} \right) \dots (I + \alpha_{max})}{\left(k_o \cdot \frac{E}{n} + 2\sqrt{nk_o} \right) \cdot \left((k_o-1) \cdot \frac{E}{n} + 2\sqrt{n(k_o-1)} \right) \dots \left(\frac{E}{n} + 2\sqrt{n} \right)} \cdot \\ &\quad \left(\frac{1}{1 + \sum_{k=1}^n \frac{\left(I \left(1 - \frac{k_0-1}{n} \right) + \frac{c}{k_0-1} \right) \cdot \left(I \left(1 - \frac{k_0-2}{n} \right) + \frac{c}{k_0-2} \right) \dots (I + \alpha_{max})}{\left(k_o \cdot \frac{E}{n} + 2\sqrt{nk_o} \right) \cdot \left((k_o-1) \cdot \frac{E}{n} + 2\sqrt{n(k_o-1)} \right) \dots \left(\frac{E}{n} + 2\sqrt{n} \right)}} \right) \end{aligned} \quad (4.18)$$

4.2. Biogeography Based Optimization with Extinction / Evolution (BBO/EE)

The factors of evolution and extinction are modeled as the phenomenon which modifies the definition of HSI and hence the extended form of the biogeography based optimization

technique with extinction / evolution based on the above concepts is proposed as detailed next. This description of the steps of BBO/EE algorithm is an extension to the original algorithm by Simon [Simon, D., 2008].

4.2.1. BBO/EE Definitions

The section begins by presenting the modified definitions of the original BBO algorithms [Simon, D., 2008] including the above additional factors where Z represents the set of integers, R is the set of real numbers and $\emptyset$ is the empty set. The value t is the maximum value of m that can be allowed for a habitat in the ecosystem.

Definition 1: A habitat $H_i \in SIV^m$ is a vector of m integers that represents a feasible solution to some problem where $m = \{1, \dots, t\}$ depending upon the habitat H_i and represents the number of problem dimensions in the considered candidate solution.

Definition 2: A suitability index variable $SIV \in C$ (where $C \subset Z^q$) is an integer that is permitted in a habitat. $C \subset Z^q$ is the set of all integers that are allowed in a habitat. The maximum number of SIVs (or decision variables) that can be allowed in the ecosystem for any habitat is t . (This number is chosen based upon the tradeoff between the computational complexity of the optimization module and the optimization efficiency guaranteed). Let the set of SIVs range from $x_1, x_2, \dots, x_t$.

Definition 3: A habitat suitability index $HSI: H \rightarrow R$ is a measure of the goodness of the solution that is represented by the habitat. HSI is defined as the function of SIVs i.e.

$$HSI_i = f(h_i) \quad (4.19)$$

where i signifies the i^{th} habitat, $i = 1$ to n . Each solution vector is considered to be characterized by a different set of SIVs h_i where the no. of SIVs in the set h_i can range from 1 to d (equal to ϕm) and is a different set for different habitats. The SIVs which are dependent upon each other (if any) are also identified and grouped together as these SIVs will migrate, evolve or become extinct together as a group. Since the dependency factor is symbolized by ϕ , which is the ratio of the number of groups of dependent SIVs (d) and the total number of SIVs (m), the HSI of a given habitat i can be calculated as given by equation (4.20).

$$HSI_i = f(h_i) = \sum_{j=1}^{h_i} f(a_j x_j) \quad (4.20)$$

where a is the coefficient characterizing the weights of different SIVs contributing towards HSI function of a given habitat i and $d = \phi m$. (4.21)

This means that BBO/EE also considers the correlation (dependency) among the candidate solutions and hence is also suitable for non-separable problems. Hence, the proposed BBO/EE will also be able to perform better than the other contemporary optimization algorithms in those cases too where the knowledge about the problem structure is required for problem solving.

Definition 4: An ecosystem H_p is a group of p habitats.

Definition 5: Immigration rate $\lambda(HSI): R \rightarrow R$ is a non-increasing function of HSI and is proportional to the likelihood that SIVs from neighboring habitats will migrate into H_i [Simon, D., 2008].

Definition 6: Emigration rate $\mu(HSI): R \rightarrow R$ is a non-decreasing function of HSI and is proportional to the likelihood that SIVs from habitat H_i will migrate into neighboring habitats [Simon, D., 2008].

Definition 7: Evolution rate $\alpha(HSI): R \rightarrow R$ is a non-increasing function of HSI and exhibits the pattern of a rectangular hyperbola. α_i is proportional to the likelihood that new SIVs (species) evolve in the habitat H_i during the optimization process itself.

The above phenomenon can be modeled in the optimization process by the inclusion of SIVs which were not applicable for the given habitat before and may become relevant and important with time during the optimization process. This can be done by making the coefficient of those SIVs in the solution vector as one. These SIVs can be selected by using the attached weights, the SIVs for which the weights satisfy the threshold value, (taken as one-third of the maximum value) are said to be relevant SIVs for the given habitat and are considered in the solution vector for HSI calculation. Thus, the mechanism is to prioritize the SIVs according to their relative importance in HSI calculation and based on the priority, the SIVs are then migrated.

Definition 8: Extinction rate $\beta(HSI): R \rightarrow R$ is a non-decreasing function of HSI and exhibits a parabolic curve. β_i is proportional to the likelihood that SIVs become extinct in the habitat H_i during the optimization process.

The above phenomenon can be modeled in the optimization process by the deletion of the SIVs which are no longer applicable for the given habitat from the calculation of HSI. For example, if a species for which a particular SIV was important is extinct now due to natural evolution or human activity or epidemic, then that particular SIV is no longer relevant to HSI calculation. This can be done by making the coefficient of those SIVs in the solution vector as zero. These SIVs can be selected by using the attached weights, the SIVs for which the weights are less than a set threshold value, (taken as one-third of the maximum value) are said to be less significant SIVs for the given habitat and hence can be considered as extinct in the solution vector for HSI calculation.

Definition 9: Habitat modification operator $M(\lambda, \alpha, \mu, \beta): H^n \rightarrow H$ adjusts the habitat based on the ecosystem H^n . The probability that the habitat H_i is modified is proportional to its immigration rate λ_i and evolution rate α_i . The probability that the source of the modification comes from H_j is proportional to the emigration rate μ_j , and the extinction rate β_i however, since the interdependent species emigrate as a group hence, the emigration rate will be determined by the factor $\varphi \cdot \mu_i$. Habitat modification can loosely be described as follows.

Select H_i with probability $\propto (\lambda_i + \alpha_i)$

For $j = 1$ to n

 Select H_j with probability $\propto (\varphi \cdot \mu_i + \beta_i)$

 Randomly select an SIV δ from H_j

 Replace a random SIV in H_i with δ

End For

Definition 10: Mutation $T(\lambda, \alpha, \mu, \beta): H \rightarrow H$ is a probabilistic operator that randomly modifies habitat SIVs based on the habitat's *a priori* probability of existence. A habitat's probability of existence is computed from λ, α, μ and β as discussed before. Mutation can be described as follows.

For $j = 1$ to m

 Use λ_i, α, μ_i and β_i to compute the probability P_i

 Select SIV $H_i(j)$ with probability $\propto P_i$

 If $H_i(j)$ is selected

 Replace $H_i(j)$ with a random SIV δ .

 End For

End For

Definition 11: An ecosystem transition function $\omega = (m, n, t, \lambda, \mu, \alpha, \beta, M, T): H^n \rightarrow H^n$ is a function that modifies the ecosystem from one optimization iteration to the next and can be written as follows in equation (4.22):

$$\omega = \lambda^n \circ \alpha^n \circ \mu^n \circ \beta^n \circ M^n \circ HSI^n \circ T^n \circ HSI^n \quad (4.22)$$

In other words, the ecosystem transition function begins by computing the immigration, evolution, emigration and extinction rates of each habitat. Then, habitat modification is performed on each habitat, followed by an HSI recalculation. Finally, mutation is performed, followed again by an HSI recalculation for each habitat.

Definition 12: A BBO/EE algorithm is defined as $BBO/EE = (I, \omega, C)$ and $I^n: \varphi \rightarrow \{H^n, HSI^n\}$ is a function that creates an initial ecosystem of habitats and computes each corresponding HSI. ω is the ecosystem transition function defined earlier, and $C: H^n \rightarrow \{true, false\}$ is a termination criterion.

The steps of the BBO/EE algorithm are given below:

- [1] Initialize the BBO/EE parameters for the given problem using definitions 1 and 2 and also the maximum species count, the maximum migration rates, maximum evolution and extinction rates, the maximum mutation rate and an elitism parameter.
- [2] Initialize the set of habitats where each habitat represents a candidate solution to the given problem.
- [3] For each habitat, map the HSI to the number of species, the immigration rate, the emigration

rate, the evolution rate and the extinction rate (as described in fig. 4.3(b) and definitions 5, 6, 7 and 8). The definition of HSI function will be different for each habitat, each definition based on a different set of SIVs which simulates the effect of the factors of evolution and extinction in BBO/EE.

- [4] Probabilistically use immigration, emigration, evolution and extinction to modify each non-elite habitat, then re-compute each HSI (as explained in definition 9).
- [5] For each habitat, update the probability of its species count using (2). Then, mutate each non-elite habitat based on its probability and re-compute the HSI of each habitat (as explained in definition 10). In each iteration of the optimization process, the definition of HSI gets changed for the purpose of HSI recalculation, the decision of choice of HSI function definition being based on these other factors.
- [6] Go to step (3) for the next iteration. This loop can be terminated after a predefined number of iterations, or after an acceptable problem solution has been found.

4.2.2. BBO/EE Algorithm

The BBO is extended by modeling the factors of extinction / evolution as the factors which influence the HSI of a solution by incorporating additional constraints in the optimization process that may arise during the optimization process itself. These other factors are modeled in BBO/EE as several definitions of HSI of a habitat, each definition based on a different set of SIVs which simulates the effect of these other factors. In each iteration of the optimization process, one of these definitions of HSI is chosen for the purpose of HSI calculation. The algorithm for the proposed BBO/EE technique is described next (figure 4.5). For each habitat $\{H_1, H_2, \dots, H_n\}$ in the ecosystem, the corresponding HSI values are computed based on the function

$$HSI_i = f(h_i)$$

where i signifies the i^{th} habitat ranging from 1 to n . The no. of SIVs in the set h_i can range from $h_1, h_2, \dots, h_{\phi m}$. Since each solution vector is characterized by a different set of SIVs, the HSI of a given habitat i will be calculated as given by equation (4.19). The SIVs which are not applicable for the given habitat are removed during the calculation of HSI by setting the coefficient of those SIVs in the solution vector to zero. The choice of applicable SIVs is dependent upon the characteristics of the candidate solution under consideration. This means that the SIVs are migrated after considering the newly evolved and now extinct SIVs for each habitat. The immigration rate λ , the evolution rate α , the emigration rate μ , and the extinction rate β is computed for each habitat based on *the* HSI function and select the habitat with the probability $\propto (\lambda_i + \alpha_i)$. Also, a neighboring habitat H_j is selected with probability $\propto (\phi \cdot \mu_i + \beta_i)$ for emigration. The immigration, emigration, evolution and extinction rates are then probabilistically used to modify each non-elite habitat by replacing the random SIV out of the

modified set of SIVs in H_i with SIV from H_j through migration. For each habitat, the probability of its species count is updated. Then, each non-elite habitat is mutated based on its probability and the HSI of each habitat is recomputed using the corresponding HSI function until an acceptable solution has been found.

4.3. Extended models of species abundance in biogeography

In this section, the extended model to the Simon's model of species abundance in biogeography based optimization proposed by Ma in his work [Ma, H., et. al., 2009] is presented. To explore the influence of migration curve shape on optimization performance, he listed six representative migration models. The work extends these migration models by considering factors other than migration (as discussed in the previous section) to formulate the extended species abundance model. The modified forms of each of the six migration models proposed by Ma [Ma, H., et. al., 2009] are hence presented and called as the generalized species abundance models. The first three are extended versions of the linear models [Ma, H., et. al., 2009], which mean that the immigration rate λ_k and the emigration rate μ_k are linear (or at least affine) functions of the number of species k , and the other three are extended versions of the nonlinear models [Ma, H., et. al., 2009].

4.3.1. Linear models

Model 1: (constant immigration and linear emigration model)

The emigration rate μ_k is linear with respect to the number of species k , and the immigration rate λ_k is constant and equal to half of I . The proposed extended species abundance curve is illustrated in figure 4.3(b) and the probability curve is shown in figure 4.4(b). The growth rate and the decline rate are as derived below in equations (4.23) and (4.24) respectively.

$$\text{Growth rate } \sigma_k = \frac{I}{2} + \frac{c}{k} \quad (4.23)$$

$$\text{Decline rate } \rho_k = \frac{Ek\phi}{n} + 2\sqrt{nk} \quad (4.24)$$

Model 2: (linear immigration and constant emigration model)

The immigration rate λ_k is linear with respect to the number of species k , and the emigration rate μ_k is constant and equal to half of E . The proposed extended species abundance model is illustrated in figure 4.3(b) and the probability curve is shown in figure 4.4(b). The growth rate and the decline rate are as derived below in equations (4.25) and (4.26) respectively.

$$\text{Growth rate } \sigma_k = I \left(1 - \frac{k}{n}\right) + \frac{c}{k} \quad (4.25)$$

$$\text{Decline rate } \rho_k = \frac{E\phi}{2} + 2\sqrt{nk} \quad (4.26)$$

Model 3: (linear immigration and linear emigration model)

This model was presented in the original BBO work [Simon, D., 2008]. The immigration rate λ_k and the emigration rate μ_k are linear functions of the number of species k in the habitat. The

proposed extended species abundance model is illustrated in figure 4.3(b) and the probability curve is shown in figure 4.4(b). The growth rate and the decline rate are as derived below in equations (4.27) and (4.28) respectively.

```

Begin
/* BBO parameter initialization */
Create a random set of habitats (population)  $H_1, H_2, \dots, H_n$  ;
/*Compute the corresponding HSI values */
  While ( $i \leq n$ )
    Compute the HSI for habitat  $H_i$  based on the corresponding function  $f(H_i)$ .
  End while
/* End of BBO parameter initialization */
While not  $T$  do /*  $T$  is the termination criterion */
  Compute the immigration rate  $\lambda$  , the evolution rate  $\alpha$ , the emigration rate  $\mu$ , and extinction
  rate  $\beta$  for each  $H_i$  based on HSI ;
  For each habitat (solution)
    For each SIV (solution feature)
      Select habitat  $H_i$  with probability  $\propto (\lambda_i + \alpha_i)$ 
      If  $H_i$  is selected then
        Select  $H_j$  with probability  $\propto (\varphi \cdot \mu_i + \beta_i)$ 
        If  $H_j$  is selected then
           $H_i(SIV) \leftarrow H_j(\delta)$  ;
        End if
      End if
      Select  $H_i(SIV)$  based on mutation probability  $m_i$ 
      If  $H_i(SIV)$  is selected then
        Replace  $H_i(SIV)$  with a randomly generated SIV  $\delta$ .
      End if
    End For
    Re-compute the HSI values
  End For
End While
End

```

Figure 4.5: Algorithm for Biogeography Based Optimization with Extinction / Evolution (BBO/EE)

$$\text{Growth rate } \sigma_k = I \left(1 - \frac{k}{n}\right) + \frac{c}{k} \quad (4.27)$$

$$\text{Decline rate } \rho_k = \frac{Ek\varphi}{n} + 2\sqrt{nk} \quad (4.28)$$

4.3.2. Non-Linear Models

Ma proposed three nonlinear migration models, a trapezoidal migration model, a quadratic migration model and a sinusoidal migration model. The extended forms of each of these nonlinear models are presented.

Model 4: (Trapezoidal migration model)

The immigration rate λ_k and the emigration rate μ_k are trapezoidal functions of the number of species k . The proposed extended species abundance model is illustrated in figure 4.3(b) and the probability curve is shown in figure 4.4(b). The growth rate and the decline rate are as derived below in equations (4.29) and (4.30) if $k \leq i'$ and (4.31) and (4.32) if $k \leq n$ respectively.

If $k \leq i'$ ($I \geq E$)

$$\text{Growth rate } \sigma_k = I + \frac{c}{k} \quad (4.29)$$

$$\text{Decline rate } \rho_k = \frac{2Ek\varphi}{n} + 2\sqrt{nk} \quad (4.30)$$

Else $k \leq n$ ($I < E$)

$$\text{Growth rate } \sigma_k = 2I \left(1 - \frac{k}{n}\right) + \frac{c}{k} \quad (4.31)$$

$$\text{Decline rate } \rho_k = E\varphi + 2\sqrt{nk} \quad (4.32)$$

where i' is the smallest integer that is greater than or equal to $(n + 1)/2$; that is, $i_0 = \text{ceil}((n + 1)/2)$.

Model 5: (quadratic migration model)

The migration rate λ_k and μ_k are convex quadratic functions of the number of species k . The proposed extended species abundance model is illustrated in figure 4.3(b) and the probability curve is shown in figure 4.4(b). The growth rate and the decline rate are as derived below in equations (4.33) and (4.34) respectively.

$$\text{Growth rate } \sigma_k = I \left(\frac{k}{n} - 1\right)^2 + \frac{c}{k} \quad (4.33)$$

$$\text{Decline rate } \rho_k = E\varphi \left(\frac{k}{n}\right)^2 + 2\sqrt{nk} \quad (4.34)$$

Model 6: (sinusoidal migration model)

The migration rate μ_k and λ_k are sinusoidal functions of the number of species k , resulting in a bell-like shape. The proposed extended species abundance model is illustrated in figure 4.3(b) and the probability curve is shown in figure 4.4(b). The growth rate and the decline rate are as derived below in equations (4.35) and (4.36) respectively.

$$\text{Growth rate } \sigma_k = \frac{I}{2} \left(\cos\left(\frac{k\pi}{n}\right) + 1\right) + \frac{c}{k} \quad (4.35)$$

$$\text{Decline rate } \rho_k = E\varphi\left(-\cos\left(\frac{k\pi}{n}\right) + 1\right) + 2\sqrt{nk} \quad (4.36)$$

It is to be noted that the extended species abundance model as well as the corresponding probability curve is same for all the six migration models. This is because of the overshadowing effect of the evolution and extinction curves on the migration curves. The evolution curve which is basically a rectangular hyperbola overshadows the effect of the species growth by immigration and the extinction curve which is the inverse of a y-axis centered parabola scales down the effect of the species decline by emigration. This means that evolution and extinction play a major role in the determination of the species count on a habitat at a given time instant and hence the species growth rates and decline rates are primarily determined by evolution and extinction of species rather than by migration under normal circumstances. Also, the addition of the dependency factor in the determination of the emigrating species further diminishes the role of emigration in the determination of species count on a habitat and hence, the proposed species abundance model is as presented in figure 4.3(b) and the corresponding probability curve is as given by figure 4.4(b). Therefore, the species count on a habitat is modeled not as a function of migration only but as a function of migration (modeling the effect of dependency factor) as well as evolution / extinction of species on the habitat under consideration.

4.4. Performance Analysis

This section presents the optimization results of the BBO/EE by running it on a set of standard benchmark functions with different cost estimates and compares its performance with the original BBO and other evolutionary optimization methods. The performance analysis of the extended models of BBO are also presented and compared with the Simon's model of BBO on standard benchmark functions in this section.

4.4.1. Performance analysis of BBO/EE

In order to explore the benefits of the BBO/EE model, its performance is compared with the original BBO and seven other population-based optimization methods of ACO [(Blum, C., 2005), (Dorigo, M., et. al., 2004)], DE [(Price, K., et. al., 1997), (Storn, R., 1999)], ES [(Back, T., 1996), (Parmee, I., 2001)], GA [(Goldberg, D., 1989), (Michalewicz, Z., 1992), (Muhlenbein, H., et. al., 1993)], PBIL [Baluja, S., 1994], PSO [(Clerc, M., 2006), (Eberhart, R., et. al., 2004), (Eberhart, R., et. al., 2001)] and SGA [Khatib, W., et. al., 1999] on the set of 14 standard benchmark functions in order to demonstrate the performance of the proposed BBO/EE. The benchmarks that are minimized are functions that are representative of those used in the literature for comparison of optimization methods and have been chosen since these are the similar functions as used in the original work [Simon, D., 2008] so as to better demonstrate the improvement in the results and for a fair and easy comparison of results. For a very clear comparison of the capabilities of this new modified BBO model, comparison results of the performance of the proposed algorithm with the other recently modified versions of BBO such

as the BBO/RE [Dawei, D., et. al., 2009], BBO/ES [Dawei, D., et. al., 2009], BBO/ES/RE [Dawei, D., et. al., 2009] and LBBO [Simon, D., et. al., 2013] which have been developed very recently and had shown significant improvements in the performance of the original BBO are also added. More information about the benchmark functions, including their domains, can be found in [Back, T., 1996], [Parmee, I., 2001] and [Yao, X., et. al., 1999]. The benchmarks were compared by implementing integer versions of all the optimization algorithms in Matlab as done in the original work by Simon [Simon, D., 2008]. As in [Simon, D., 2008], each algorithm is considered to have a population size of 50, an elitism parameter of 2, and ran for 50 generations. Each algorithm was run over 50 Monte Carlo simulations on each benchmark to get representative performances.

The parameter settings that were used are set to be the same as that used in the original BBO work which compared the representative performances of the other algorithms with that of BBO. Thus, the same results have been used for the existing algorithms as published in previous works [(Dawei, D., et. al., 2009), (Simon, D., 2008), (Simon, D., et. al., 2013)] and the published results are reproduced here, however, the values are normalized with respect to the newly added BBO algorithms. For BBO/EE, the habitat modification probability = 1, immigration probability bounds per gene = [0,1], evolution probability bounds per gene = [0,1], extinction probability bounds per gene = [0,1], step size for numerical integration of probabilities = 1, maximum immigration, emigration, evolution and extinction rates for each island = 1, and mutation probability = 0, and threshold value = 1/3. For BBO/RE, BBO/ES, BBO/ES/RE, the parameters are chosen as: number of SIVs per habitat is taken as 20, habitat modification probability = 1, mutation probability = 0.005, elitism parameter = 1 and threshold of immigration refusal is taken as 0.5. For LBBO, a mutation probability $pm = 0.01$ is used, and the LBBO parameter κ is set to an integer that is randomly distributed between 1 and D for each migration where D is the problem dimension [Simon, D., et. al., 2013]. For BBO, the habitat entropy = 0, immigration entropy bounds per gene = [0,1], step size for numerical integration of entropies = 1, maximum immigration and migration rates for each island = 1, and mutation entropy = 1 [(Dawei, D., et. al., 2009), (Simon, D., 2008)]. For ACO, the initial pheromone value was taken as $1E - 6$, pheromone update constant as 20, global pheromone decay rate as 0.9, exploration constant as 1, local pheromone decay rate as 0.5, pheromone sensitivity as 1, and visibility sensitivity as 5 [(Blum, C., 2005), (Dorigo, M., et. al., 2004)]. For DE, a weighting factor of 0.5 and a crossover constant equal to 0.5 [(Price, K., et. al., 1997), (Storn, R., 1999)] is used. For the ES, $\lambda = 10$ offsprings were produced in each generation, and standard deviation $\sigma = 1$ for changing solutions [(Back, T., 1996), (Parmee, I., 2001)]. For the GA, a single point crossover with a crossover probability of 1, and a mutation probability of 0.01 was used [(Goldberg, D., 1989), (Michalewicz, Z., 1992), (Muhlenbein, H., et. al., 1993)]. For PBIL, the

parameters used were, a learning rate of 0.05, 1 good population member and 0 bad population members, an elitism parameter of 1, and a 0 probability vector mutation rate [Baluja, S., 1994]. For SGA, a single point crossover with a crossover probability of 1, and a mutation probability of 0.01 was used [Khatib, W., et. al., 199]. For PSO, an inertial constant = 0.3, a cognitive constant = 1, and a social constant for swarm interaction = 1 was used [(Clerc, M., 2006), (Eberhart, R., et. al., 2004), (Eberhart, R., et. al., 2001)].

Tables 4.1 and 4.2 show the results of the simulations. Table 4.1 shows the absolute best minima and the CPU times found by each algorithm over 50 Monte Carlo runs on standard benchmark functions. Table 4.2 shows the *average* minima found by each algorithm, averaged over 50 Monte Carlo runs. In other words, Table 4.2 shows the *average* performance of each algorithm, while Table 4.1 shows the *best* performance of each algorithm. (Since the normalizations in the tables are based on different scales, so numbers cannot be compared between the two tables). From Table 4.1, it can be seen that the BBO/EE performed the best (on average) on seven of the 14 benchmarks followed by LBBO which performs the best on five of the standard benchmark functions and BBO/ES and SGA which perform best on one benchmark function each. From the table, it is observed that the BBO/EE outperforms the standard BBO on all the benchmark functions which is evident from the fact that the average and the absolute minimal value found by BBO/EE is much lower than that found by the original BBO model for all the 14 benchmark functions. For example, the normalized absolute minima found by BBO/EE model for the Step function is 100 whereas that for BBO is 973 which is a significant difference, that for the Quartic function by BBO/EE is 178 and original BBO is 8081 and similarly for the other benchmark functions too. From Table 4.1, it can also be observed that the CPU time for the BBO/EE has reduced significantly (CPU time = 0.61) over the original BBO model which has a CPU time of 2.4. Also, BBO/EE is compared with the recently developed BBO/RE, BBO/ES, BBO/ES/RE and LBBO, and found that BBO/EE on the whole performs better than these since LBBO shows best performance on 5 benchmarks followed by BBO/ES and SGA which perform the best on one benchmark function each. Also, it can be seen that BBO/EE performs the best on non-separable functions where it almost outperforms every other algorithm for all of the non-separable benchmark functions namely, Fletcher, Griewank, Penalty #1, Penalty #2, Rosenbrock, Schwefel 1.2, Schwefel 2.21, Schwefel 2.24. For the non-separable functions Ackley and Schwefel 2.21, the values are the second best since LBBO is also a good performer on non-separable functions.

This is because with the incorporations of factors of extinction and evolution, the proposed model tries to streamline the process of migration by improving the quality of the solutions preliminarily and by the inclusion of the concept of dependency for the determination of true migration rates. Since, BBO/EE also considers the correlation (dependency) among the

candidate solutions and hence it is also suitable for non-separable problems as demonstrated in this section. Also, going by the conceptual significance of this inclusion of the additional factors which signify the inclusion of additional constraints in the optimization process that may arise during the optimization process itself, the other factors are modeled in the BBO/EE as several definitions of HSI of a habitat, each definition based on a different set of SIVs (depending upon the characteristics of the habitat under consideration) which simulates the effect of these other factors. Hence, the SIVs which are not applicable for the given habitat are removed during the calculation of HSI by making the coefficient of that particular SIV in the solution vector as zero, thereby reducing the computational complexity of the standard BBO which applies the same HSI function for each habitat regardless of the habitat characteristics. This in turn leads to the decrease in the minimal value found by the BBO/EE model and hence faster convergence to the minimal value. From the above discussion and the optimization results in Tables 4.1 and 4.2, it can be concluded that the proposed BBO/EE far outperforms the standard BBO and the other optimization algorithms as well and hence proposes an advanced optimization algorithm.

Table 4.1: Mean normalized optimization results and CPU times on benchmark functions. The numbers shown are the minimum function values found by the algorithms, averaged over 50 Monte Carlo simulations, and normalized so that the smallest number in each row is 100. Note that these are not the absolute minima found by each algorithm, but the average minima found by each algorithm.

	BBO/EE	LBBO	BB/ES	BBO/RE	BBO/ES/RE	ACO	BBO	DE	ES	GA	PBIL	PSO	SGA
Ackley	117	100	126	284	133	607	333	487	657	657	773	640	343
Fletcher	100	150	235	325	118	5065	500	1925	2470	2075	4585	3995	570
Griewank	100	119	128	175	132	239	172	401	1026	761	4172	1508	147
Penalty #1	326	330	442	12152	331	2.22E7	1.16E4	9.70E4	1.26E6	2.46E5	2.82E7	2.09E6	100
Penalty #2	100	110	112	1110	1242	6.956E3	991	8.123E3	5.86E4	3.41E5	7.44E5	8.8E4	139
Quartic	178	100	106	4875	139	9.9E4	8081	3.6E4	2.2E5	8.8E4	1.5E6	2.6E5	3.1E3
Rastrigin	315	100	159	6421	159	1.4E4	3068	1.2E4	1.6E4	1.3E4	1.9E4	1.4E4	4110
Rosenbrock	100	110	120	201	126	2802	167	414	1173	701	3048	845	164
Schwefel 1.2	100	127	150	453	191	1639	811	3172	3448	1347	4916	4.8E4	892
Schwefel 2.21	104	100	110	146	150	185	115	260	186	211	304	205	167
Schwefel 2.24	100	110	153	1074	153	844	123	356	1342	614	1056	816	174
Schwefel 2.26	101	100	102	128	113	119	130	151	155	157	196	157	111
Sphere	256	243	100	3219	268	5.8E4	4292	1.1E4	3.9E4	3.9E4	1.2E5	4.3E4	4678
Step	100	136	110	612	110	2154	973	2623	7060	4785	2.8E4	1.0E4	868
CPU Times	0.61	1.2	1.5	2.1	1.7	3.2	2.4	3.3	2.3	2.1	1.0	2.9	2.1

Table 4.2: Best normalized optimization results on benchmark functions. The numbers shown are the best results found after 50 Monte Carlo simulations of each algorithm, and normalized so that the smallest number in each row is 100. Note that these are the absolute best minima found by each algorithm.

	BBO/EE	LBBO	BBO/ES	BBO/RE	BBO/ES/RE	ACO	BBO	DE	ES	GA	PBIL	PSO	SGA
Ackley	117	100	126	284	133	607	333	487	657	657	773	640	343
Fletcher	100	150	235	325	118	5065	500	1925	2470	2075	4585	3995	570
Griewank	100	119	128	175	132	239	172	401	1026	761	4172	1508	147
Penalty #1	326	330	442	12152	331	2.22E7	1.16E4	9.70E4	1.26E6	2.46E5	2.82E7	2.09E6	100
Penalty #2	100	110	112	1110	1242	6.956E3	991	8.123E3	5.86E4	3.41E5	7.44E5	8.8E4	139
Quartic	178	100	106	4875	139	9.9E4	8081	3.6E4	2.2E5	8.8E4	1.5E6	2.6E5	3.1E3
Rastrigin	315	100	159	6421	159	1.4E4	3068	1.2E4	1.6E4	1.3E4	1.9E4	1.4E4	4110
Rosenbrock	100	110	120	201	126	2802	167	414	1173	701	3048	845	164
Schwefel 1.2	100	127	150	453	191	1639	811	3172	3448	1347	4916	4.8E4	892
Schwefel 2.21	104	100	110	146	150	185	115	260	186	211	304	205	167
Schwefel 2.24	100	110	153	1074	153	844	123	356	1342	614	1056	816	174
Schwefel 2.26	101	100	102	128	113	119	130	151	155	157	196	157	111
Sphere	256	243	100	3219	268	5.8E4	4292	1.1E4	3.9E4	3.9E4	1.2E5	4.3E4	4678
Step	100	136	110	612	110	2154	973	2623	7060	4785	2.8E4	1.0E4	868
CPU Times	0.61	1.2	1.5	2.1	1.7	3.2	2.4	3.3	2.3	2.1	1.0	2.9	2.1

4.4.2. Performance analysis of the extended models of species abundance

In order to explore the benefits of the proposed extended models of BBO, their performance is compared on two standard benchmark functions, the Step function and the Griewank function. Each algorithm had a population size of 50, an elitism parameter of 2 and is run for 50 Monte Carlo simulations on the benchmark to get representative performances.

Tables 4.3 and 4.4 show the absolute best minima and the *average* minima found by each model, averaged over 50 Monte Carlo runs for the Step and the Griewank functions respectively. The tables show the *average* performance of each model, as well as the *best* performance of each model for the step as well as the Griewank functions. It can be observed from the tables that the best and the average performances of each of the proposed extended species abundance models is better than the original Simon's species abundance model [Goel, L., et. al., 2012D] which is evident from the fact that these values are much lower than that found by the original BBO for each of the models. For example, the absolute minima for model 3 is 70 and 9 for the Step and the Griewank functions respectively whereas for original BBO it is 102 and 20, almost half of that of the Simon's model, that for model 2 is 286 and 12 whereas for Simon's model, it is 564 and 16 for the Step and Griewank functions respectively and similarly for the other models too. From Tables 4.3 and 4.4, it can also be observed that the CPU times have not increased substantially as opposed to the decrease in the cost functions that they provide for each of the models. The CPU times are greater than the Simon's model for model 2 and 4.4(a) only for both the Step and the Griewank functions while they are comparable for rest of the other models. Figures 4.6(a) - (f) present the graph plot of the minimum cost values for each of the proposed extended species abundance models of BBO for the Step function over 50 simulations. The color codes are 'red' for the minimum cost graph plot for the proposed extended model and 'green' for the minimum cost values found by the original Simon's model of BBO over 50 Monte Carlo simulations [Goel, L., et. al., 2012D]. From the graphs, it can be easily observed that the absolute minima found by each of the proposed extended models of BBO are lower than the Simon's model.

Table 4.3: Mean optimization results and CPU times on benchmark function 'Step'. The numbers shown are the minimum and the average function values found by the models, averaged over 50 Monte Carlo simulations.

Step function	Models of BBO			BBO		
	Best	Mean	CPU Time	Best	Mean	CPU Time
Model 1	22	793.84	0.5470	103	1081.32	0.5470
Model 2	286	1194.02	0.6410	564	1318.26	0.5160
Model 3	70	1079.74	0.5470	102	1104.14	0.5000
Model4(a)	270	714.90	0.5630	537	1629.44	0.4850
Model4(b)	149	738.72	0.5000	199	986.88	0.5160
Model 5	11	1171.68	0.5310	22	1824.02	0.5150
Model 6	79	396.44	0.5000	182	531.50	0.5310

Table 4.4: Mean optimization results and CPU times on benchmark function ‘Griewank’.

Griewank function	Models of BBO			BBO		
	Best	Mean	CPU Time	Best	Mean	CPU Time
Model 1	3	40.27	2.6250	4	51.36	2.6250
Model 2	12	62.52	2.6400	16	72.16	2.6100
Model 3	9	66.49	2.5940	20	68.38	2.6090
Model4(a)	11	32.17	2.6720	15	63.11	2.6090
Model4(b)	11	44.35	2.5780	17	75.05	2.5940
Model 5	13	65.73	2.5630	16	83.32	2.5780
Model 6	12	83.32	2.5780	15	92.55	2.8120

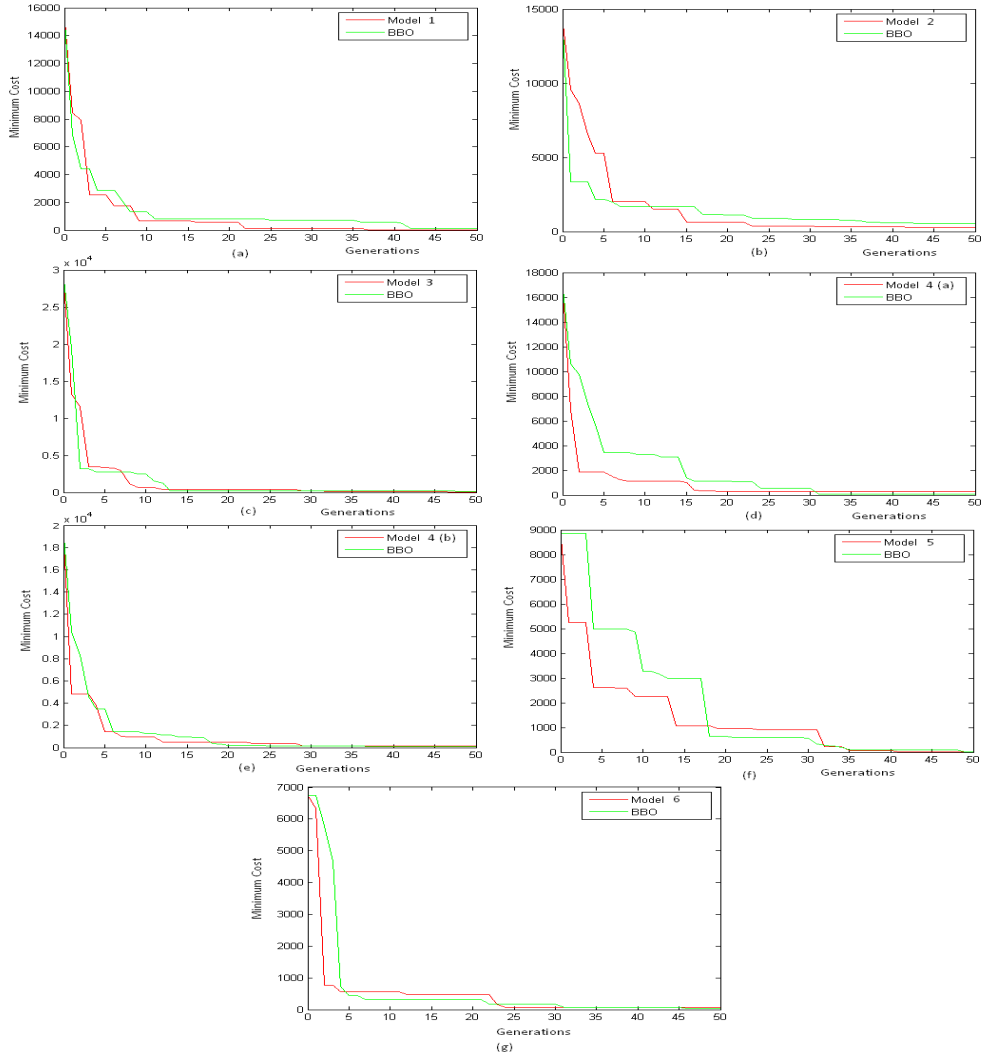


Figure 4.6 (a) - (f): Graph plot of the minimum cost values for each of the proposed extended migration models of BBO over 50 simulations for the Step function. The color codes are: ‘red’ – proposed extended model and ‘green’ – original Simon’s model of BBO.

4.5. Observations

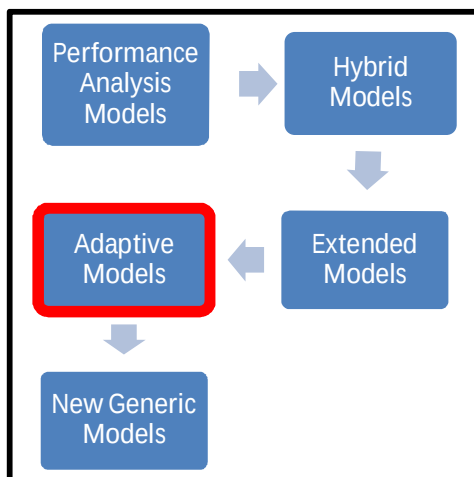
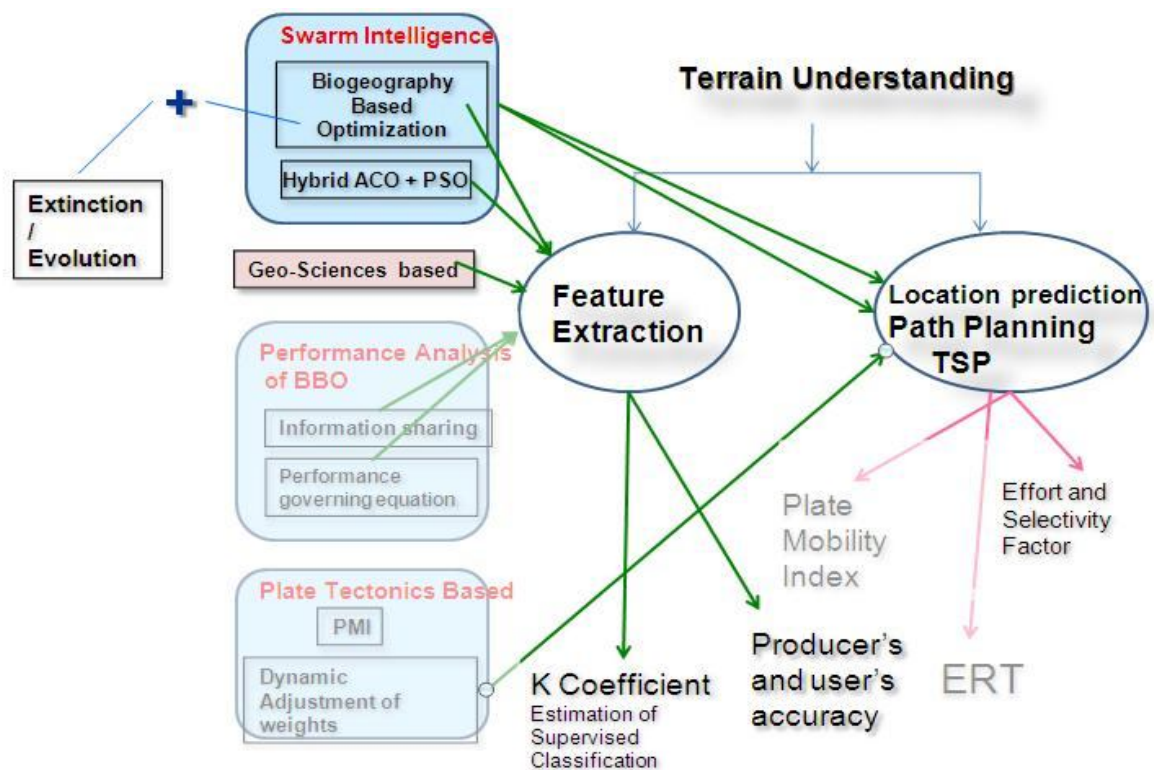
The proposed algorithm incorporates broader definition of species population modeling in classic BBO by taking into account the factors of extinction and evolution which make the HSI function simpler by streamlining the process of migration. With the introduction of the above concepts, the break in analogy between BBO and biogeography that was mentioned in [Simon,

D., 2011] has been successfully incorporated back here. Also, by taking the HSI function to be different for each habitat, a lot of flexibility is brought in the choice of best solution. For example, taking the sensor selection problem for illustration, using the proposed algorithm, sensor selection can be done simultaneously for two different applications i.e. two different universal habitats comprising all possible candidate solutions for each application, at the same time and migrating the best of SIVs across applications for improvement for the solutions obtained for both the applications. If the sensor for aircraft engine health estimation and say, some thermo-static equipment, is to be selected, then there will be some SIVs pertaining to aircraft problem that are not relevant for the sensor selection problem regarding the thermo-static equipment. At the same time, there will be some SIVs that are important for best results in both applications. These SIVs will be the ones that can be migrated across applications for optimization of solutions for both applications. Also, the factor of relevance of these SIVs will be different for both applications. For example, contribution of temperature sensitivity of the sensor may be a linear curve for aircraft problem and a parabolic curve for the thermo static equipment problem. This phenomenon has been modeled in the proposed algorithm by considering that the HSI function for each habitat which is part of the optimization problem is different. The extended forms of each of the six representative migration models that were originally proposed by Haiping Ma [Ma, H., et. al., 2009] as an extension to the work by Simon [Simon, D., 2008] are also presented.

The performance of BBO/EE was demonstrated by running it on a set of 14 standard benchmark functions and comparing its performance with original BBO, LBBO, modified BBO algorithms of BBO/ES, BBO/RE, BBO/ES/RE and other recent EAs. From the results obtained, it was found that the BBO/EE outperformed the standard BBO, the recent modified BBO algorithms and the other EAs on these benchmarks. The performance of each of the extended models of BBO was also demonstrated by running them on standard benchmark functions and found that the average convergence of each of the proposed extended species abundance models is faster than the original Simon's species abundance model of BBO hence leading to better optimization results than the original model. It is noted that CPU time is a bottleneck to the implementation of many population-based optimization algorithms. However, the proposed extended models of BBO require CPU times comparable with that of the Simon's model of BBO which further adds to the strengths of these models.

This chapter has introduced a new optimization tool that can hopefully be applied to many different types of problems by adapting the extended definition of HSI to the given problem. [Alpaydin, E., 2004]. The next chapter considers this aspect and deals with the building of an adaptive framework from the BBO/EE optimization algorithm generated in this chapter to suit the terrain understanding applications.

Adaptive Models Of Nature Inspired Computational Intelligence



Chapter 5

- 5.1. Two-phase recommender system design based on BBO/EE for Intelligent Battlefield Preparation
- 5.2. BBO/EE for Land Cover Feature Extraction
- 5.3. Observations

This chapter presents the adaptive models of the biogeography based optimization (BBO) with extinction and evolution and also introduces a new concept of efforts required in migration from a low HSI solution to a high HSI solution for optimization in BBO. The chapter is divided into 3 sections. Based on the concepts introduced, Section 5.1 presents a two - phase recommender system architecture design for intelligent preparation of the battlefield which is the targeted optimization problem in this case. The proposed recommender system serves a dual purpose by predicting the deployment strategies of enemy troops in the battlefield and also finding the shortest and the best feasible path for attack on the enemy base station.

Section 5.2 demonstrates the applicability of the BBO/EE optimization algorithm to the problem of land cover feature extraction of satellite images and compares the performance of the BBO/EE based hybrid ACO2/PSO/(BBO/EE) feature extractor with the original hybrid classifier and twelve other optimization methods that have been adapted till date for land cover feature extraction application. Section 5.3 concludes the work and discusses the observations made.

5.1. Two-phase recommender system design based on BBO/EE for Intelligent Battlefield Preparation

In this modern warfare era, it has become necessary to have the ability to anticipate the likely actions of the enemy's maneuver and the troop mobilization strategies. The ability to accurately predict the deployment strategy of the enemy troops – to identify the most likely location for the enemy to position their forces and finding a feasible route is of critical importance to the commanders in the battlefield [Tzu, S., 6th Century B.C.]. There are a few works available in this field of natural computing for situation awareness, which are modified to suit the purpose of intelligent battlefield planning, some of which are disaster situation awareness [Jaiswal, R. K., et. al., 2002], terrorist camps detection [Yasar, G. S., et. al., 2008] and electronic situation awareness [Patrick, C. H., 2006]. However, no computing method is available for the prediction of enemy base station directly since the known available methods for the prediction of the enemy base camps are still the conventional methods employed by military which involve manual study of geography of the area and various other factors. To understand and simulate the best available routes for attacking the enemy forces and to predict the ability of the enemy to position forces and mount attack, this section has put forward natural computation algorithms.

Based on the concepts introduced in the extended model of BBO and its mathematics in the previous chapter, a two - phase recommender system architecture is designed for intelligent preparation of the battlefield which is the targeted optimization problem in this case [(Goel, L., et. al., 2012C), (Goel, L., 2014)]. For the military application, a combination of three natural

computation techniques of ACO [(Blum, C., 2005), (Dorigo, M., et. al., 2004), (Otero, F. E. B., et. al., 2008), (Parpinelli, R. S., et. al., 2002)], PSO [(Bratton, et. al., 2007), (Clerc, M., 2006), (Eberhart, R., et. al., 2001)] and BBO [(Currie, D. J., 2012), (Ergezer, M., et. al., 2009), (Ma, H., et. al., 2009), (Ma, H., et. al., 2011), (Ma, H., 2010), (Simon, D., 2008), (Simon, D., et. al., 2009), (Simon, D., 2011)] are used which are all a part of population based optimization. The reason for the selection of these techniques is the versatile and distributive nature of swarm intelligence techniques. The proposed recommender system serves a dual purpose by predicting the deployment strategies of enemy troops in the battlefield and also finding the shortest and the best feasible path for attack on the enemy base station. Hence, the proposed recommender system can be used to improve the traditional approaches, since they lack the ability to predict the destination and can only find a suitable path to the given destination, leading to coordination problems and target misidentification which can lead to severe casualties. The designed system can be of major use for the commanders in the battlefield who have been using traditional decision making techniques of limited accuracy for predicting the destination. Using the above natural computation technique can help in enabling the commanders in the battlefield for intelligent preparation of the battlefield by automating the process of assessing the likely base stations of the enemy and the ways in which these can be attacked, given the environment and the terrain considerations.

The section is organized into five subsections, where the first section presents the proposed problem solving framework based on extended form of biogeography based optimization. The second section discusses the proposed two-phase recommender system architecture design based on the extended model of BBO. This section includes fixing of the parameters for both the phases of the recommender system, the system as well as functional architecture and the mathematical formulation of the recommender system. The third section presents the design and implementation of the two-phase recommender system on synthetic scenarios. For the purpose, two scenarios have been studied and demonstrated in the section namely plain/desert region and the mountainous region. The fourth section presents the performance comparison results of the recommender system with the traditional decision making as well as the other recent optimization techniques. The last section discusses the commercial & scientific applications of the algorithm.

5.1.1. Proposed Problem Solving Framework Based on BBO/EE

The proposed extended form of BBO/EE is presented in figure 5.1. It is based on the extended species abundance model of biogeography that was proposed in the work [(Goel, L., et. al., 2012B), (Goel, L., et. al., 2012D), (Goel, L., et. al., 2013B)] by considering the fact that the characteristics of each habitat vary greatly and hence, comparing all the habitats using the same

set of SIVs may be misleading and also may not lead to an optimum solution. Thus, the HSI value of a habitat is a function of different combinations of SIVs depending upon the characteristics of the habitat under consideration as detailed and discussed in the work [(Goel, L., et. al., 2012B), (Goel, L., et. al., 2012D), (Goel, L., et. al., 2013B)].

Let the problem dimension = t .

Now, t is constant for all solutions. The no. of decision variables is equal to t for all solution vectors which is analogous to SIVs. HSI is defined as the function of SIVs i.e. $HSI_i = f_i(SIVs)$ where i signifies the i^{th} habitat ranging from 1 to n . Let the set of SIVs range from $x_1, x_2, \dots, x_t$.

Since, each solution vector is considered to be characterized by a different set of SIVs, the HSI of a given habitat i will be calculated as given by equation below.

$$\begin{aligned} HSI_i &= \alpha_1 x_1^{\theta_1} + \alpha_2 x_2^{\theta_2} + \alpha_3 x_3^{\theta_3} + \dots + \alpha_t x_t^{\theta_t} \\ &= \sum_{k=1}^t \alpha_k x_k^{\theta_k} \end{aligned} \quad (5.1)$$

The SIVs which are not applicable for the given habitat is removed during the calculation of HSI by making the coefficient of those SIVs in the solution vector as zero. For the purpose, the factor of relevance of a particular SIV in the HSI calculation is calculated on different habitats and the SIVs for which the relevance factor value is below 50% are eliminated. This will be useful when the problem of finding the best suited destination enemy base station is considered on different terrain types such as plain or a desert region and mountainous region since some SIVs may be more relevant on a plain / desert terrain and may not be applicable for a hilly terrain and vice versa.

Also, the contributions of each SIV for the HSI calculation may be different for different habitats (representing different terrain types in this case) and hence the HSI function may change dynamically depending upon the type of terrain under consideration and its characteristics. This is well modeled through the factor α that is added in the HSI calculation for a given habitat. The factor α is also called the weight assigned to each SIV in the HSI calculation for a given habitat.

In the initial stage of the algorithm, no. of species = no. of candidate solutions to the problem and each of these solutions are put in the universal habitat. The algorithm also considers an isolated habitat with zero SIVs. This habitat represents the state of no solution in the initial stage of the algorithm. A standard habitat which is analogous to an ideal solution with ideal characteristics is also considered. This habitat is called the feature habitat and consists of the collection of standard SIVs (or ideal SIVs). The feature habitat is used for comparison with the candidate solutions in the universal habitat. Next, the HSI of the feature habitat is calculated

based on its SIVs and hence the ideal HSI value (SIV_{ideal}) and the threshold HSI value ($SIV_{threshold}$).

Now the HSI for the candidate solution is calculated based on its SIVs using the corresponding HSI formula and the HSI of the habitat (base station) is compared with the threshold HSI. The threshold HSI denotes that the candidate solutions of the universal habitat can only be made feature habitat after migration if their HSI falls within this range i.e. it is greater than the threshold HSI). If the HSI is greater than this value, then the SIVs to be migrated are found out and the effort required for migration is calculated using the weights assigned to those SIVs (on a scale of 1-5). Probabilistically, the immigration and emigration are used to modify the non elite solution by immigrating the ideal SIVs of the feature habitat that match with the SIVs of the candidate input solution of the universal habitat to the universal habitat & emigrating the SIVs of the candidate input solution that do not match with the ideal SIVs of the feature habitat from the universal habitat to the isolated habitat. This is the effort required in migrating the SIV's during the computation for the selection of the best solution. The effort required for each candidate solution is calculated by using equation (5.2).

$$Effort = \sum_i (SIV_{ideal} - SIV_i) \times \alpha_k \quad (5.2)$$

where SIV_{ideal} is the ideal value of the i^{th} SIV and SIV_i is the actual value of the i^{th} SIV and i ranges from $1 - n$. (depending upon the no. of factors (SIVs) considered)

If the HSI is less than the threshold, the SIVs of this candidate solution are transferred to isolated habitat. This input solution will not be considered further and is removed from the universal habitat. The algorithm repeats the steps for each of the candidate solutions in the universal habitat. Next, the selectivity factor for the remaining solutions in the universal habitat is calculated. The term selectivity factor is defined as the factor which determines the selectivity of the candidate solution to be chosen as the best solution. The solution which has the highest selectivity factor is most likely to be the best solution and will be recommended by the system as output. The selectivity factor is calculated as given by equation (5.3).

$$Selectivity\ Factor = \frac{(HSI_i - HSI_{Threshold})}{Effort} \quad (5.3)$$

where 'i' ranges from $1 - n$. Hence, the proposed algorithm incorporates the changes in the definition of HSI function during the optimization process.

5.1.2. Two-Phase Recommender System Architecture Based on modified BBO/EE model

This section presents the proposed two-phase recommender system design. The section begins by giving some important relevant definitions for the first and the second phase of the recommender system. Next, the section presents the functional architecture of the two-phase

Begin

/ Ecosystem initialization */*

Create a universal habitat consisting of all the candidate solutions to be considered also called as the species (population) $H_1, H_2, \dots, H_n$ which comprises H_u ;

Create a feature habitat H_f (ideal solution) and an isolated habitat H_i with zero species;

*/*Compute the corresponding HSI values */*

While ($i \leq n$)

Compute the HSI for each species H_i in H_u based on the function $f_i(HSI) = f_i(SIVs)$

$$HSI_i = \sum_{k=1}^t \alpha_k x_k^{\theta_k}$$

Where $\alpha_k = 0$ for the k for which the factor of relevance f_r is zero.

Also, calculate the HSI for each of the feature habitats H_j .

End While

/ End of parameter initialization */*

For each species H_i in the universal habitat H_u

Select a species from H_u

For each of the feature habitat H_j .

Migrate the specie H_i to H_j .

If $HSI_i > HSI_{Threshold}$

Find out the SIVs to be migrated and the effort required for migration using the weights assigned to those SIVs.

$$Effort = \sum_i \sum_k \alpha_k \cdot (x_{ideal(k)} - x_k)$$

Else

Transfer the SIVs of this solution to the isolated habitat H_i */* remove this solution from the universal habitat */*

End If

End For

End For

Calculate the selectivity factor for each of the remaining solutions in the universal habitat.

$$Selectivity\ Factor = \frac{(HSI_i - HSI_{Threshold})}{Effort}$$

The candidate solution with the highest selectivity factor is the best solution.

Figure 5.1: Extended BBO/EE based on HSI function modeling.

recommender system and the algorithms for both the phases. The mathematical formulation of the recommender system which takes as input the candidate base stations and gives as output the best feasible destination base station and the most suitable path of attack is also developed.

A. Definitions

The section identifies some important relevant definitions of the parameters originally proposed by Dan Simon [Simon, D., 2008] which have been modified to suit the purpose of enemy base station prediction and thereafter finding the most feasible path of attack [(Goel, L., et. al., 2010A), (Goel, L., et. al., 2012C), (Simon, D., 2008)]. These parameters define the preprocessing steps of the two-phase recommender system design.

I. First phase of the Recommender System

Definition 1: Suitability Index Variables (SIVs): To predict the best suited destination for the enemy to position its forces, the following SIVs [(Goel, L., et. al., 2010A), (Goel, L., et. al., 2012C), (Simon, D., 2008)] are defined. Features that correlate with HSI for the prediction include factors [(Goel, L., et. al., 2010A), (Goel, L., et. al., 2012C), (Simon, D., 2008)] which are called as SIVs. A suitability index variable $SIV \in C$ is an integer that is allowed in a habitat. In accordance with the problem statement and the expert knowledge of the military domain, the following SIV's are taken into consideration for deciding the HSI for the base stations. Each of these factors are weighted on a scale of 1 – 5 where 1 – 5 are decoded as below depending upon their values in the ideal and the actual scenario.

- [1] Insignificant/very low
- [2] Low
- [3] Average
- [4] High
- [5] Very high

The factors are:

- a. *Transportation network availability (N).* The more the accessible the area is with respect to transport facilities, the greater are the chances of it being selected as the probable base station for the enemy to position their forces since the transportation of logistics and mobile forces would be easier. In case of a mountainous region, the ease of deployment of air traffic is a major factor in the consideration of the most likely base station of the enemy since the areas which are nearby to an airfield network are more suitable for air lifting of forces such as para-commandos and launching of aircrafts and are also favorable for the replenishment of arms and ammunition. Hence, the transportation network availability should be high in order for a base station to be selected and hence, HSI is directly proportional to N ($HSI \propto N$).

- b. *Concentration of signal traffic (S)*. Signal traffic is undesirable since it can lead to congestion and jamming situations and hence, HSI is inversely proportional to S ($HSI \propto 1/S$).
- c. *Safety and fast deployment (D)*. Areas where the enemy can safely deploy their troops Deployment of enemy troops with minimum amount of effort invested in terms of time are more likely to become the destination base station for the enemy to position its forces. Hence, HSI is directly proportional to D i.e. ($HSI \propto D$).
- d. *Enemy and firepower range constraints (F)*. Firepower range is a constraint since an enemy's base station has an arc of influence within which the arms and ammunition are effective on the war theatre. The enemy's base station which has more firepower range is more probable for being selected by the enemy as their base station. Hence, HSI is directly proportional to the enemy's firepower range constraints. ($HSI \propto F$)
- e. *Concentration of enemy stores and logistics (L)*. Concentration of logistics like ammunition & communication equipments, tanks, missiles, boffer guns & other strategic assets play an important role in the selection of the destination base station. The more the concentration of logistics in a particular area, the more is that candidate base station likely to be the enemy's base station for positioning their troops. Hence, HSI is directly proportional to L ($HSI \propto L$).
- f. *Ease of setting up forward base stations and maintaining all-time, all-weather connectivity with the headquarters (C)*. Better connectivity with the neighboring areas is a desirable factor in the selection of a base station as the destination base station. Hence, HSI is directly proportional to the connectivity factor ($HSI \propto C$).
- g. *Mobility factor or the difficulty factor (M_F)* (depends upon the terrain). HSI is directly proportional to M_F i.e. ($HSI \propto M_F$). M_F can be measured by analyzing the evenness of the area. This can be determined by considering the neighboring area of the small area taken into consideration at a time. If the neighboring area was of similar height than the area was considered even. The levels were decided by considering the percentage of area (around the area under consideration) that is even. The scale from 1 – 5 is defined in Table 5.1.

Table 5.1: Scales for Mobility Factor

1	Very Low	Less than equal to 11%
2	Low	12% to 23%
3	Medium	24% to 44%
4	High	45% to 67%
5	Very High	Greater than equal to 68%

- h. *Enemy's ability to influence the friendly base stations (I)*. Enemy's ability to influence the friendly base station is also an important factor while considering the location suited for the enemy to position its forces. The areas that are geographically located in areas where the

enemy can get an in-depth access into the territory of the friendly forces is more suitable to be chosen. Hence, $HSI \propto I$.

- i. *Aspect (Direction / Angle) (H)*. Aspect can be measured in terms of alignment of the area under consideration. The area which is more vertically aligned towards the enemy is more suitable for becoming the likely base station for the enemy troop deployment since it can act as a natural shield for them. The aspect can be categorized in the range of $0 - 360$ degrees at an interval of 45 degrees as $0 - 45, 45 - 90, 90 - 135, 135 - 180, 180 - 225, 225 - 270, 270 - 315, 315 - 360$ degrees. Based on these categories the scales for H can be classified as given in Table 5.2 ($HSI \propto H$).

Table 5.2: Scales for Aspect ratio

1	Least desired	Category 6, 7, 8
2	Less Desired	Category 5
3	Moderately desired	Category 4
4	Highly desired	Category 1, 3
5	Extremely desired	Category 2

- j. *Enemy's ability to counter maneuver and counter deployment of troops (A)*. The enemy's ability to counter-maneuver the enemies can be measured by the area covered under slope. More is the area covered under slope the greater is the probability of locating a base station there since the visual access improves on such regions. In mountainous regions, this area under slope plays a major role since more is the area under slope, more is the dominating height ground and hence more suitable for setting up of a base station. The area covered under slope can be classified into subcategories as: less than 5% , $6 - 10\%$, $11 - 16\%$ and greater than 16% are covered under slope. Hence, $HSI \propto A$. The scales are defined as in Table 5.3.

Table 5.3: Scales for area covered under slope

1	Very low	$< 5\%$
2	Low	$6 - 10\%$
3	Medium	$11 - 15\%$
4	High	$\geq 16\%$

- k. *Areas of Strategic importance (G)*. The areas which are strategically important for the enemy troops are more likely to be selected. An area can be of strategic importance for the enemy to select it as the base station if it gives in depth access into the friendly base station's territory. These are also the areas where the enemy can retrieve strategically important statistics through geo-stationary satellite observations and such strategically important areas are more suitable for the setting up of a base station. The strategic importance of an area is measured in terms of the elevation of the area which can be calculated through the Digital Elevation Model (DEM) values by fixing ranges on a scale of

1 – 5. However, since a region with lower elevation is more desirable, hence, $HSI \propto 1/G$. These ranges are categorized as in Table 5.4.

Table 5.4: Scales for areas of strategic importance

1	Very low	247 to 256
2	Low	257 to 278
3	Medium	279 to 325
4	High	326 to 419
5	Very High	420 to 593

- l. *Ease of Terrain Activities (T)*. Terrain where activities like mine clearance, bridging, track improvement, vehicle movement, etc. can be performed with ease and hence, $HSI \propto T$ where T depends upon the terrain (terrain can be urban (U), rocky (R), vegetation (V), water (W) or barren land (B)) and can be measured on a scale of 1 – 5 for each of the type of terrain as in Table 5.5.

Table 5.5: Scales for ease of Terrain activities

1	Very low	$\leq 11\%$
2	Low	12 – 23%
3	Average	23-44%
4	High	45-67%
5	Very High	$\geq 68\%$

- m. *Perception of enemy strengths, weaknesses and intent (P)*. Ability to perceive the enemy's intents is a measure of the cognitiveness of the commanders in the battlefield. The perception ability is an important factor in the choice of the destination base station. Hence, the base station which provides relatively lower perception of enemy strengths and weaknesses is more likely to be selected by the enemy and hence, $HSI \propto 1/P$.

Each SIV is assigned a weight value according to its priority place in deciding the best location for the enemy to position themselves. SIV's are assigned equal weights if they have the same role and equal priority in deciding the location. The weights assigned to each of these factors are as given in Table 5.6 below (based on military inputs).

Definition 2: Habitat Suitability Index: A habitat $H \in (SIV^m)$ is a vector of m integers that represent a feasible base station for the enemy. A *Habitat Suitability Index* $HSI: H \rightarrow R$ is a measure of the goodness of the solution that is represented by the habitat. In the algorithm, the HSI is first calculated for the ideal base station and then for the other candidate base stations using the formulation derived in equation (5.4) which has been derived from the interpretation of military data and by analyzing the results of a statistical survey of the military camping areas.

Table 5.6: SIV's and their assigned weights.

S. No.	SIVs	Assigned weight (α_k)
1	Transportation Network Availability (N)	5
2	Concentration of Signal Traffic (S)	4
3	Safety and fast Deployment (D)	4
4	Firepower range constraints (F)	4
5	Concentration of logistics (L)	3
6	Ease of setting up forward base stations and maintaining all-time , all-weather connectivity with the headquarters (C)	4
7	Mobility/Difficulty factor (M_F)	3
8	Enemy's ability to influence the friendly base stations (I)	4
9	Aspect(Direction/Angle) (H)	3
10	Ability to counter-maneuver or counter-deployment of troops (A)	3
11	Areas of Strategic Importance (G)	5
12	Ease of Terrain activities (T) (V, R, U, B, W)	5
13	Perception of Enemy's strengths, weaknesses and intents (P)	5

$$\text{HSI} = \begin{cases} \text{if } (H > 3) \\ \frac{\left((F * L) + (N + D + \frac{M_F}{G} + T + C) + \left(\frac{I}{S}\right) + \left(\frac{H * A}{10}\right) \right)}{P} & \text{(i)} \\ \text{Else} \\ \frac{\left((F * L) + (N + D + \frac{M_F}{G} + T + C) + \left(\frac{I}{S}\right) - \left(\frac{H * A}{10}\right) \right)}{P} & \text{(ii)} \\ \text{where, } T = (V + \text{mod}((6 - R), 4) - U - B - (W - 2)) \end{cases} \quad (5.4)$$

Definition 3: Required Effort: A variable called 'Effort' is also defined which calculates the amount of effort required in migrating the SIV's during the computation for the selection of the most probable base station which enemy is likely to choose and in this case, this is the destination base station which the second phase of the algorithm takes as input. The effort is calculated for each base station (equation (5.2)) where 'i' ranges from 1-13 (depending upon the no. of factors (SIVs) considered).

Definition 4: Ideal Base Station: The term 'Ideal Base Station' is defined as the base station which is the collection of ideal values of each SIV under consideration. The ideal SIV values are determined by a statistical data analysis and through the expert knowledge in the military domain. These values are then normalized so that they range on a scale of 1-5. The HSI calculated for this base station is also ideal and used for comparison in the algorithm.

Definition 5: Threshold HSI: is defined as the minimum HSI value which makes the input/candidate base station an eligible candidate for the further consideration in the algorithm for the choice of the destination base station. If the HSI value for the input base station is less

than the threshold HSI, then the base station is rejected without proceeding further in the algorithm because this base station cannot be an enemy base station.

Definition 6: Selectivity Factor: The term ‘Selectivity Factor’ is defined as the factor which determines the selectivity of the candidate base station to be chosen by the enemy as their base station. The base station which has the highest selectivity factor is most likely to be chosen by the enemy and hence at the end of the processing by the algorithm, this candidate base station will be the one which will be recommended by the system to be considered by the friendly troops as the enemy base station and the destination base station for the next phase of the algorithm to begin with. The selectivity factor is calculated as given by equation (5.3) where ‘i’ ranges from 1 - 13.

Definition 7: Ecosystem: An ecosystem H_n is a group of n habitats. The size ‘n’ of an ecosystem is constant. Here ‘n’ is the no. of candidate base stations of the enemy which are put in the universal habitat plus one standard habitat with ideal SIVs for matching in order to find the best solution; which is called the feature habitat, since it contains all the features which are required in the destination base station.

Definition 8: Migration rates: Immigration rate λ (HSI): $R \rightarrow R$ is a monotonically non-increasing function of HSI and is proportional to the likelihood that SIVs from neighboring habitats will migrate into habitat H_i . Emigration rate μ (HSI): $R \rightarrow R$ is a monotonically non-decreasing function of HSI and is proportional to the likelihood that SIVs from the habitat H_i will migrate to the neighboring habitats.

Definition 9: Habitat modification Ω (λ , μ): $H_n \rightarrow H$ is a probabilistic operator that adjusts habitat H based on the ecosystem H_n . The probability that H is modified is proportional to its immigration rate λ_i , and the probability that the source of the modification comes from H_j is proportional to the emigration rate μ_j . An ecosystem transition function (equation (2.5) of chapter 2) calculates the immigration and the emigration rates followed by habitat modification and HSI calculation in turn followed by mutation and finally HSI recalculation.

II. Second phase of the Recommender System

Definition 1: Pheromone value. T_{ij} - Pheromone value

Definition 2: Heuristic function. $N_{ij} = 1/d_{ij}$, where d_{ij} represents the heuristic information.

D_{ij} –distance between city i and city j

F_{ki} - Feasible neighborhood of the ant k when being at city i .

Definition 3: Effort Factor Calculation.

P_{ij} - Probability of choosing j as the next city when at city i .

‘a’ and ‘b’ are the parameters that determine the relative influence of the pheromone trail and the heuristic information.

- i) If $a = 0$, the closest nodes are more likely to be selected.
 - ii) If $b = 0$, then only pheromone is used to select the next node without using the heuristics.
- Hence, P_{ij} is given by the following equation.

$$Effort = P_{ij} = \frac{[\tau_{ij}]^a [N_{ij}]^b}{\sum_{l \in N_{ij}} [\tau_{il}]^a [N_{il}]^b} \quad \text{if } j \in n. \quad (5.5)$$

where 'n' is the feasible neighborhood of ant 'k' when at city 'i'.

$$Definition 4: Selectivity Factor = \tau_{if}(t+1) = (1 - \rho)\tau_{if}(t) + \sum_{k=1}^m \Delta \tau_{if} \quad (5.6)$$

B. System Architecture

The proposed extended BBO/EE model is implemented for the design of a two-phase recommender system wherein the first phase is the automated prediction of the deployment strategies followed by the enemy troops in warfare coupled with the prediction of the shortest and the safest route for attack on the enemy which forms the second phase of the recommender system. In the second phase of the recommender system design, the proposed model is integrated with the Particle Swarm Optimization technique and the Ant Colony Optimization technique for the computation of global thresholds and the detection of the shortest path respectively. The system architecture of the two-phase recommender system based on the extended BBO/EE model is shown in figure 5.2.

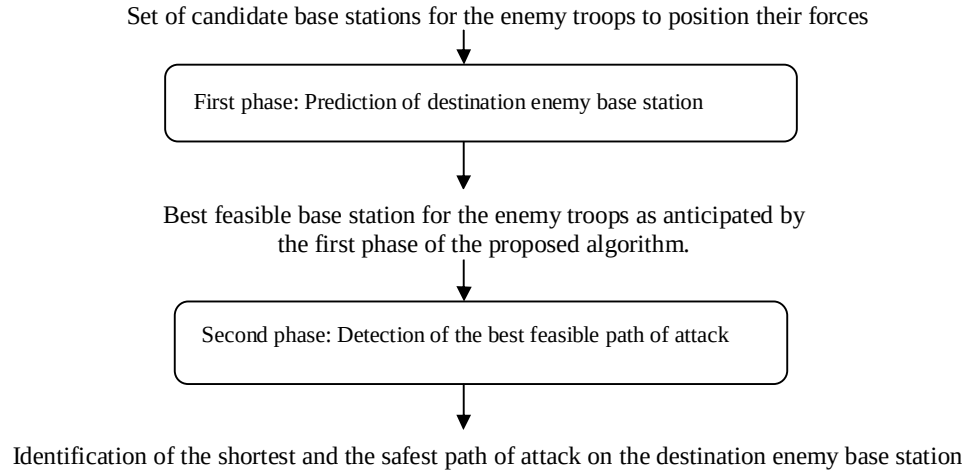


Figure 5.2: Two-phase recommender system architecture based on the extended BBO/EE model

C. Functional Architecture

The generic functional architecture of the two-phase recommender system is given below in figure 5.3 and is explained layer by layer next.

I. First phase of the Recommender System

The first phase of the recommender system is used to predict the deployment strategies of enemy troops by predicting the location of the enemy base station for positioning their forces [(Goel, L., et. al., 2010A), (Goel, L., et. al., 2012C)]. The input to the phase is the set of

suitable base stations for the enemy to position its forces and the output is the most probable enemy base station detection. The functional architecture for the first phase of the model based on extended BBO/EE for enemy base station prediction is detailed next.

Assumptions:

- [1] Initially it is considered that there exists a universal habitat consisting of all the candidate base stations.
- [2] It has been assumed that the SIVs migrate between the universal habitat, feature habitat and the isolated habitat.
- [3] All the candidate base stations are considered exactly once.
- [4] At the completion of the algorithm a base station that is best suited as the destination base station is found.

Input / Output Layer. This layer is used to input the set of candidate base stations suitable for the enemy to position its forces and mount attack (derived based on the drainage pattern of the area under consideration) and output the best feasible base station for the enemy troops to mount attack.

Ecosystem Initialization Layer. This layer is responsible for the initialization of the ecosystem which comprises of the universal habitat, feature habitat and the isolated habitat.

- (a) *Universal habitat Initiator.* This layer is responsible for initializing the set of species here, the suitable base stations which correspond to the set of candidate solutions to the problem of finding the most suited destination in the universal habitat. The no. of species in the universal habitat is equal to the no. of candidate base stations (solutions) to the problem.
- (b) *Isolated habitat Constructor.* This layer is responsible for isolated habitat construction and initialization. The isolated habitat represents the state of no solution in the initial stage of the algorithm. Hence, in the system design, the isolated habitat is initialized with zero SIVs.
- (c) *Feature habitat Constructor.* The feature habitat represents a standard habitat is constructed which is analogous to the ideal solution. In this case, it consists of the collection of standard SIVs (or ideal SIVs) on the basis of historic and intelligence data obtained through a statistical survey of the area, the parameters (SIVs) of which were defined in sections before. The feature habitat is used for comparison with the candidate base stations in the universal habitat.

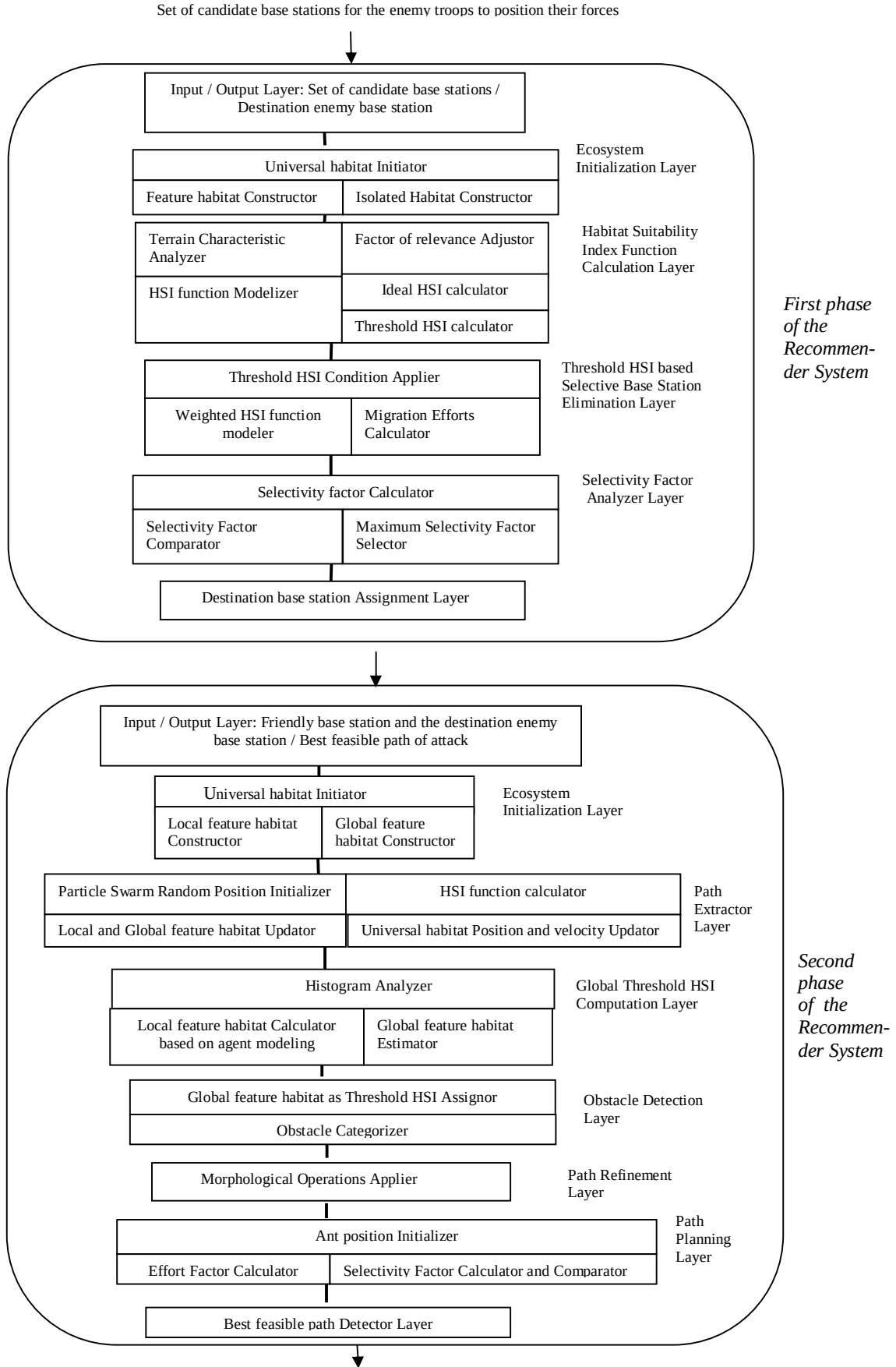


Figure 5.3: Functional architecture of the proposed two-phase recommender system design for intelligent battlefield planning

Habitat Suitability Index Function Calculation Layer. This layer is responsible for calculation of the HSI function value for each of the habitats using the appropriate HSI function applicable depending upon the habitat characteristics. Also, the ideal and the threshold HSI are calculated in this layer. The subunits that are categorized under this layer are described in detail next.

(a) *Terrain Characteristic Analyzer.* The HSI function for each of the base stations in H_u will depend upon the type of terrain under consideration since the SIVs which are applicable on a particular terrain may not be relevant for the other types of terrain. Hence, this layer analyzes the terrain under consideration for the decision about the applicable HSI function. In this case, two case studies have been taken up one, is that of a plain or a desert region (Alwar region in Rajasthan) which have similar characteristics from the viewpoint of base station location prediction, and the other, is of mountainous / hilly region (Mussoorie region in Himachal Pradesh).

(b) *Factor of Relevance Adjustor.* Based upon the terrain characteristics, the HSI for each of the base stations in the universal habitat is defined. This layer models this phenomenon through the factor of relevance adjustor by calculating the factor of relevance of a particular SIV in the HSI function calculation. The SIVs which are less relevant are eliminated from the HSI function of equation (5.4). In this case, the SIVs which have their factor of relevance value, $f_r < 0.5$, are eliminated. Hence, depending upon the habitat characteristics, the HSI function gets modified since some SIVs get eliminated. In the system design, the HSI function is modeled as

$$\begin{aligned} f_i(\text{HSI}) &= f_i(\text{SIVs}) \\ \text{HSI}_i &= f_i(\sum_{k=1}^t k) \end{aligned} \quad (5.7)$$

In the above HSI function,

$$\text{If } f_r(k) < 0.5$$

Then, k is removed from the HSI function and the HSI function is modified to remove all k occurrences from the formula.

(c) *HSI Function Modeler.* Since, the HSI function is not a fixed function of SIVs and changes dynamically, the function chosen for HSI calculation may vary for each base station in H_u and hence this layer models the HSI function. To model this phenomenon for the system design, two HSI functions are defined depending upon the value of H , where H is the enemy's ability to counter-maneuver or counter deployment of troops which in turn depends upon the aspect (direction / angle) of the area under consideration. If the value of H is less than 3, the first function is used else the second.

Hence,

$$\text{HSI}_i = \begin{cases} f_i^1(\sum_{k=1}^t k) & \text{if } H < 3 \\ f_i^2(\sum_{k=1}^t k) & \text{if } H \geq 3 \end{cases}$$

$$\text{where } k = (N, S, D, F, L, C, M_F, I, H, A, G, T, P) \quad (5.8)$$

In this case, the HSI function is formulated as in the above equation. The exact HSI function is as given in equation (i) or (ii) (as applicable) where HSI_i is the HSI of the i^{th} base station in H_u .

- (d) *Ideal HSI Calculator*. This layer is responsible for calculating the HSI of the feature habitat based on its SIVs and hence calculating the ideal HSI value (SIV_{ideal}) using a similar procedure as above.
- (e) *Threshold HSI Calculator*. This layer is responsible for calculate the threshold HSI value. The threshold HSI value is calculated in the system design by taking one-third of the ideal HSI value (based on military inputs).

Threshold HSI based Selective Base Station Elimination Layer. This layer is responsible for elimination of the candidate base stations which do not satisfy the threshold HSI condition. The subunits under this layer are:

- (a) *Threshold HSI Condition Applier*. This layer compares the HSI of the habitat (in the system design, candidate base station) with the threshold HSI. The threshold HSI denotes that the candidate base stations of the universal habitat can only be made feature habitat after migration if their HSI falls within this range i.e. it is greater than the threshold HSI.
- (b) *Weighted HSI Function Modeler*. If the HSI value of the base station under consideration is greater than the threshold HSI value, then this layer finds out the SIVs to be migrated and the effort required for migration using the *weights* assigned to those SIVs (on a scale of 1 to 5). This is how this layer models the concept proposed in the extended BBO that the contributions of each SIV for the HSI calculation may be different and hence the HSI function should reflect this phenomenon. This is well modeled through the factor α that is added in the HSI calculation for a given habitat as given by equation (5.1).
- (c) *Migration Efforts Calculator*. This layer models the factor α in the migration efforts calculation for the choice of the best suited enemy base station amongst the remaining base stations in H_u . The migration rates are probabilistically used to modify the non elite base station (as explained in section 5.1.1) and thus the effort required in the migration of SIVs to and from the universal habitat is calculated from the formula given in equation (5.2) where the weights associated with each SIV are as given in Table 5.6. If the HSI is less than the threshold, transfer the SIVs of this base station to isolated habitat. This base station will not be considered further and is removed from the universal habitat. The steps above are repeated for each of the candidate base stations in the universal habitat.

Selectivity Factor Analyzer Layer. This layer calculates and analyzes the selectivity factor for the remaining candidate solutions in this case, base stations, in the universal habitat.

- (a) *Selectivity Factor Calculator*. This layer calculates the selectivity factor for the remaining base stations in H_u using the formula given in equation (5.3). In this case, these are the set of left over base stations, one among which will be the most likely enemy base station and the output of the first phase of the 2-phase recommender system algorithm.
- (b) *Selectivity Factor Comparator*. This layer is responsible for comparing the selectivity factor of all the base stations so that the base stations with high selectivity factors can be sidelined.
- (c) *Maximum Selectivity Factor Selector*. This layer selects the base station with the highest selectivity factor.

Destination Base Station Assignment Layer. The base station with the highest selectivity factor is the most suitable base station for the enemy troops for their deployment and this layer assigns this base station as the destination base station for the friendly troops to mount attack on.

II. Second Phase of the Recommender System

The algorithm for finding the shortest as well as an obstacle free i.e. best feasible path of attack from the friendly to the destination enemy base station is the second phase of the recommender system. The algorithm begins by the obstacle detection and path extraction of image procedures by automatically categorizing all pixels in the image into roads, forest, barren land, water bodies and settlements [(Goel, L., et. al., 2012C), (Kaur, P., et. al., 2010)]. Next, the paths from image are extracted using PSO algorithm. To do this, appropriate threshold values have to be computed and then segmentation is performed [(Goel, L., et. al., 2012C), (Kaur, P., et. al., 2010)]. Now, obstacles have to be avoided and the goal needs to be reached through the shortest path for which the ACO technique is used [(Blum, C., 2005), (Dorigo, M., et. al., 2004), (Parpinelli, R. S., et. al., 2002)]. The steps followed in the algorithm are explained in detail layer by layer below.

Input / Output Layer. This layer is used to input the friendly base station and the best suited enemy base station obtained as the output of the first phase and output the shortest and the safest path for attacking the enemy base station.

Ecosystem Initialization Layer. This layer is responsible for the initialization of the ecosystem comprising of the universal habitat and the feature habitat.

- (a) *Universal habitat Initiator*: In the system design, this layer initializes a universal habitat consisting of a set of random positions from the source to the destination base station also called as the species (population) $H_1, H_2, \dots, H_n$ which comprises H_u .
- (b) *Local and Global Feature habitat Constructor*. This layer initializes a feature habitat H_g representing the global best position and H_l representing the local best position.

Path Extractor Layer. This layer is responsible for extracting the probable paths for reaching the destination enemy base station by means of detection of inaccessible area and an analysis of the

histogram of satellite image for which it applies the modified particle swarm optimization algorithm.

(a) *Particle Swarm Random Position Initializer*. In this layer, the particles are initialized, which scan their respective regions to find the global best (threshold value). Particle i ($i = 1, 2, 3, \dots, n$) is randomly selected where n is the total number of particles.

(b) *HSI Function Calculator*. This layer is responsible for the HSI function calculation in order to determine the next best position of each particle. Consider $x_i(n)$ and $v_i(n)$ as the current position and velocity of particle i . Let $f(i)$ be the HSI function also called as the fitness function in PSO terminology to calculate the next best position of each particle. If T_x and T_y are the x, y coordinates of the target (accessible area) and S_x and S_y are the x, y coordinates of the source (current position of particle) then HSI function is given by:

$$HSI_i = \frac{1}{\sqrt{(S_x - T_x)^2 + (S_y - T_y)^2}} \quad (5.9)$$

For each candidate position H_i in H_u , this layer computes the HSI at a random position H_i based on the corresponding function $f_i(HSI) = f_i(SIVs)$.

(c) *Local and Global Feature Habitat Updator*. After calculating HSI function from equation above, this layer searches for the next best position. Let $H_l(i)$ represents the local best position of the particle (feature habitat) which has been found so far and it is updated as:

If $(HSI_i < HSI_{i-1})$

/* Update the H_l value */

$$H_l(i+1) = \begin{cases} H_l(i) & f(X_j(i+1)) > f(H_l(i)) \\ X_j(i+1) & f(X_j(i+1)) < f(H_l(i)) \end{cases}$$

End For

/* Find global best position */

$H_g(i)$ represents the global best position of the swarm and is to be updated as:

$$H_g(i+1) = \arg \min f(H_l(i+1))$$

(d) *Universal Habitat Position and Velocity Updator*. This layer updates the position and velocity of each H_i in H_u . After getting local best and global best position from equations above, the following equation is used for updating the position and velocity of each particle:

$$v_j(i+1) = [v_j(i) + c_1 \cdot r_1(H_l(i) - X_j(i)) + c_2 \cdot r_2(H_g(i) - X_j(i))]$$

$$X_j(i+1) = X_j(i) + v_j(i+1)$$

where c_1 and c_2 are positive constants; r_1 and r_2 are random variables. (5.10)

Global Threshold HSI Function Computation Layer. This layer assigns the value of H_g as the threshold HSI value, $HSI_{\text{threshold}}$ to be used as input for the obstacle detection layer. For calculation of the global threshold (average), this layer uses the PSO algorithm.

- (a) *Histogram Analyzer*. To obtain the desired results, this layer plots the histograms of several of the scenes in the database containing objects, for analysis. Based on the observations, the histogram is divided in five main regions. Region I starts from the lowest intensity values to half of the midway of average intensity. For most of the cases, the intensity values on this region cover water bodies. Region II covers intensities that go from approximately half of the midway of the average intensity to midway of average intensity, and typically covers trees. Region III goes from midway of average intensity, to average intensity and covers bright gray objects such as highways. Region IV goes approximately from average intensity to midway of highest intensity and generally covers bright objects like buildings, road dividers, etc. Region V goes approximately from midway of highest intensity to highest intensity and generally covers barren land [(Goel, L., et. al., 2012C), (Kaur, P., et. al., 2010)].
- (b) *Local Feature Habitat Calculator based on Agent Modeling*. In this layer, a numerical simulation model based on agent modeling is devised for calculation of the local feature habitat value. To evaluate the threshold values, ten agents are created which scan their respective regions assigned to them from the image. Every agent scans the image row wise to find local best values for each row.
- (c) *Global Feature Habitat Estimator*. This layer is responsible for estimating the global best value. For the purpose, every agent scans the image row wise to find local best values for each row and eventually evaluates the global best for the assigned region. Among these regional global bests the agents select the most suitable candidate solution by communicating with each other. This is known as the threshold value or the global best (global feature habitat value).

Obstacle Detection Layer. This layer is responsible for calculating the threshold value (global best) which is used in the image to extract the paths as accessible area (barren area) and obstacles (like vegetation, water, rocky area etc.). This corresponds to converting the satellite image into a binary image in which 0 (black=0) predicts inaccessible area and 1 (white=1) predicts the paths. For the purpose, the algorithm proceeds as detailed next.

- (a) *Global Feature Habitat as Threshold HSI Assignor*. This layer assigns the value of H_g as the threshold HSI value, $HSI_{\text{threshold}}$ for the obstacle detection step.
- (b) *Obstacle Categorizer*. This layer is responsible for categorizing the paths as obstacle free or an inaccessible area by using the global threshold HSI value which is the output of the previous layer. The pseudo-code for the obstacle detector is detailed next in figure 5.4.

```

Create a feature habitat  $H_f$  representing the nearest position to  $H_i$ .
For each species  $H_i$  in the universal habitat  $H_u$ 
Select a species from  $H_u$ 
For each of the feature habitat  $H_f$ .
Migrate the specie  $H_i$  to  $H_f$ .
If ( $HSI_i > HSI_{Threshold}$  )
     $H_i$  is a obstacle
Else
     $H_i$  is an accessible area.
End For
End For

```

Figure 5.4: Pseudo-code for the obstacle detector

Path Refinement Layer. This layer is responsible for refining of the paths obtained as output of the previous layers using morphological operations.

- (a) *Morphological Operations Applier.* This layer refines the extracted paths using morphological operations. After the extraction of the paths from the output of the previous layer, it is observed that there are some isolated white pixels that correspond to trees, shadows etc. and therefore for further refinement of the paths, the morphological operations are used [(Goel, L., et. al., 2010A), (Goel, L., et. al., 2012C), (Kaur, P., et. al., 2010)]. Four morphological operations are applied in the system design as given in figure 5.5.

```

Clean: Remove isolated pixels (0's surrounded by 1's).
Majority: Set a pixel to 0 if five or more pixels in its 3-by-3 neighborhood are 0's.
Fill: Fill isolated interior pixels (1's surrounded by 0's).
Holes: Fill the holes in the binary image.

```

Figure 5.5: Morphological operations

- (b) *Path Planning Layer.* This layer is responsible for path planning using ACO. After the operations in the previous layers, many possible paths are extracted amidst the cross-country. In this layer, ants are created and randomly initialized from the starting point. They randomly chose different ways of getting to the target. While moving to the target they deposit pheromone, after reaching to the target they retrace their respective paths. The path of the ant which reaches the starting point earliest is considered the shortest. Now the shortest path is obtained. While finding the way to next node they check for the obstacle, if obstacle occurs they choose the next node. Hence, the ants avoid obstacles coming in their way. On the whole, the safest and shortest path is obtained.
- (c) *Ant Position Initializer.* This layer is responsible for initializing the ants at the starting point wherein they chose random paths from getting to the target. Anti ($i = 1, 2, 3, \dots, m$) denotes an ant in the ant colony where i ranges from 1 to m and m is the total number of

ants. τ_{if} denotes the pheromone between g_i and g_f ($g_i, g_f \in RS$ and $g_i \neq g_f$). RS is research space.

- (d) *Effort Factor Calculator*. This layer is responsible for effort factor calculation for the next nearest node for which the system design proceeds as detailed next. Given g_i is the current position of the ant and g_f is the neighborhood of g_i , or the intervisible area of the ants at g_i , the probability of this ant choosing g_f as its next point is modeled by the effort factor in the proposed extended BBO/EE algorithm. The pseudo-code for the above procedure is presented in figure 5.6.

```

/* Ant Colony Optimization */
For each species  $H_i$  in the universal habitat  $H_u$ 
Select a species from  $H_u$ 
  For each of the feature habitat  $H_f$ .
    If ( $HSI_i > HSI_{Threshold}$ )
      Migrate the specie  $H_i$  to  $H_f$ .
       $Effort = P_{if} = \frac{(\tau_{if})^\alpha (\eta_{if})^\beta}{\sum (\tau_{if})^\alpha (\eta_{if})^\beta}$ 

       $Selectivity\ Factor = \tau_{if}(t + 1) = (1 - \rho)\tau_{if}(t) + \sum_{k=1}^m \Delta \tau_{if}$ 

    End If
  End For
End For

```

Figure 5.6: Pseudo-code for effort factor calculation

where α and β are the tuning parameters for balancing the effects of pheromone τ and heuristics η .

- (e) *Selectivity Factor Calculator and Comparator*. The distance between the current position and the neighborhood is the selectivity factor since here this distance plays the role of the deciding factor in determining the choice of the best path (shortest and the best feasible path) [(Goel, L., et. al., 2012C), (Zhifeng, H., et. al., 2008)]. From equation above, the effort factor is calculated for the next nearest node. Then the pheromone is deposited on length (L) covered and the distance covered so far is calculated also called as the selectivity factor i.e. the total distance covered by each ant was obtained according to equation in the pseudo-code above.

Best Feasible Path Detector Layer. This layer selects the path with the highest selectivity factor which corresponds to the shortest distance among the given paths. The obtained path is the shortest and the safest path. The overall recommender system design is represented through the framework in figure 5.7.

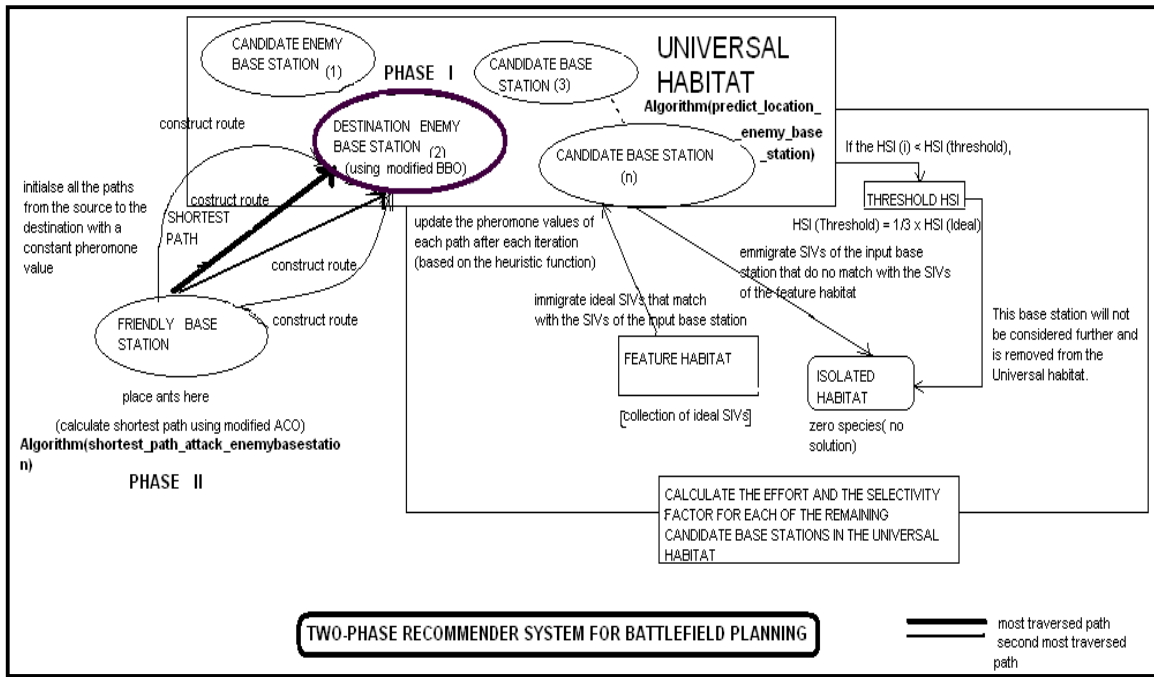


Figure 5.7: Integrated Framework for the Two-Phase Recommender System for Intelligent Battlefield Planning

D. Mathematical Framework for the two-phase recommender system based on extended BBO/EE

The functional architecture of the proposed two - phase recommender system can be represented mathematically in the form of the following equation. The equation presented in figure 5.8 is the combined representation of the output of the first and the second phase of the recommender system.

In the first part of the mathematical equation, the set of candidate base stations suitable for the enemy to position its forces is taken as input and the best feasible base station for the enemy troops to mount attack is the output base station and the corresponding output of the first phase of the recommender system. The output of the first part of the equation is $Max(c)$ which is the base station with the maximum value of selectivity factor and also the output of the first part of the equation. The value of $Max(c)$ is expressed as

$$Max(c) = Max_{c=1}^Z \left(\frac{(HSI_c - HSI_{Threshold})}{Effort} \right) \quad (5.11)$$

where 'z' is the subset of the initial candidate base stations in H_u since it contains the stations for which $HSI_c > HSI_{Threshold}$ where, $HSI_c = f_c^{(1,2)}(\sum_{k=1}^t k)$

In the second part of the mathematical equation, the set of friendly base station and the destination enemy base station for the enemy to position its forces is taken as input and the shortest and the safest path of attack for the enemy troops to mount attack is the output path and the corresponding output of the second phase of the recommender system.

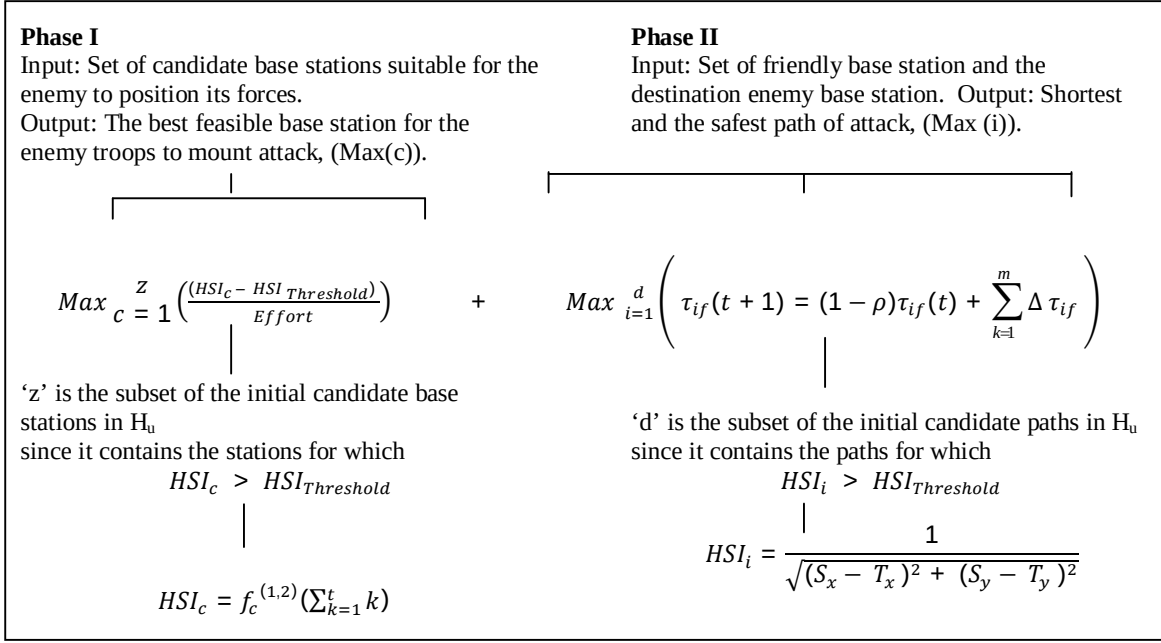


Figure 5.8: Mathematical Framework for the two-phase recommender system based on extended BBO/EE

5.1.3. Two-Phase Recommender System on varied natural terrain scenarios

Now the truth value of the proposed framework is evaluated on different natural terrain scenarios. For the purpose, two scenarios mainly, plain/desert region and the mountainous region are considered. The reason for choosing the above two scenarios is that the SIVs (factors) under consideration for the choice of the most likely enemy base station for the plain as well as the desert region are the same (concluded from military inputs). Also the weights assigned to each of these SIVs is also equal for both the regions and hence the plain and the desert regions are both considered on equal grounds while proving the proposed framework for the two-phase recommender system. The mountainous region is a different scenario since it will have different factors for consideration in making a particular base station more suitable than the other candidate base stations in the hilly region.

The SIVs applicable to a base station are also defined in each of the scenarios. Thus, the implementation of the newly introduced concepts of migration efforts calculation and dynamically modeling the HSI function in the extended BBO is demonstrated by presenting the implementation results of the two-phase recommender system for military application of battlefield preparation on two different terrains that of Alwar area in Rajasthan (Plain / Desert) and that of Mussourie region in Himachal Pradesh (Mountainous / hilly) [Goel, L., et. al., 2012C].

A. Plain / Desert Region

For demonstration of the implementation of the extended BBO/EE based two-phase recommender system, the Alwar area in Rajasthan, India (Desert Scenario) of size 472 X 576 is chosen with the ground resolution of 23.5 meters as shown in Figure 5.9 [Goel, L., et. al., 2012C].

I. First phase of the recommender system

Input / Output Layer: The set of candidate base stations suitable for the enemy to position its forces and mount attack (derived based on the drainage pattern of the area under consideration [Goel, L., et. al., 2012C]) on the Alwar image is taken as input (shown in red in figure 5.10).



Figure 5.9: 7-band FCC satellite image of size 472 X 546 of Alwar area in Rajasthan

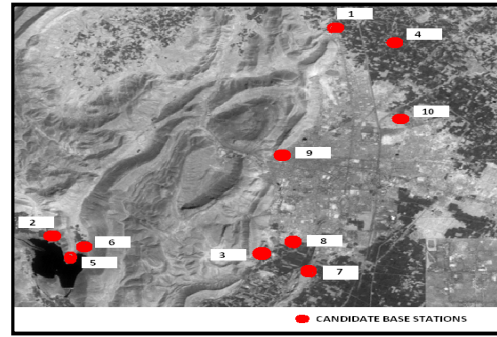


Figure 5.10: 7-band Alwar image with the areas suited as candidate base stations marked in red.

Ecosystem Initialization Layer: In this layer, the ecosystem consisting of the universal habitat, feature habitat and the isolated habitat is initialized.

- (a) Universal Habitat Initializer: H_u consists of 'n' base stations, where $n = 10$, and all the SIV values for each of the base stations taken as input are shown in figure 5.10 (based on military inputs).
- (b) Isolated Habitat Constructor: The isolated habitat is initialized with zero SIVs.
- (c) Feature Habitat Constructor: Table 5.7 fixes the SIVs and the corresponding ideal values on a scale of 1-5 for an ideal base station in a plain / desert region.

Table 5.7: Fixing the SIVs and their values on a plain / desert region

1	Transportation Network Constraints (N)	3
2	Concentration of Signal Traffic (S)	1
3	Safety and fast Deployment (D)	3
4	Firepower Range Constraints (F)	3
5	Concentration of Enemy Stores & Logistics (L)	3
6	Ease of setting up forward base stations and maintaining all-time, all-weather connectivity with the Headquarters (C)	3
7	Ability to influence the friendly base stations (I)	3
8	Ease of Terrain Activities (T)	
	Degree of forest (V)	5
	Degree of Rocky region (R)	3
	Degree of Urbanity(U)	1
	Degree of Barren land (B)	1
	Degree of Water (W)	2
9	Perception of enemy strengths , weaknesses and intents (P)	1

Habitat Suitability Index Function Calculation Layer: This layer is responsible for calculation of the HSI function value for each of the habitats using the appropriate HSI function applicable.

- (a) Terrain Characteristic Analyzer: Since Alwar region is a desert region, some SIVs may not be relevant for the HSI calculation on this type of terrain and hence, the applicable HSI function may be different. The process begins with the terrain characteristic analysis wherein it is observed that in a plain/desert region, the aspect factor is of less importance since it is the hilly regions in which the choice of the base station location is more aspect-driven, the aspect factor is not considered. Also, in a planar region, M_F and G are not applicable. Same is the case with A i.e. the area under the slope which is also applicable in case of a mountainous region [(Goel, L., et. al., 2010A), (Goel, L., et. al., 2012C)].
- (b) Factor of Relevance Adjustor: Hence, the above 4 factors have their factor of relevance values $f_r < 0.5$ and therefore these SIVs are omitted from the HSI function calculation.
- (c) HSI Function Modeler: In this case, the factor H was eliminated in the factor of relevance adjustor layer and hence, the HSI function becomes static since, the parameter governing the dynamic modeling of the HSI function is no longer relevant in a desert scenario. Hence, the HSI function remains the same as in the previous layer and is given by:

$$HSI = [(F * L) + (N + D + T + C) + (I/S)] / P$$

$$\text{where, } T = (V + \text{mod}((6 - R), 4) - U - B - (W - 2))$$

- (d) Ideal HSI Calculator:

$$\text{Ideal HSI value} = [(3 * 3) + (3 + 3 + (5 + 3 - 1 - 1 - 0) + 3) + (3 / 1)] / 1]$$

$$= 27$$

- (e) Threshold HSI Calculator:

$$\text{Threshold HSI} = 1/3 * HSI_{\text{ideal}}$$

$$= 1/3 * 27 = 9$$

Threshold HSI based Selective Base Station Elimination Layer:

- (a) Threshold HSI Condition Applier: This layer compares the HSI of each of the 10 candidate base stations with the threshold HSI value of 9. From Table 5.8, it is observed that the HSI values for the candidate base stations 3, 5, 6 and 9 being 6.33, 7.56, 8.5 and 5.58 respectively is less than the threshold value of 9 and hence these base stations are preliminarily rejected.
- (b) Weighted HSI Function Modeler: Since the HSI value of the base stations 1,2,4,7,8 and 10 is greater than the threshold HSI value, the effort required for migration will be calculated for these base stations using the weights assigned to the SIVs (on a scale of 1 to 5).

(c) Migration Efforts Calculator: For each of the 6 remaining base stations, the migration efforts is calculated as shown in Table 5.8 which depicts the HSI, effort and selectivity factor calculations for each of the candidate base stations.

Selectivity Factor Analyzer Layer: This layer calculates and analyzes the selectivity factor for the remaining candidate base stations in H_u .

(a) Selectivity Factor Calculator: The selectivity factor calculations for remaining base stations are shown in Table 5.8.

(b) Selectivity Factor Comparator: This layer is responsible for comparing the normalized selectivity factor values so that the base stations can be categorized into different probability classes. The findings are categorized into the following ranges as given in Table 5.9.

(1) Since base station 4 is the base station with the normalized selectivity factor = 1, the destination base station is $n = 4$.

(2) Since $0.5 < N_{SF} (n = 1, 2) < 1.0$, base stations 1 & 2 are most likely to be the destination enemy base stations,

(3) Since $0.3 < N_{SF} (n = 10) < 0.5$, base station 10 is a likely destination base station.

(4) Since $0 < N_{SF} (n = 7, 8) < 0.3$, base stations 7 & 8 are the least likely base stations.

(5) Since $N_{SF} = 0$ for base stations 3, 5, 6 & 9, these are not considered as candidate stations. Table 5.10 is the table of categorized output based on Table 5.8 and is represented through color coding as in figure 5.11.

(c) Maximum Selectivity Factor Selector: This layer selects the base station with the highest selectivity factor.

Destination Base station Assignment Layer: The base station with the highest selectivity factor is assigned as the destination base station for the friendly troops to mount attack on.

Table 5.8: HSI, effort and selectivity factor calculation for each of the candidate base stations

S. No. (n)	F	L	N	D	T					C	I	S	P	HSI	Effort	Selectivity Factor (SF)	Norma- lized SF
					V	R	U	B	W								
1	3	3	2	3	5	3	1	1	1	2	3	2	1	24.500	18	0.86	0.54
2	4	4	2	2	3	2	1	1	4	2	5	2	2	13.750	62	0.077	0.05
3	4	2	2	1	3	2	1	1	2	1	2	1	3	6.333	∞	0	0
4	3	3	2	3	5	3	2	1	2	3	3	1	1	25.000	10	1.6	1.000
5	4	2	3	2	4	2	1	1	4	4	2	3	3	7.567	∞	0	0
6	2	2	1	1	5	4	1	1	1	1	4	1	2	8.500	∞	0	0
7	3	2	3	2	3	4	1	2	3	2	4	3	1	17.333	48	0.174	0.11
8	5	4	2	1	3	3	2	1	3	2	3	2	2	14.250	57	0.09	0.06
9	5	4	2	1	1	5	1	5	1	1	4	3	4	5.583	∞	0	0
10	4	2	3	3	4	3	1	2	2	2	3	2	1	21.500	25	0.5	0.31

Table 5.9: Categorization of the base stations based on the normalized selectivity factor values

Normalized Selectivity Factor Range (N_{SF})	Category
$N_{SF} = 1$	Destination Enemy Base Station (Output to the second phase)
$0.5 < N_{SF} < 1.0$	Most likely
$0.3 < N_{SF} < 0.5$	Likely
$0 < N_{SF} < 0.3$	Least likely
$N_{SF} = 0$	Not a base station

Table 5.10: Table of categorized output

Base stations (n)	Category
4	Destination Enemy Base Station
1	Most likely Destination Enemy base stations
10	Likely Destination Enemy base stations
2,7, 8	Least likely Destination Enemy base stations
3, 5, 6, 9	Not a base station

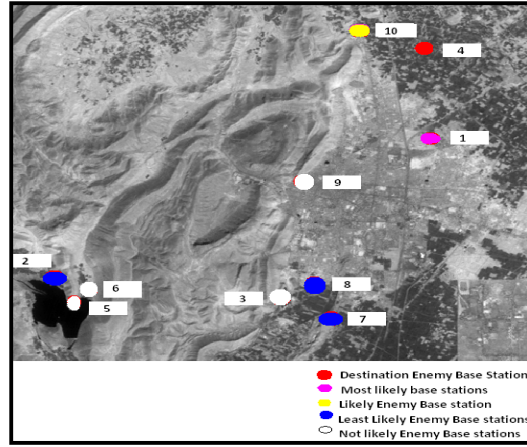


Figure 5.11: 7-Band Alwar Image with the candidate base stations categorized into respective classes.

II. Second phase of the recommender system

The results of the second phase of the recommender system on the Alwar image are described in this section.

Input / Output Layer: This layer is used to input the friendly base station with coordinates (250,7) and the best suited enemy base station obtained which has coordinates (26, 164) as the output of the first phase. Figure 5.12 below represents a portion of the Alwar image with the source (friendly) and the destination station marked in red. The destination enemy base station ($n = 4$) is obtained as the output of the first phase. The source station is chosen to demonstrate the second phase of the recommender system.

Ecosystem Initialization Layer: This layer is responsible for the initialization of the universal habitat and the feature habitat.

- Universal Habitat Initiator:** In the system design, this layer initializes a universal habitat consisting of a set of random positions from the source to the destination base station also called as the species (population) $H_1, H_2, \dots, H_n$ which comprises H_u .
- Local and Global Feature Habitat Constructor:** This layer initializes a feature habitat H_g representing the global best position and H_l representing the local best position.

Path Extractor Layer: This layer analyzes the histogram of satellite image for which it applies the modified Particle Swarm Optimization algorithm.

- (a) Particle Swarm Random Position Initializer: In this layer, ten agents are created and particles are initialized, which scan their respective regions to find the global best (threshold value). Particle_{*i*} (*i* = 1,2,3..., *n*) is randomly selected where *n* is the total number of particles.
- (b) HSI Function Calculator: This layer is responsible for the HSI function calculation in order to determine the next best position of each particle according to equation (5.7).
- (c) Local and Global feature habitat Updater: After calculating HSI function from equation above, this layer searches for the next best position.
- (d) Universal Habitat Position and Velocity Updater: This layer updates the position and velocity of each H_i in H_u .

Global Threshold HSI Function Computation Layer: This layer assigns the value of H_g as the threshold HSI value, $HSI_{threshold}$ to be used as input for the obstacle detection layer.

- (a) Histogram Analyzer: Based on the observations, the histogram is divided in five main regions as described in the functional architecture in the section before.
- (b) Local feature habitat Calculator based on Agent Modeling: The local feature habitat calculations for each of the ten agents are represented in Table 5.11.
- (c) Global feature habitat Estimator: The global feature habitat calculations for each of the ten agents are represented in Table 5.11.



Figure 5.12: Portion of the Alwar image with the source and the target marked in red.

Obstacle Detection Layer: This layer is responsible for calculating the threshold value (global best) which is used in the image to extract the paths as accessible area (barren area) and obstacles (like vegetation, water, rocky area etc.).

- (a) Global feature habitat as Threshold HSI Assignor: This layer assigns the value of H_g as the threshold HSI value, $HSI_{threshold}$ for the obstacle detection step.
- (b) Obstacle Categorizer: The paths extracted after applying PSO is shown in figure 5.13, where black area depicts inaccessible area and white depicts the accessible path.

Path Refinement Layer: The image obtained with the paths refined after applying morphological operations is shown in figure 5.14 below.

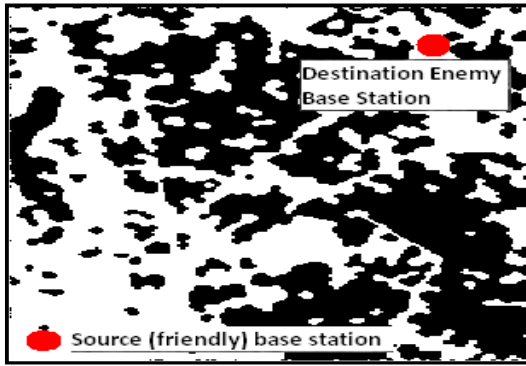


Figure 5.13: Paths extracted after applying PSO



Figure 5.14: Refined paths after morphological operations

Path planning Layer: This layer is responsible for path planning using ACO.

- (a) Ant Position Initializer: This layer is responsible for initializing the ants at the starting point wherein they chose random paths from getting to the target.
- (b) Effort Factor Calculator: This layer is responsible for effort factor calculation for the next nearest node. In the system design, the effort factor is modeled by calculating the HSI function as a function of the shortest distance from the current position to the destination base station which is based on the observation that the destination coordinates is reached when the HSI function or the fitness value of the solution at that step becomes infinity or the shortest distance between the current position and the target becomes zero.
- (c) Selectivity Factor Calculator and Comparator: Table 5.12 presents the table of highest selectivity factor values chosen at each step as the next best position.

Best Feasible Path Detector Layer: This layer is the formation of the path with the highest selectivity factor values which corresponds to the best path between the source friendly base station (250, 7) and the destination base station (26, 164) on the Alwar image as shown in figure 5.15.

Table 5.11: Local and Global feature habitat update in the Global Threshold HSI Computation Layer.

Candidate position 'n'	Global best position for 'n' (H_g)	Local best position (H_l)									
1	88.20	91.08	105.56	90.04	89.72	99.56	99.56	106.88	98.84	101.92	99.24
		94.32	86.60	73.88	73.76	66.48	56.20	61.20	79.20	94.88	95.12
2	54.50	48	60	80	44	84	39	23	27	23	35
		46	27	52	64	109	82	89	85	25	48
3	60.20	29	74	82	39	41	33	27	37	76	68
		78	72	113	52	33	31	37	101	125	56
4	47.30	29	27	25	27	50	89	105	56	50	17
		39	37	85	48	29	33	29	25	37	109
5	49.25	31	35	31	39	39	25	25	35	42	37
		29	33	29	23	27	85	91	97	123	109
6	80.85	27	44	41	29	62	125	119	117	109	105
		123	103	136	95	80	119	80	35	29	39
7	33.95	64	23	35	54	62	31	50	109	44	21
		21	19	19	17	25	11	13	19	19	23
8	85.40	127	89	76	93	103	91	64	54	82	132
		121	127	107	76	37	39	41	72	115	62
9	61.10	50	95	58	89	87	87	93	111	33	29
		25	21	99	87	19	27	42	48	52	70
10	44.35	29	23	29	31	64	109	76	23	23	66
		54	21	33	44	31	91	60	33	37	52



Figure 5.15: Alwar image with the safest and the shortest path from the source (friendly) base station to the target destination enemy base station

B. Mountainous / Hilly Region

The second case study for the demonstration of the concepts of the extended BBO/EE for the development of the recommender system is the Mussourie area in Himachal Pradesh, India (Mountainous scenario) of size 472 X 576 with the ground resolution of 23.5 m as shown in Figure 5.16) [(Goel, L., et. al., 2010A), (Goel, L., et. al., 2012C)]:

I. First phase of the recommender system

Input / Output Layer: The set of candidate base stations suitable for the enemy to position its forces and mount attack (derived based on the drainage pattern of the area under consideration [(Goel, L., et. al., 2010A), (Goel, L., et. al., 2012C)]) are input on the Mussourie image (shown in red in figure 5.17).

Table 5.12: HSI computation and path planning by ACO. Table presents the HSI determination factor, the Euclidean Distance Calculations from the friendly base station to the destination enemy base station

Initial position (Source friendly base station) = (250,7) [gf]									
123.15	122.59	122.04	121.76	121.20	120.65	120.37	119.82	119.27	118.98
118.43	117.89	117.60	117.05	116.50	116.21	115.66	115.12	114.83	114.28
113.74	113.44	112.90	112.36	112.06	111.52	110.98	110.68	110.14	109.60
109.29	108.76	108.23	107.91	107.38	106.85	106.53	106.00	105.48	105.15
104.62	104.10	103.77	103.25	102.73	102.40	101.87	101.36	101.02	100.50
99.99	99.64	99.13	98.62	98.27	97.75	97.25	96.90	96.38	95.88
95.52	95.02	94.51	94.15	93.65	93.15	92.78	92.28	91.79	91.42
90.92	90.43	90.05	89.55	89.07	88.68	88.19	87.71	87.32	86.83
86.35	85.96	85.48	85.00	84.60	84.12	83.65	83.24	82.76	82.30
81.88	81.41	80.95	80.53	80.06	79.61	79.18	78.71	78.26	77.83
77.37	76.92	76.48	76.03	75.58	75.13	74.69	74.25	73.79	73.35
72.92	72.45	72.01	71.59	71.11	70.68	70.26	69.78	69.35	68.94
68.45	68.03	67.62	67.12	66.71	66.31	65.80	65.39	65.00	64.47
64.08	63.69	63.98	63.56	63.16	63.51	63.07	62.65	63.06	62.61
62.17	62.18	61.72	61.27	61.29	60.83	60.37	60.42	59.94	59.48
59.54	59.06	58.59	58.67	58.18	57.71	57.80	57.31	56.82	57.45
56.94	56.44	56.60	56.08	55.57	55.23	54.71	54.20	53.85	53.34
52.84	52.48	51.97	51.48	51.11	50.61	50.12	49.74	49.24	48.76
48.37	47.89	47.41	47.01	46.53	46.07	45.65	45.18	44.72	44.29
43.83	43.38	42.94	42.49	42.05	41.59	41.15	40.72	40.25	39.81
39.40	38.91	38.48	38.08	37.58	37.16	36.77	37.16	36.77	36.40
36.77	36.40	36.06	35.47	35.11	34.79	34.18	33.84	33.53	32.89
32.57	32.28	31.62	31.32	31.05	30.36	30.08	29.83	29.73	29.41
29.12	29.15	28.79	28.46	28.23	27.86	27.51	27.31	26.93	26.57
26.40	26.00	25.63	25.50	25.08	24.70	25.06	24.60	24.17	24.19
23.71	23.26	23.32	22.83	22.36	22.47	21.95	21.47	21.10	20.59
20.12	19.72	19.24	18.79	18.36	17.89	17.46	17.00	16.55	16.16
15.65	15.23	14.87	14.32	13.93	13.60	13.00	12.65	12.37	11.70
11.40	11.18	10.44	10.20	10.05	9.22	9.06	9.00	8.06	8.00
7.28	7.07	7.00	6.71	6.32	6.08	5.83	5.39	5.10	4.47
4.12	4.00	3.16	3.00	2.24	2.00	1.41	1.00	1.00	0.00
Final position (Destination Enemy Base Station) = (26, 164)									

Decreasing Euclidean distance values from the destination station (gif) and correspondingly increasing HSI values and hence, increasing inverse selectivity factor values.
Note: The destination base station coordinates are reached when the distance

$$\sqrt{(S_x - T_x)^2 + (S_y - T_y)^2} = 0$$

Ecosystem Initialization Layer: In this layer, the ecosystem consisting of the universal habitat, feature habitat and the isolated habitat is initialized.

(a) Universal Habitat_INITIALIZER: The universal habitat is considered as consisting of 4n/5 base stations in a mountainous region (since the ratio of no. of base stations in a mountainous



Figure 5.16: Google Earth Image of size 472 X 576 of Mussourie region in Himachal Pradesh



Figure 5.17: Google earth Mussourie image with the suited candidate base stations marked in red.

region to a plain / desert region is 4: 5 derived from military inputs) as shown in figure 5.17 and input all the SIV values for each of the base stations (based on military inputs). Hence, $n = 8$.

(b) Isolated Habitat Constructor: The isolated habitat is initialized with zero SIVs.

(c) Feature Habitat Constructor: Table 5.13 fixes the SIV's and the corresponding ideal values on a scale of 1-5 for an ideal base station in a mountainous / hilly region.

Habitat Suitability Index Function Calculation Layer: This layer is responsible for calculation of the HSI function value for each of the habitats using the appropriate HSI function applicable.

(a) Terrain Characteristic Analyzer: Since Mussourie region is a mountainous region, some SIVs may not be relevant for the HSI calculation on this type of terrain and hence, the applicable HSI function may be different. The algorithm begins with the terrain characteristic analysis wherein it is observed that in a hilly region, the aspect factor is very important since it is the hilly regions in which the choice of the base station location is more aspect-driven. Also, in a hilly region, M_F is also applicable. In a hilly region, a base station is chosen depending upon its strategic importance which is decided on the basis of measuring the elevation of the area under consideration. Ability to counter-maneuver and counter deployment of troops depends upon the area covered under slope which is also applicable in case of mountainous regions and hence, A is considered as an SIV in case of hilly region.

(b) Factor of Relevance Adjustor: Hence, the above 4 factors have their factor of relevance values $f_r > 0.5$ and therefore in the factor of relevance adjustor layer, these SIVs are included in the HSI function calculation.

Table 5.13: SIV and their values for mountainous region

1	Transportation Network Availability (N)	3
2	Concentration of Signal Traffic (S)	1
3	Safety and fast Deployment (D)	3
4	Firepower range constraints (F)	3
5	Concentration of logistics (L)	3
6	Ease of setting up forward base stations and maintaining all-time , all-weather connectivity with the headquarters (C)	3
7	Mobility/Difficulty factor (M_F)	3
8	Enemy's ability to influence the friendly base stations (I)	3
9	Aspect(Direction/Angle) (H)	5
10	Ability to counter-maneuver or counter-deployment of troops (A)	4
11	Areas of Strategic Importance (G)	1
12	Ease of Terrain Activities (T)	
	Degree of forest (V)	5
	Degree of Rocky region (R)	3
	Degree of Urbanity(U)	1
	Degree of Barren land (B)	1
	Degree of Water (W)	2
13	Perception of enemy strengths, weaknesses and intents (P)	1

- (c) HSI Function Modeler: In this case, the factor H is included in the factor of relevance adjustor layer and hence, the HSI function becomes dynamic since, the parameter governing the dynamic modeling of the HSI function becomes relevant in a mountainous scenario. To model this phenomenon for the system design, two HSI functions are defined depending upon the value of H. If the value of H is less than 3, the first function is used else the second.

Hence,

if $(H > 3)$

$$HSI = [[(F * L) + (N + D + (M_F / G) + T + C) + (I / S) + ((H * A) / 10)] / P]$$

else

$$HSI = [[(F * L) + (N + D + (M_F / G) + T + C) + (I / S) - ((H * A) / 10)] / P]$$

$$\text{where, } T = (V + \text{mod}((6 - R), 4) - U - B - (W - 2))$$

- (d) Ideal HSI Calculator:

Since in the ideal case, $H = 5$ i.e. $H > 3$,

$$\text{Ideal HSI value} = [(3 * 3) + (3 + 3 + (3/1) + (5 + 3 - 1 - 1 - 0) + 3 - (5 * 4) / 10)) + (3 / 1)] / 1] = 28$$

- (e) Threshold HSI Calculator:

$$\begin{aligned} \text{Threshold HSI} &= 1/3 * HSI_{\text{ideal}} \\ &= 1/3 * 28 \\ &= 9.33 \end{aligned}$$

Threshold HSI based Selective Base Station Elimination Layer:

- (a) Threshold HSI Condition Applier: This layer compares the HSI of each of the 8 candidate base stations with the threshold HSI value of 9.33. From the Table 5.14, it is observed that the HSI values for the candidate base stations 4 and 7 being 9.1 and 6.433 respectively is less than the threshold value of 9.33 and hence these base stations are preliminarily rejected.
- (b) Weighted HSI Function Modeler: Adding weights to the applicable SIVs, the mathematical formula for calculating the HSI value can be derived in the form of the equation above for HSI in a mountainous / hilly region.
- (c) Migration Efforts Calculator: For each of the 6 remaining base stations, the migration efforts is calculated as shown in Table 5.14 which depicts the HSI, effort and selectivity factor calculations for each of the candidate base stations.

Table 5.14: HSI, effort and selectivity factor calculation for each of the candidate base stations

S. No. (n)	F	L	N	D	M_F	G	T					C	I	S	H	A	P	HSI	Effort	Selectivity Factor	N_{SF}
							V	R	U	B	W										
1	3	3	2	3	3	1	5	3	1	1	1	2	3	2	4	4	1	25.90 (ii)	21	0.81	0.50
2	4	4	2	2	3	3	3	2	1	1	4	2	5	2	3	4	2	15.95 (i)	78	0.089	0.05
3	3	3	2	3	3	1	5	3	2	1	2	3	3	1	5	4	1	26.00 (ii)	10	1.7	1.00
4	2	2	1	1	1	3	5	4	1	1	1	1	4	1	3	2	2	9.10 (i)	∞	0	0
5	3	2	3	2	2	2	3	4	1	2	3	2	4	3	3	4	1	19.53 (i)	62	0.169	0.10
6	5	4	2	1	2	4	3	3	2	1	3	2	3	2	4	2	2	13.95 (ii)	92	0.054	0.03
7	5	4	2	1	1	4	1	5	1	5	1	1	4	3	3	2	4	6.43 (i)	∞	0	0
8	4	2	3	3	4	1	4	3	1	2	2	2	3	2	4	5	1	23.50 (ii)	34	0.43	0.25

Selectivity Factor Analyzer Layer: This layer calculates and analyzes the selectivity factor for the remaining candidate base stations in H_u .

(a) Selectivity Factor Calculator: This layer calculates the selectivity factor values as shown in Table 5.14. The table also shows which of the two formulae is applicable on each of the candidate stations by representing this information in brackets along with the HSI calculation in Table 5.14.

(b) Selectivity Factor Comparator: The findings are categorized based on the normalized selectivity factor values into the following ranges as given in Table 5.15.

- (1) Since base station 3 is the base station with the normalized selectivity factor value as 1, the destination base station is $n = 3$.
- (2) Since $0.5 < N_{SF} (n = 1, 2) < 1.0$, base station 1 are most likely to be the destination enemy base stations,
- (3) Since $0.3 < N_{SF} (n = 10) < 0.5$, base station 8 is a likely destination enemy base station.
- (4) Since $0 < N_{SF} (n = 7, 8) < 0.3$, base stations 2, 5 & 6 are the least likely base stations for the enemy troops to position their forces.
- (5) Since $N_{SF} = 0$ for base stations 4 and 7, these are not considered as candidate base stations. Table 5.16 is the table of categorized output based on results of Table 5.14 and is represented through color coding as in figure 5.18.

Table 5.15: Categorization of the base stations based on the Normalized selectivity factor values

Normalized Selectivity Factor Range (N_{SF})	Category
$N_{SF} = 1$	Destination Enemy Base Station (Output to the second phase)
$0.5 < N_{SF} < 1.0$	Most likely
$0.3 < N_{SF} < 0.5$	Likely
$0 < N_{SF} < 0.3$	Least likely
$N_{SF} = 0$	Not a base station

Table 5.16: Table of categorized output

Base stations (n)	Category
3	Destination Enemy Base Station (Output to the second phase)
1	Most likely Destination Enemy base station
8	Likely Destination Enemy base station
2, 5, 6	Least likely Destination Enemy base stations
4, 7	Not a base station

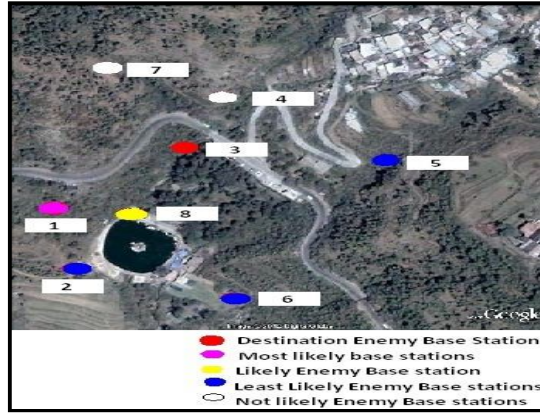


Figure 5.18: Mussourie image with the candidate base stations categorized into respective classes.

(c) Maximum Selectivity Factor Selector: This layer selects the base station with the highest selectivity factor.

Destination Base station Assignment Layer: The base station with the highest selectivity factor is given as output as the most suitable base station for the enemy troops for their deployment.

II. Second phase of the recommender system

The results of the second phase of the recommender system on the Mussourie image are described in this section.

Input / Output Layer: This layer is used to input the friendly base station with coordinates (250,7) and the best suited enemy base station obtained which has coordinates (16, 190) as the output of the first phase. Figure 5.19 below represents a portion of the Mussourie image with the source station and the destination enemy base station marked in red. The destination station is obtained as the output of the first phase where the base station numbered as 3 ($n = 3$) is obtained as the output. The source base station is chosen to demonstrate the second phase of the recommender system.

Ecosystem Initialization Layer: This layer is responsible for the initialization of the universal habitat and the feature habitat.

(a) Universal Habitat Initiator: In the system design, this layer initializes a universal habitat consisting of a set of random positions from the source to the destination base station also called as the species (population) $H_1, H_2, \dots, H_n$ which comprises H_u .

(b) Local and Global Feature Habitat Constructor: This layer initializes a feature habitat H_g representing the global best position and H_l representing the local best position.

Path Extractor Layer: This layer analyzes the histogram of satellite image for which it applies the modified Particle Swarm Optimization Algorithm.

(a) Particle Swarm Random Position Initializer: In this layer, ten agents are created and particles are initialized, which scan their respective regions to find the global best

(threshold value). Particle_{*i*} (*i* = 1,2,3..., *n*) is randomly selected where *n* is the total number of particles.



Figure 5.19: Portion of the Mussourie image with the source and the target marked in red.

- (b) HSI Function Calculator: This layer is responsible for the HSI function calculation in order to determine the next best position of each particle using equation (5.7).
- (a) Local and Global Feature Habitat Updator: After calculating HSI function from equation above, this layer searches for the next best position.
- (b) Universal Habitat Position and Velocity Updator: This layer updates the position and velocity of each H_i in H_u .

Global Threshold HSI Function Computation Layer: This layer assigns the value of H_g as the threshold HSI value, $HSI_{threshold}$ to be used as input for the obstacle detection layer.

- (a) Histogram Analyzer: Based on the observations, the histogram is divided in five main regions as described in the functional architecture in sections before.
- (b) Local Feature Habitat Calculator based on Agent Modeling: The local feature habitat calculations for each of the ten agents are represented in Table 5.17.
- (c) Global feature habitat Estimator: The global feature habitat calculations for each of the ten agents are represented in Table 5.17.

Obstacle Detection Layer: This layer is responsible for calculating the threshold value (global best) which is used in the image to extract the paths as accessible area (barren area) and obstacles (like vegetation, water, rocky area etc.).

- (a) Global Feature Habitat as Threshold HSI Assignor: This layer assigns the value of H_g as the threshold HSI value, $HSI_{threshold}$ for the obstacle detection step.
- (b) Obstacle Categorizer: This layer is responsible for categorizing the paths as obstacle free or an inaccessible area by using the global threshold HSI value which is the output of the previous layer. The paths extracted after applying PSO is shown in figure 5.20, where black area depicts inaccessible area and white depicts the accessible path.

Path Refinement Layer: The image obtained with the paths refined after applying morphological operations is shown in figure 5.21.

(a) Path planning Layer: This layer is responsible for path planning using ACO.

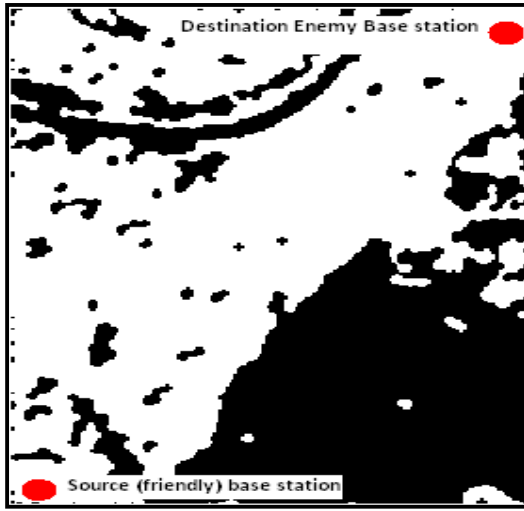


Figure 5.20: Paths extracted after applying PSO

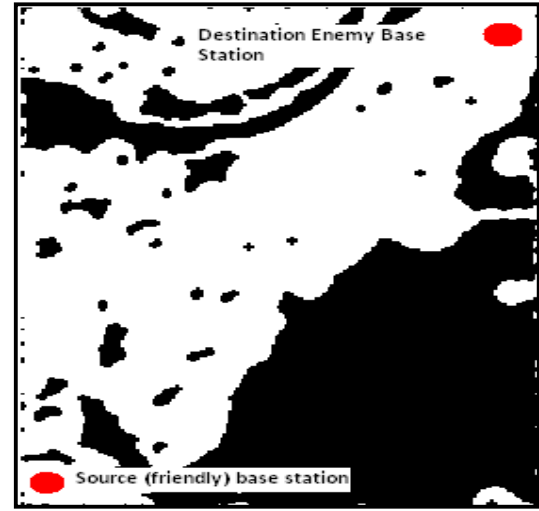


Figure 5.21: Refined paths after morphological operations

(b) Ant Position Initializer: This layer is responsible for initializing the ants at the starting point wherein they chose random paths from getting to the target.

(c) Effort Factor Calculator: This layer is responsible for effort factor calculation for the next nearest node.

(d) Selectivity Factor Calculator and Comparator: Table 5.18 presents the table of highest selectivity factor values chosen at each step as the next best position.

Table 5.17: Local and Global feature habitat Update in the Global Threshold HSI Computation Layer.

Candidate position 'n'	Global best position for 'n' (H_i)	Local best position (H_i)									
		86.60	87.20	86.56	82.16	76.32	75.12	81.52	89.24	90.52	86.84
1	85.83	84.56	86.04	88.16	87.52	85.24	83.60	85.32	89.40	92.12	92.56
2	95.55	116	133	122	101	62	72	103	122	139	136
		91	50	60	71	74	90	97	92	86	94
3	136.85	105	124	136	130	120	120	123	118	119	119
		129	139	140	140	150	165	167	165	168	160
4	128.95	189	108	78	107	140	152	144	132	146	148
		140	138	130	124	123	122	124	121	111	102
5	96.90	103	86	90	116	101	92	96	91	92	91
		99	106	88	79	89	106	111	110	99	93
6	96.05	93	88	79	65	70	82	94	105	114	113
		107	94	100	98	95	91	99	113	111	110
7	49.35	161	150	78	37	46	37	35	40	27	33
		38	41	42	42	35	27	28	28	31	31
8	30.00	29	29	29	28	28	29	30	31	30	29
		30	29	30	28	29	32	33	34	24	39
9	33.80	30	31	40	46	56	37	36	31	28	35
		37	32	29	31	33	32	28	29	28	27
10	89.44	29	36	40	38	33	29	38	31	28	33
		39	45	59	157	200	185	175	168	126	97

Best Feasible Path Detector Layer: This layer is the formation of the path with the highest selectivity factor values which corresponds to the safest and the shortest path between the source friendly base station (250, 7) and the destination base station (16, 190) on the Alwar image as shown in figure 5.22.

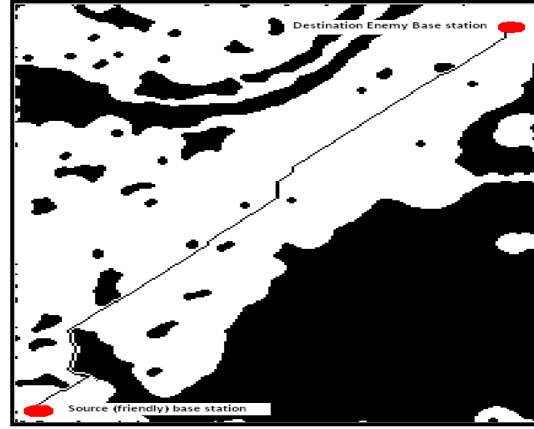


Figure 5.22: Mussourie image with the safest and the shortest path from the source (friendly) base station to the target destination enemy base station

Table 5.18: HSI Computation and Path planning by ACO. Table presents the HSI determination factor, the Euclidean Distance Calculations from the friendly base station to the destination enemy base station

Initial position (Source friendly base station) = (250,7)									
123.15	122.59	122.04	121.76	121.20	120.65	120.37	119.82	119.27	118.98
118.43	117.89	117.60	117.05	116.50	116.21	115.66	115.12	114.83	114.28
113.74	113.44	112.90	112.36	112.06	111.52	110.98	110.68	110.14	109.60
109.29	108.76	108.23	107.91	107.38	106.85	107.38	106.85	106.33	106.85
106.33	105.82	105.48	104.96	104.45	104.10	103.59	103.08	102.73	102.22
101.71	101.36	100.85	100.34	99.99	99.48	98.98	98.62	98.11	97.62
97.25	96.75	96.25	95.88	95.38	94.89	94.51	94.02	93.54	94.89
94.41	93.94	95.29	94.83	94.37	95.71	95.26	94.81	96.18	95.71
95.26	95.77	95.29	94.83	95.38	94.89	94.41	95.02	94.51	94.02
94.67	94.15	93.65	95.19	94.67	94.15	95.52	95.02	94.51	96.04
95.52	95.02	95.72	95.19	94.67	96.25	95.72	95.19	95.96	95.41
94.87	95.69	95.13	94.58	95.44	94.87	94.30	94.63	94.05	93.48
93.23	92.66	92.09	92.42	91.84	91.26	91.61	91.02	90.44	90.80
90.21	89.63	89.40	88.81	88.23	88.60	88.01	87.42	87.80	87.20
86.61	87.01	86.40	85.80	86.21	85.60	85.00	84.81	84.20	83.60
84.02	83.41	82.80	83.24	82.62	82.01	82.46	81.84	81.22	82.33
81.69	81.06	81.57	80.92	80.28	81.47	80.81	80.16	80.72	80.06
79.40	79.98	79.31	78.64	79.25	78.57	77.90	78.52	77.83	77.16
77.79	77.10	76.42	77.78	77.08	76.38	77.08	76.37	75.66	75.66
74.95	74.25	74.25	73.54	72.84	72.84	72.12	71.42	71.42	70.71
70.01	70.01	69.30	68.59	68.59	67.88	67.18	67.18	66.47	65.76
65.76	65.05	64.35	64.35	63.64	62.94	62.94	62.23	61.52	61.52
60.81	60.11	60.11	59.40	58.69	58.69	57.98	57.28	57.28	56.57
55.87	55.87	55.15	54.45	54.45	53.74	53.04	53.04	52.33	51.62
51.62	50.91	50.21	50.91	50.21	49.52	49.50	48.80	48.10	48.08
47.38	46.69	46.67	45.97	45.28	45.25	44.55	43.86	43.84	43.14
42.45	42.43	41.73	41.04	41.01	40.31	39.62	39.60	38.90	38.21
38.18	37.48	36.80	36.77	36.07	35.38	35.36	34.66	33.97	33.94
33.24	32.56	32.53	31.83	31.14	31.11	30.41	29.73	29.70	29.00
28.32	28.28	27.59	26.91	26.87	26.17	25.50	25.46	24.76	24.08
24.04	23.35	22.67	22.63	21.93	21.26	21.21	20.52	19.85	19.80
19.11	18.44	18.38	17.69	17.03	16.97	16.28	15.62	15.56	14.87
14.21	14.14	13.45	12.81	12.73	12.04	11.40	11.31	10.63	10.00
9.90	9.22	8.60	8.49	7.81	7.21	7.07	6.40	5.83	5.66
5.00	4.47	4.24	3.61	3.16	2.83	2.24	2.00	1.41	1.00
1.00	0.00								
Final position (Destination Enemy base station) = (16, 190)									

Decreasing Euclidean distance values from the destination station (gif) and correspondingly increasing HSI values and hence, increasing inverse selectivity factor values.

Note: The destination base station coordinates are reached when the distance

$$\sqrt{(S_x - T_x)^2 + (S_y - T_y)^2} = 0$$

5.1.4. Performance comparison results of the two-phase recommender system with other optimization techniques

This section presents the performance analysis of both the phases of the proposed two-phase recommender system. In order to demonstrate the efficiency of the proposed system accurately which integrates three techniques that of extended BBO/EE based on HSI function modeling, the PSO and the ACO2 techniques, the performance comparison results of each of the two phases of the recommender system are presented separately.

A. First phase of the recommender system:

The first phase of the recommender system which is used to predict the deployment strategies of enemy troops by predicting the location of the enemy base station for positioning their forces is based on the extended BBO/EE [Goel, L., et. al., 2012D]. Hence, the performance of the extended BBO/EE is compared with the traditional decision making techniques that have been used till date for anticipating the probable enemy base stations but have shown limited accuracy. Table 5.20 presents the results of the performance analysis of the extended-BBO/EE technique (first phase) on the datasets of the Alwar and Mussourie regions.

From the results in Tables 5.10 and 5.16, it is observed that the proposed extended BBO/EE model for predicting the enemy base station resulted in the 4th and the 3rd base stations as the destination enemy base stations for the Alwar and the Mussourie images respectively. However, if the traditional decision making would have been applied, the SIV values should have been added up linearly for each of the candidate base stations and the value that is nearest to the ideal value will be chosen as the output base station. Using the above traditional decision making, the HSI values for each of the candidate base stations for the Alwar and the Mussourie regions is calculated as given by Table 5.19.

Table 5.19: HSI calculations using traditional decision making

Base station / Dataset	Alwar ($HSI_{ideal} = 32$) $HSI = \sum_{i=1}^{10} SIV_i$	Mussourie ($HSI_{ideal} = 45$) $HSI = \sum_{i=1}^8 SIV_i$
1	30	42
2	32	47
3	25	45
4	32	35
5	35	44
6	26	45
7	33	47
8	33	46
9	37	--
10	32	--

Table 5.20: Prediction efficiency of traditional decision making v/s the proposed extended BBO/EE model.

Technique used / Dataset		Traditional Decision making					Extended BBO/EE model
Alwar	Output Base stations = [2, 4, 10]	Base station	Probability of selection	Prediction Efficiency	N _{SF} Range	% Efficiency	Output base station = 4
		2	1/3	33%	0-0.3	5%	
		10	1/3	33%	0.3-0.5	13%	
Mussourie	Output Base stations = [3, 6]	6	½	50%	0 – 0.3	8%	Output base station = 3

From the above table, the base stations which are closest to the ideal HSI values are 2, 4 and 10 for Alwar and 3 and 6 for the Mussourie region. This means that the traditional decision making technique could not produce refined results since it resulted in more than one (3 for Alwar and 2 for Mussourie) base stations as output. If the probability of the selection of the 4th and the 3rd base stations is modeled through traditional decision making technique, then the probability of their selection is 1/3 and ½ respectively. In other words, the efficiency of the destination enemy base station prediction of the traditional technique can be written as 33% and 50% for Alwar and Mussourie regions respectively. Also, for the Alwar image, the base stations 2 and 10 which have been outputted as the destination base stations are categorized as ‘Least Likely’ and ‘Likely’ base stations by the proposed system selection with the N_{SF} values as ‘0 – 0.3’ and ‘0.3 – 0.5’ respectively. This means that the efficiency of the traditional classification of these base stations is inaccurate. In other words, the classification accuracy can be derived as a percentage of that predicted by the proposed model as $\frac{1}{3} \times ((0 + 0.3)/2) \times 100$ and $\frac{1}{3} \times ((0.3 + 0.5)/2) \times 100$ i.e. 5% and 13% respectively.

For the Mussourie image, the base station 6 which has been outputted as the destination base stations is categorized as ‘Least Likely’ base stations by the proposed system selection with the NSF value as ‘0 – 0.3’. and the classification accuracy as $\frac{1}{2} \times ((0 + 0.3)/2) \times 100$ i.e. 8% . Also, the secondary outputted base stations 5 and 8 which can be considered as the next closest to the ideal value are categorized as ‘Least Likely’ and ‘Likely’ base stations by the proposed system selection with the NSF values as ‘0 – 0.3’ and ‘0.3 – 0.5’ and the classification accuracy as $\frac{1}{2} \times ((0 + 0.3)/2) \times 100$ and $\frac{1}{2} \times ((0.3 + 0.5)/2) \times 100$ i.e. 8% and 20% respectively. From the above discussion, it is reflected that the results of the traditional decision making techniques are imprecise and do not reflect a true suitability index which is reflected by the lower values of normalized selectivity factor values of the destination base stations outputted by this technique. Moreover, the technique is imprecise as it produces multiple base stations as output of the first phase.

B. Second phase of the recommender system

The second phase of the recommender system which is used to find the best feasible path of attack (the safest (obstacle-free) and shortest) on the enemy base station is based on an integration of extended BBO/EE with the PSO and ACO techniques of path extraction and path planning respectively [(Afaq, H., et. al., 2011), (Zhifeng, H., et. al., 2008), (Kaur, P., et. al., 2010)]. Hence, the performance of the hybrid ACO2/PSO technique that is employed is compared with the results obtained by the use of other optimization techniques [(Alpaydin, E., 2004), (Back, T., 1996), (Zhou, G., et. al., 2008)] such as fuzzy inference mechanism [Kim, B. N., et. al., 1999], genetic algorithms [(Burchardt, H., et. al., 2006), (Goldberg, D., 1989), (Michalewicz, Z., 1992)] etc. that have been used extensively for path planning applications which form the second phase of the recommender system. Table 5.21 presents the results of path planning implementation for the Alwar as well as the Mussourie images using the optimization techniques of Stud Genetic Algorithm (SGA) [Khatib, W., et. al., 1998], Evolutionary Strategy (ES) [(Parmee, I., 2001), (Yao, X., et. al., 1999)], Self-Organizing Feature Maps (SOFM), Fuzzy Inference mechanism (FI) and Genetic Algorithms (GA). The figures in the table are the pixel count of the path planned from the source to the destination base station. In other words, it is the distance of the path planned from the friendly to the destination enemy base station by using each of the said optimization techniques on Alwar as well as the Mussourie region. From the table, it can be seen that the proposed extended BBO based hybrid ACO2/PSO technique employed in the second phase produces the lowest pixel count of 351 for Alwar and 310 for Mussourie region when compared to the other techniques of SGA, ES, SOFM, FI and GAs that were implemented for the path planning application which produce a pixel count of 352, 358, 363, 361 and 358 for Alwar and 311, 316, 321, 319 and 316 for Mussourie region. This means that the path planned by the proposed technique is the shortest and the safest path of attack on the enemy base station for both the datasets used.

Table 5.21: Results of path planning on the given datasets by various optimization techniques.

Optimization technique used / Dataset	Ext-(BBO/EE) /ACO2/PSO based Recommender System	Stud Genetic Algorithm	Evolutionary Strategy	Self-Organizing Feature Maps	Fuzzy Inference	Genetic Algorithm
Alwar	351	352	358	363	361	358
Mussourie	310	311	316	321	319	316

5.1.5. Applications

The proposed hybrid algorithm can be of major use to the military and the commanders in the battlefield as demonstrated in the section. However, the algorithm can be generalized to fit in commercial applications also, such as in predicting the most suitable location for establishing a

manufacturing plant for some industry and thereafter finding the shortest and the safest path for transportation of finished goods and raw materials to and from the manufacturing plant. The algorithm can also be used to predict the best location for exploration of ground water possibilities by taking the ideal features required in it as the feature habitat consisting of the set of ideal SIVs. The set of attributes that comprise of the SIVs are obtained by considering factors such as geology, land form, soil, land use, slope, and linearment, etc. In fact, the algorithm can be generalized in such a way so that it can be used in any application wherein the best suited location and the shortest path for reaching there has to be found out. Thus, the algorithm can be of major use in both military and scientific applications and commercial applications as well.

5.2. BBO/EE for Geo-Spatial Feature Extraction

This section presents the application of BBO/EE for geo-spatial feature extraction. The proposed algorithm has been applied to two different datasets of Alwar and Patalganga regions. The section then finally presents the feature extraction results of BBO/EE based extractor and compares the results obtained with that of the original BBO based feature extractor and the other contemporary evolutionary optimization techniques that have been used for land cover feature extraction on the Alwar image.

The organization of the section is as follows: The first section presents a brief review of the biogeography based land cover feature extraction technique with the HSI function as ‘deviation from mean metric’ as well as ‘entropy’ both. The second section presents the proposed extended BBO/EE based feature extraction, the software architecture and the algorithm. The third section presents the classification results on two different datasets and compares the efficiency with the original model. Also, this section presents a comparison of the proposed classifier with the traditional probabilistic classifiers and other recent computational intelligence based techniques.

5.2.1. Proposed Architecture

This section presents a land cover feature extraction technique based on the extended BBO/EE model [(Goel, L., et. al., 2012B), (Goel, L., et. al., 2012D), (Goel, L., et. al., 2013B)] where the HSI is considered as a function of different combinations of SIVs depending upon the characteristics of the habitat under consideration as an extension to the classical BBO [(Panchal, V. K., et. al., 2009), (Simon, D., 2008)]. In each iteration of the optimization process, one of these definitions of HSI is chosen for the purpose of HSI calculation [Goel, L., et. al., 2013B]. Making use of the proposed hypotheses, the *HSI* of each of the habitats representing the image pixels is calculated using two different functions namely entropy and DMM and hence maximize the classification efficiency achieved by adapting to dynamic changes in the HSI function definition.

Therefore, the following extended model of BBO/EE is proposed in figure 5.23. [Panchal, V. K., et. al., 2009]. The input to the proposed algorithm is the multi-spectral satellite image and the output is the extracted features from the image. The proposed architectural framework for the extended BBO/EE based land cover feature extractor is presented in figure 5.24 below and explained layer by layer next. The above layers of the software architecture of the proposed feature extractor (or classifier) are explained step by step below:

Input layer: A high resolution multi-spectral satellite image is taken as input. For the illustration, two different datasets are used that of Alwar and Patalganga region.

Ecosystem Initialization Layer: This layer is responsible for the initialization of the ecosystem for generating the input equivalence classes initialize the proposed extended biogeography based feature extractor. The sub-layers are described next.

- (a) *Image Grid Division Layer:* In this layer, the satellite image is divided into n clusters.
- (b) *Rough Sets Theory Applier:* For the purpose, the rough sets part of the biogeography based land cover feature extraction algorithm is used [(Ohm, A., et. al., 1997), (Panchal, V. K., et. al., 2009)] which uses multiple bands of the image (which represent the texture of each of the land cover features in the image clearly, hence, clustering is based on texture analysis) during the discretization and partitioning step [(Pawlak, Z., 1982), (Pawlak, Z., 1998)] to generate the unsupervised clusters called the equivalence classes. In the initial state of the ecosystem required to initialize the extended biogeography based feature extractor, each of these equivalence classes is considered as a member of the Universal habitat, ' H_u ', where ' j ' ranges from 1 to ' n ' where ' n ' is the total no. of members in ' H_u '.
- (c) *Band selector:* Depending on the application, in other words depending on which feature is required to be extracted from the image most efficiently, the band for partitioning can be chosen [Goel, L., et. al., 2010C]. The NIR and MIR bands of the 7-band image have been chosen since these bands contain a good amount of geo-spatial information and these are the bands in which the features are particularly more highlighted and best viewed [Goel, L., et. al., 2010C].
- (d) *Discretizor and Partitioner:* Therefore, the NIR and the MIR bands are used for discretization and partitioning step in the semi – naïve algorithm used for creating rough set equivalence classes for each of the clusters. Each of these resultant classes are put in the universal habitat.

Surface Entropy / DMM Representation Layer: In the extended BBO/EE for land cover feature extraction, it is considered that the HSI function can vary dynamically during optimization [Goel, L., et. al., 2013B]. Making use of the proposed hypotheses, the HSI of each of the habitat H_i in the universal habitat H_u is calculated using two different functions namely entropy and 'deviation from mean metric' and hence the HSI function definition is defined as

```

/* Ecosystem initialization */
Create a Universal habitat consisting of the rough set generated equivalence classes also
called as the species (population)  $H_1, H_2, \dots, H_n$  which comprises  $H_u$ ;
Create a Feature habitat consisting of the expert generated training set for each of the
land cover features to be extracted  $H_f$ ;
/*Compute the corresponding HSI values */
while ( $i \leq n$ )
    Compute the HSI for each species  $H_i$  in  $H_u$  based on the corresponding function
     $f_i(HSI)$  where  $f_i(HSI) = \{\text{Entropy, Deviation from mean metric}\}$ ;
    Also, calculate the HSI for each of the feature habitat  $H_j$ .
end while
/* End of parameter initialization */
While not  $T(H_u \neq \text{NULL})$  do /* T is the termination criterion */
for each species  $H_i$  in the Universal habitat  $H_u$ 
    Select a species from  $H_u$ 
    for each of the feature habitat  $H_j$ .
        Migrate the specie  $H_i$  to  $H_j$ .
        Recalculate the HSI of the feature habitat  $H_j$  after the migration of the species  $H_i$  to
        it based on entropy and deviation from mean metric separately.
    End For
    Calculate the percentage minimum difference obtained between  $H_i$  and  $H_j$  for both
    metrics.
    If % Differenceentropy < % DifferenceDMM
         $f_i(HSI) = \{\text{Entropy}\}$ ; FeatureIndex( $f_i(HSI)$ ) = FeatureIndex(Entropy);
    Else
         $F_i(HSI) = \{\text{Deviation from mean metric}\}$ ;
        FeatureIndex( $f_i(HSI)$ ) = FeatureIndex(Deviation from mean metric);
    End
    Absorb the species  $H_i$  in the feature habitat  $H_j$ .
End For
End While.

```

Figure 5.23: Algorithm for land cover feature extraction based on the extended BBO/EE

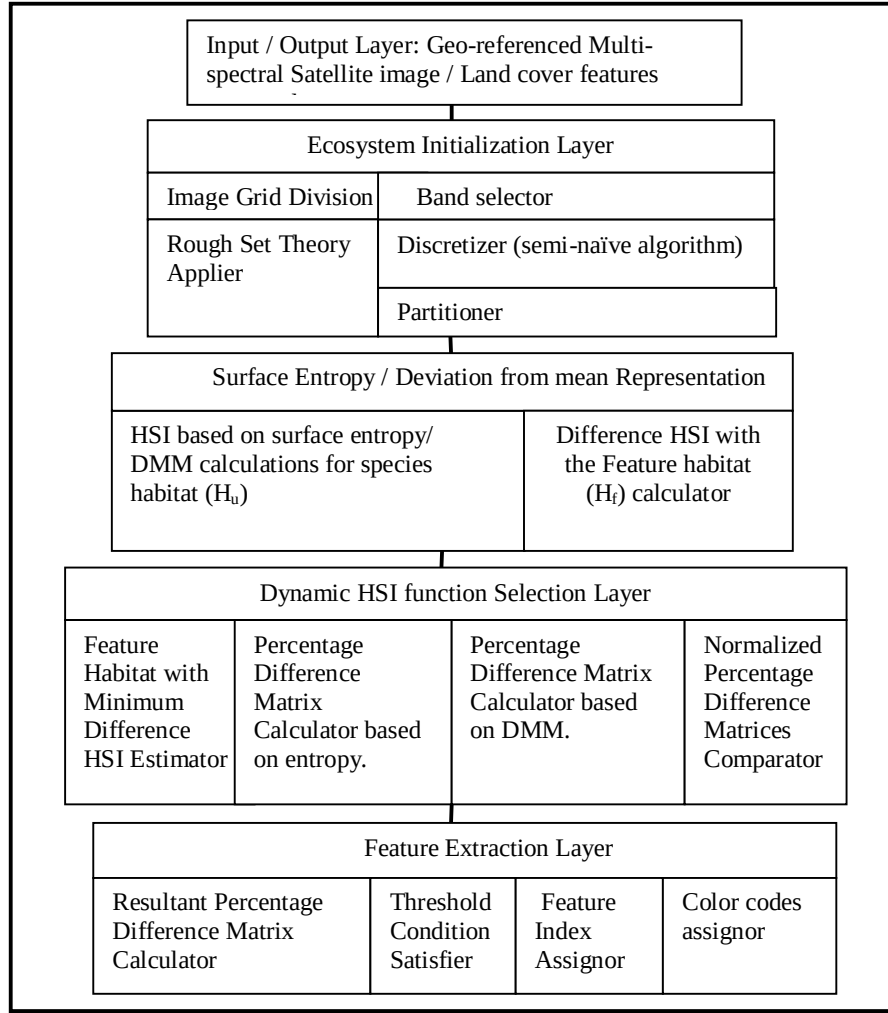


Figure 5.24: Software architecture of the proposed land cover feature extractor based on extended BBO/EE

{Entropy, Deviation from mean metric} such that the feature index decided by the function chosen achieves a smaller percentage difference value. The sub-layers are described below:

- (a) *HSI based on surface entropy / DMM calculations for species habitat*: The HSI calculation procedure using deviation from mean metric and entropy functions for the species habitat is described next [(Goel, L., et. al., 2011A), (Goel, L., et. al., 2011C), (Goel, L., et. al., 2013C)]. The HSI calculations based on deviation from mean metric or entropy of the species habitat C_i (or rough sets generated equivalence classes) from H_i of the universal habitat Π are represented by:

$$\begin{aligned}
 & HSI_{((DMM \text{ or } Entropy)_{C_i})} \\
 &= \frac{(|\omega_{1_i}| + |\omega_{2_i}| + |\omega_{3_i}| + |\omega_{4_i}| + \dots + |\omega_{m_i}|)}{m}
 \end{aligned} \tag{5.14}$$

where, $|\omega_{1_i}|, |\omega_{2_i}|, |\omega_{3_i}|, |\omega_{4_i}|, \dots, |\omega_{m_i}|$ are the absolute values of the deviation from mean metrics or entropy of the DN values of each of the m bands respectively of the i^{th} species habitat in the universal habitat H_u . The HSI calculations based on deviation from mean /

entropy for each habitat H_j in H_f can be performed similarly. The HSI calculations based on deviation from mean metric or entropy of any of the feature islands F_j are given by:

$$HSI_{((DMM \text{ or } Entropy)_{F_j})} = \frac{(|\omega_{1_j}| + |\omega_{2_j}| + |\omega_{3_j}| + |\omega_{4_j}| + \dots + |\omega_{m_j}|)}{m} \quad (5.15)$$

$$HSI_{((DMM \text{ or } Entropy)_{H_i})} - HSI_{((DMM \text{ or } Entropy)_{H_f})} = |HSI_{c_i} - HSI_{F_j}| \quad (5.16)$$

- (b) *Difference HSI with the Feature habitat (H_f) calculator:* The deviation from mean metric and entropy calculations for each habitat H_j in H_f can be performed similarly as explained before. The difference in HSI of the species habitat (ω_{c_i}) with the feature habitat (ω_{F_j}) is given by the equation stated above.

Dynamic HSI function selection Layer: This layer is responsible for the selection of the suitable HSI function dynamically with each iteration. The HSI function which decides the classified feature for each species habitat H_i changes (evolves or becomes extinct) based on the classification efficiency that is provided by it. The detailed procedure for the dynamic HSI function modeler layer is described below.

- (a) *Feature Habitat with Minimum Difference HSI Estimator:* After calculating the difference between the HSI of the selected species H_i and each of the feature habitats H_j based on entropy and deviation from mean metric separately, this layer is responsible for finding out the feature habitat with which H_i has the minimum difference i.e. H_j for which $|HSI_{c_i} - HSI_{F_j}|$ is minimum for both entropy and deviation from mean metric.
- (b) *Percentage Difference Matrix Calculator based on entropy and DMM:* This layer is responsible for calculating the percentage of this minimum difference value with the HSI of the species habitat H_i based on entropy and deviation from mean metric separately and is given by the following expression.

$$\frac{\min(HSI_{entropy \mid DMM}(H_i) - HSI_{entropy \mid DMM}(H_j))}{HSI_{entropy \mid DMM}(H_i)} \times 100 \quad (5.17)$$

- (c) *Normalized Percentage Difference Matrices Comparator:* This layer compares the percentage difference matrices based on entropy and deviation from mean metric. The output percentage difference values are dynamically decided based on the HSI function chosen for the equivalence class under consideration. For comparing the percentage difference matrices, the values are normalized so that both the HSI function based on entropy return the percentage difference values that are scaled in the interval of the percentage difference values returned by the HSI function based on deviation from mean

metric. Scaling factor for each of the table entries i.e. for the percentage difference values for each equivalence class used is defined as below:

$$\text{Scaling factor}_i = \frac{\text{Max}(\text{Percentage diff}(\text{DMM})) - \text{Min}(\text{Percentage diff}(\text{DMM}))}{\text{Max}(\text{Percentage diff}(\text{Entropy})) - \text{Min}(\text{Percentage diff}(\text{Entropy}))} \times \text{Percentage diff}_i(\text{Entropy}) \quad (5.18)$$

where, i represents the i^{th} equivalence class.

Feature Extractor Layer: This layer is responsible for feature extraction based on dynamic HSI function modeling.

- (a) *Resultant Percentage Difference Matrix Calculator:* This layer constructs the matrix of the output percentage difference values which will be mapped to the corresponding feature indices which will form the classified image. The function (entropy or deviation from mean metric) which yields smaller percentage difference value is chosen as the HSI function.
- (b) *Threshold Condition Satisfier:* This layer checks if the HSI of the species habitat H_i for which the percentage difference is minimum is within the threshold ‘ δ ’ in order for the input equivalence class to be classified into the corresponding feature index. If the HSI calculated is within the threshold, then the input class is classified into the corresponding feature and removed from ‘ H_u ’ otherwise, the input class which remained in ‘ H_u ’ (since it could not be classified) are further split through rough sets and the above procedure is repeated for the input equivalence class left in the universal habitat. Initialize a new ecosystem for the classification of the remaining classes in ‘ H_u ’.
- (c) *Feature Index Assigner:* This layer is responsible for assigning the feature index of the HSI function which has the smaller percentage difference value. These feature index values are the values into which the equivalence class is classified into finally based on the dynamically decided HSI function.
- (d) *Color Codes Assignment Layer:* Finally color codes are assigned for each pixel of the image corresponding to the classified feature of each pixel.

Output Layer: Final classified image is obtained in .jpeg, .tiff or any other image format.

5.2.2. Results and Discussion

Next the applications of the proposed algorithm for land cover feature extraction are discussed using the datasets of Alwar and Patalganga.

A. Case study: Alwar region in Rajasthan

The case study presents the results and also assesses the accuracy of the results of the proposed extended BBO/EE based feature extractor and compares them with the original biogeography based feature extractor on the Alwar image. The percentage difference for the HSI values based on entropy and deviation from mean metric in the multi-spectral bands are calculated for each

rough sets generated equivalence class with the equivalence class with which it has minimum difference with in each of the 'n' ($n = 20$) partitions of the image after the values have been normalized and scaled respectively. Hence the output percentage difference values are calculated which will be mapped to the corresponding feature indices to form the classified image based on the dynamically decided HSI function. Figure 5.25 presents the bar graph representing the percentage difference matrix based on HSI function based on 'entropy' and 'deviation from mean metric' for the Alwar Image. The color codes are red for entropy and blue for deviation from mean metric. From the graphs, it can be observed that there is a significant difference between the percentage difference matrix plot for entropy and that of deviation from mean metric for most of the equivalence classes and hence, the accuracy provided by these functions may vary significantly on these equivalence classes. This means that the classification accuracy of the HSI function based on entropy might be significantly less than the classification accuracy provided by the HSI function based on deviation from mean metric and vice versa. Hence, the dynamic functioning in BBO is guaranteed to improve the classification accuracy provided by either of the two HSI functions used independently which is what is reflected and also verified from the classification results of Alwar too. Table 5.22 presents the table of the output percentage difference values which will be mapped to the corresponding feature indices which will form the classified image. Table 5.23 represents the final feature index values for each of the equivalence classes based on the dynamically decided HSI function.

The feature index codes are 1-barren, 2-rocky, 3-urban, 4-vegetation and 5-waterbody. Table 5.24 carries the information about the HSI function that was chosen for each of the equivalence classes based on the criteria of maximization of classification accuracy at runtime. The HSI function codes are 1 is for entropy and 2 is for deviation from mean metric.

The classified image is obtained in figure 5.28 (kappa coefficient = 0.8271) which clearly shows the extraction of land cover features of water, urban, rocky, vegetation and barren. The yellow color represents rocky area, green color represents vegetation area, black color represents barren area and red color represents the urban area. The classified image through BBO with the HSI function based on 'deviation from mean metric' [Panchal, V. K., et. al., 2009] is shown in figure 5.26 which has a kappa coefficient of 0.6751 and that with the HSI function based on 'entropy' [Goel, L., et. al., 2011C] is shown in figure 5.27 which has a kappa coefficient of 0.7440 which reflects that the proposed algorithm performs better than the original BBO used for land cover feature extraction [Panchal, V. K., et. al., 2009].

From the figures 5.26, 5.27 and 5.28, it can also be seen that the encircled urban region in grey which was wrongly classified as mixture of barren with urban by the original BBO with HSI function based on 'deviation from mean metric' as well as the original BBO with the HSI based on 'entropy', has been classified correctly by the proposed classifier.

Table 5.22: Resultant percentage difference matrix based on the dynamically decided HSI function for the Alwar image.

Result % difference matrix (based on dynamic HSI function) for equivalence class (i) / Partition (z)	1	2	3	4	5	6	7	8	9
1	3.1387	0.3156	2.6645	2.3874	1.0133	0.8855	1.4097	0.3414	0.3197
2	2.5095	0.5967	3.1925	2.1780	1.2364	1.4634	1.3080	0.7829	0.6457
3	3.3650	1.0817	2.4894	2.2944	2.1934	1.1082	0.0010	0.4121	0.5788
4	3.4060	2.7764	1.4383	2.7176	2.4378	3.0953	1.4464	0.8451	1.1230
5	2.9662	0.6659	1.2697	0.4937	3.1466	2.3419	1.7159	1.0119	0.8046
6	2.7015	1.7846	0.7089	1.7502	2.5295	0.4444	0.0433	2.4822	2.2536
7	0.9142	0.4377	0.9412	0.2463	1.5018	3.2462	0.0780	3.1369	1.0562
8	0.3304	0.9013	3.0350	1.4279	1.9940	3.0107	0.0231	0.7091	1.8127
9	0.4167	0.7449	4.0645	0.8191	1.1809	2.0885	1.3138	1.1681	1.0605
10	2.8680	1.4106	1.2324	1.0254	2.4472	2.1280	1.3375	1.0379	1.1226
11	1.2879	0.3028	2.9969	1.9558	2.8283	2.8712	3.5600	2.0904	0.0000
12	0.0480	0.8665	0.0384	1.4400	2.3192	0.3045	0.0149	3.4109	0.9663
13	4.7319	1.7193	4.0712	2.6879	2.1738	2.7199	0.2169	2.6836	1.0931
14	0.3967	1.4640	3.9184	0.5054	1.2255	1.0397	0.4773	0.6835	1.0674
15	2.7057	2.5381	1.7159	1.3473	0.1594	0.1044	0.7475	1.2777	1.8372
16	1.8155	1.4522	1.5695	1.0300	3.2664	3.8690	3.1855	2.1780	0.0000
17	1.5321	0.2648	2.5766	2.3653	2.3758	3.1161	0.0019	0.2038	3.3854
18	3.6384	3.4229	0.4210	3.0027	2.7517	2.8416	3.8026	1.0664	0.9535
19	1.9808	0.4728	1.1273	3.5561	2.3165	1.5299	1.3880	3.1042	1.6254
20	4.2054	3.6353	1.4102	0.4450	2.0559	0.1914	0.5930	2.3053	1.2168

The encircled barren region in pink which was wrongly classified as urban by the original BBO with HSI function based on ‘deviation from mean metric’ and as mixture of urban with barren by the original BBO with the HSI based on ‘entropy’, has been classified correctly by the proposed classifier. Also, the encircled urban region in blue which was wrongly classified by the original BBO (with HSI function based on ‘Deviation from mean metric’) as barren has been classified correctly by original BBO with ‘entropy’ as the basis of the HSI function as well as by the proposed extended BBO/EE based classifier.

Tables 5.25 and 5.26 present the error matrices for the original BBO classifier with the HSI function based on ‘deviation from mean metric’ and the HSI function based on ‘entropy’ respectively [(Goel, L., et. al., 2011A), (Goel, L., et. al., 2011C), (Goel, L., et. al., 2013C), (Panchal, V. K., et. al., 2009)]. Table 5.27 presents the error matrix for the proposed biogeography based classifier. Table 5.28 represents the producer’s accuracy of classification from which it is reflected that water pixels have been extracted most efficiently followed by barren and vegetation pixels, respectively. Table 5.29 represents the user’s accuracy of classification from which it is observed that water shows 100% accuracy of classification

Percentage Difference matrix based on the HSI function 'Entropy' and 'Deviation from mean metric'

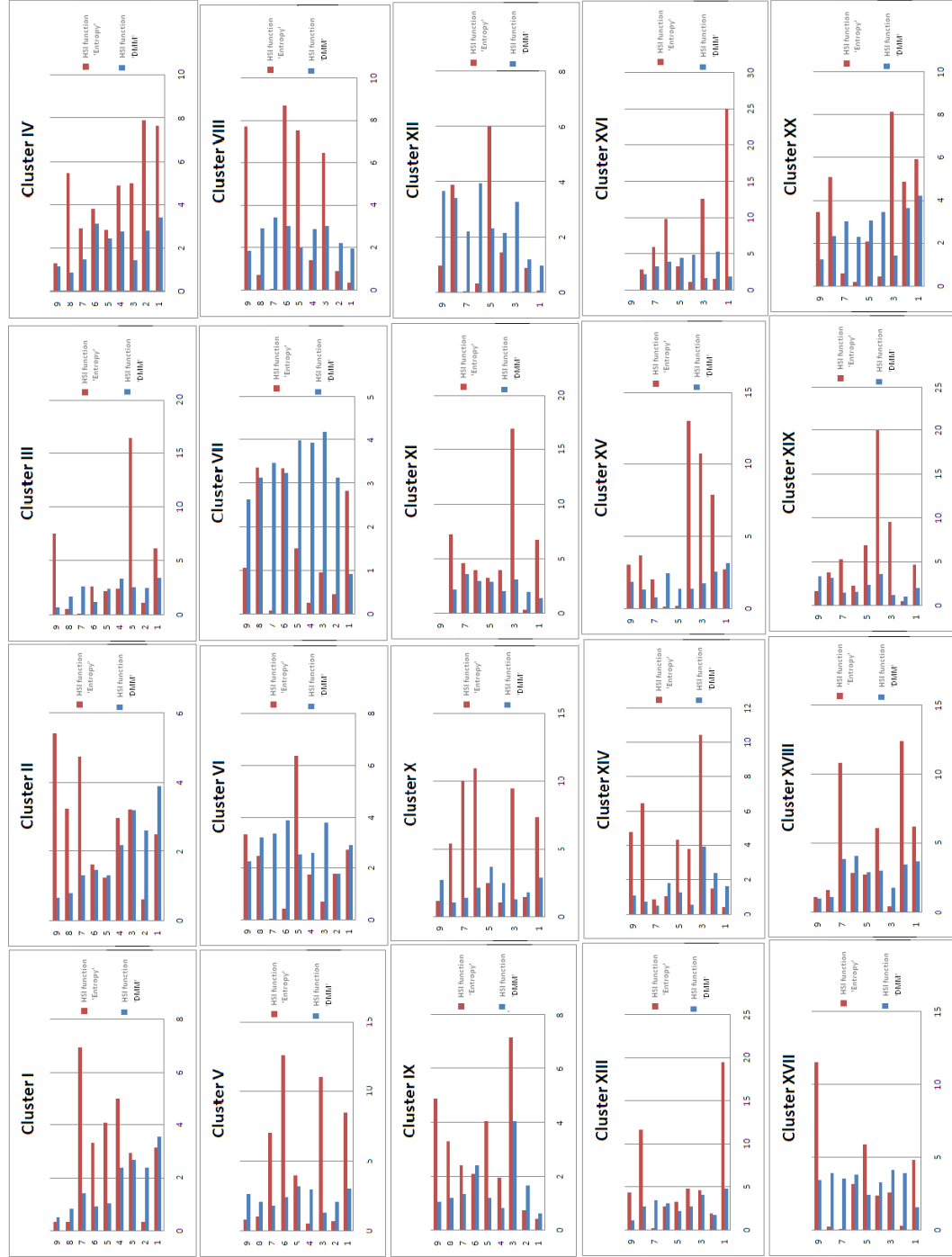


Figure 5.25: Bar graph representing the percentage Difference Matrix based on HSI function based on 'Entropy' and 'Deviation from mean metric' for the Alwar Image. Color codes are: Red- Entropy and Blue-Deviation from mean metric.

Table 5.23: Classified feature index matrix based on dynamically decided HSI function for the Alwar image. The feature index codes are 1-Barren, 2-Rocky, 3-Urban, 4-Vegetation and 5-Waterbody.

Classified Feature Index based on dynamic HSI function for equivalence class (i) / Partition (z)	1	2	3	4	5	6	7	8	9
1	2	1	1	1	1	1	1	1	1
2	2	1	4	1	3	1	1	1	1
3	2	2	4	2	1	1	1	1	1
4	4	4	4	3	3	4	1	1	1
5	4	4	4	1	4	4	3	3	1
6	2	2	2	2	2	2	2	2	1
7	2	2	2	2	2	3	2	3	1
8	2	2	4	2	2	3	2	2	3
9	1	1	4	3	3	3	1	1	1
10	4	4	4	3	1	4	1	1	1
11	2	2	4	2	2	1	2	1	0
12	2	2	2	2	2	2	2	2	2
13	2	3	3	2	3	3	2	3	1
14	1	1	4	3	3	1	3	3	3
15	1	4	4	1	3	1	1	1	1
16	5	2	4	2	2	3	3	3	0
17	2	2	2	2	2	2	2	2	3
18	2	4	4	3	1	1	3	1	1
19	4	4	4	4	4	4	3	3	1
20	3	3	4	3	3	1	1	3	1

Table 5.24: HSI functions chosen dynamically for each equivalence class in the Alwar image. The HSI function codes are 1-Entropy and 2-Deviation from mean metric.

Dynamic HSI function for equivalence class (i) / Partition (z)	1	2	3	4	5	6	7	8	9
1	2	2	1	1	1	1	1	2	2
2	2	2	1	1	2	1	1	1	1
3	1	2	1	2	2	1	2	2	1
4	1	1	1	1	1	1	1	1	1
5	1	2	1	2	1	1	1	2	2
6	2	2	2	2	1	2	2	2	1
7	1	2	2	2	2	1	2	1	2
8	2	2	1	2	1	1	2	2	1
9	2	2	1	1	1	2	1	1	1
10	1	2	1	2	2	1	1	1	2
11	1	2	1	1	1	1	1	1	2
12	2	2	2	2	1	2	2	1	2
13	1	1	1	1	1	2	2	1	1
14	2	2	1	1	1	2	1	1	1
15	2	1	1	1	2	2	1	1	1
16	1	2	1	2	2	1	1	1	2
17	1	2	2	2	1	2	2	2	1
18	1	1	2	1	2	2	1	1	1
19	1	2	1	1	1	1	1	1	2
20	1	1	1	2	2	2	2	1	1

followed by vegetation pixels. From the error matrix, the KHAT statistics of the extended BBO/EE based land cover feature extractor is calculated as 0.8271.

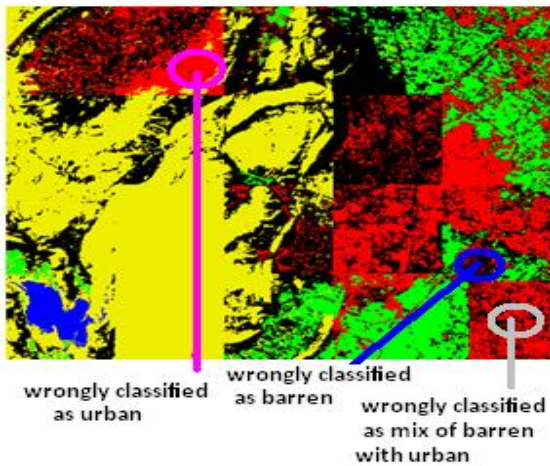


Figure 5.26: Final classified image of Alwar after applying BBO for land cover feature extraction with the HSI function based on 'deviation from mean metric'. (Kappa coefficient = 0.6751)

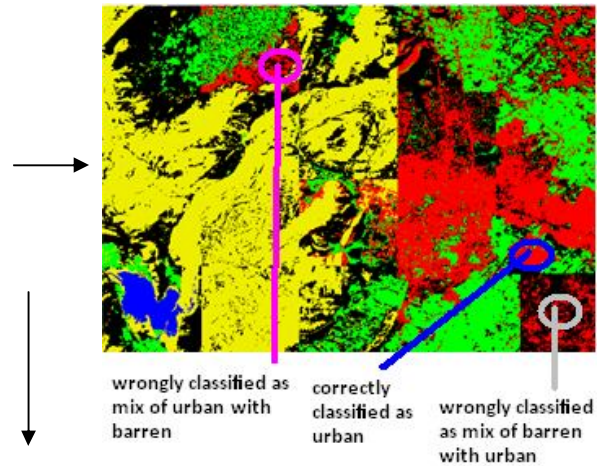


Figure 5.27: Final classified image of Alwar after applying original BBO for land cover feature extraction with the HSI function based on 'entropy'. (Kappa coefficient = 0.7440)

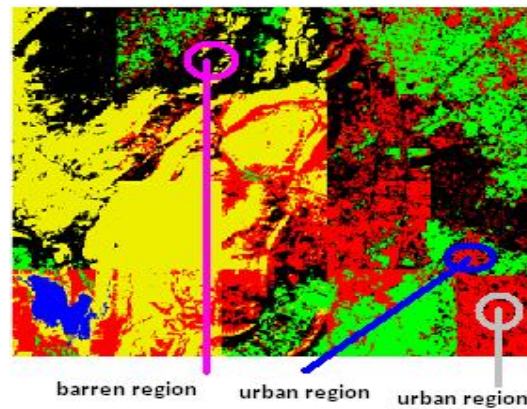


Figure 5.28: Final classified image of Alwar after applying extended BBO/EE based land cover feature extraction (Kappa coefficient = 0.8271)

Table 5.25: Error matrix of BBO with the HSI function based on 'deviation from mean metric' on Alwar region. (Kappa coefficient = 0.6715)

	Vegetation	Urban	Rocky	Water	Barren	Total
Vegetation	127	9	0	0	2	138
Urban	0	88	1	0	32	121
Rocky	6	2	176	1	17	202
Water	0	0	3	69	0	72
Barren	17	91	20	0	119	247
Total	150	190	200	70	170	780

Table 5.26: Error matrix of original BBO with the HSI function based on 'entropy' on Alwar region. (Kappa coefficient = 0.7440)

	Veg	Urban	Rocky	Water	Barren	Total
Vegetation	142	0	0	0	1	143
Urban	3	66	8	0	4	81
Rocky	0	14	185	1	4	204
Water	0	0	0	69	0	69
Barren	5	111	6	0	161	283
Total	150	190	200	70	170	780

Table 5.27: Error matrix of the extended BBO/EE based feature extractor on Alwar region.
(Kappa coefficient = 0.8271)

	Vegetation	Urban	Rocky	Water	Barren	Total
Vegetation	145	2	0	0	37	184
Urban	0	151	7	0	6	164
Rocky	0	10	182	1	2	194
Water	0	0	0	69	0	70
Barren	4	27	10	0	121	162
Total	149	190	199	70	166	774

Table 5.28: Producer's Accuracy

Feature	Accuracy Calculation	Producer's Accuracy
Vegetation	145/149	97.3%
Urban	151/190	79.5%
Rocky	182/199	91.5%
Water	69/70	98.7%
Barren	121/166	72.9%

Table 5.29: User's Accuracy

Feature	Accuracy Calculation	User's Accuracy
Vegetation	145/184	78.8%
Urban	151/164	92.1%
Rocky	182/194	93.8%
Water	69/70	98.7%
Barren	121/162	74.7%

B. Case study: Patalganga region in Shivalik ranges

This section demonstrates the proposed concept in the extended BBO/EE through the case study of Patalganga region in Shivalik ranges. The case study also compares the results with the original biogeography based feature extractor [(Goel, L., et. al., 2011B), (Panchal, V. K., et. al., 2009)] on the Patalganga image. The image is subjected to 'n' (here, $n = 15$) simple partitions, for the sake of simplicity. Table 5.30 presents the table of the output percentage difference values which will be mapped to the corresponding feature indices to form the classified image [(Goel, L., et. al., 2011A), (Goel, L., et. al., 2011C), (Goel, L., et. al., 2013C), (Panchal, V. K., et. al., 2009)]. Table 5.31 represents the final feature index values for each of the equivalence classes. The feature index codes are 1-rocky, 2-snow and 3-vegetation. Table 5.32 carries the information about the HSI function that was chosen for each of the equivalence classes based on the criteria of maximization of classification accuracy at runtime. The HSI function codes are 1 is for entropy and 2 is for deviation from mean metric. Figure 5.29 presents the bar graph representing the percentage difference matrix based on HSI function based on 'entropy' and 'deviation from mean metric' for the Patalganga image. The color codes are red for entropy and blue for deviation from mean metric.

The classified image is obtained in figure 5.32 (Kappa coefficient = 0.8014) which clearly shows the extraction of land cover features of snow, vegetation and rocky area. The yellow color represents rocky area, green color represents vegetation area, and red color represents the snowy area. The classified image through original BBO with the HSI function based on

‘deviation from mean metric’ [Panchal, V. K., et. al., 2009] is shown in figure 5.30 which has a kappa coefficient of 0.7248 and that with the HSI function based on ‘entropy’ [(Goel, L., et. al., 2011A), (Goel, L., et. al., 2011C), (Goel, L., et. al., 2013C)] is shown in figure 5.31 which has a kappa coefficient of 0.7870 which reflects that the proposed algorithm performs better than the original BBO used for land cover feature extraction [Panchal, V. K., et. al., 2009]. From the figures 5.30, 5.31 and 5.32, it can also be seen that the encircled snowy region in black which was wrongly classified as rocky by the original BBO with HSI function based on ‘deviation from mean metric’ and as the mixture of rocky with snowy by the original BBO with the HSI based on entropy, has been classified correctly by the proposed classifier. Also, the encircled vegetation region in green which was wrongly classified as mixture of rocky with vegetation by the original BBO with HSI function based on ‘deviation from mean metric’ and as rocky region by the original BBO with the HSI based on entropy, has been classified correctly by the proposed classifier. However, the encircled rocky region in blue which was wrongly classified by the original BBO (with HSI function based on ‘Deviation from mean metric’) as snowy has been classified correctly by original BBO with entropy as the basis of the HSI function as well as by the proposed classifier.

Table 5.30: Resultant percentage difference matrix based on the dynamic HSI function for the Patalganga image.

Result % difference matrix (based on dynamic HSI function) for equivalence class (i) / Partition (z)	1	2	3	4	5	6	7	8	9	10
1	31.4595	32.6077	36.5877	2.6325	3.6958	13.1228	24.6765	17.4335	37.5536	46.3884
2	37.2776	46.7695	14.0495	37.9567	1.4395	46.1176	18.3926	9.5926	10.4002	46.8037
3	42.8281	54.0777	41.7646	1.2983	1.6602	39.7041	48.7101	58.2453	8.1338	18.5350
4	5.7710	49.0691	38.2160	2.9689	22.6297	19.6380	34.2123	50.8345	7.3310	55.3811
5	0.9113	19.5651	35.4748	41.1007	7.0122	45.2776	11.6049	53.9228	4.1491	46.6975
6	25.1551	30.6992	20.6391	33.9123	35.5987	36.8342	56.0569	38.8241	10.7753	46.9550
7	51.4970	33.1910	16.2469	46.1758	32.7724	29.1094	32.1576	0.8653	36.1015	49.0937
8	4.6016	42.5368	39.3962	4.3798	0.3924	32.6203	27.8994	3.9418	16.9248	3.9046
9	17.3630	26.9176	47.0444	18.6882	18.6694	35.8005	4.0049	22.6204	49.2508	30.5126
10	43.8309	2.0041	37.3043	8.5308	45.3732	42.2212	3.0492	25.4183	6.4234	45.0970
11	41.7998	20.9356	17.4706	17.3992	24.0679	32.6477	42.4163	14.5413	6.8747	1.4292
12	22.2286	35.1091	12.8016	9.3076	0.2770	9.6848	40.1519	35.5811	54.4791	37.7329
13	45.3393	18.7474	44.7532	38.8009	20.0577	18.6259	59.1142	47.8397	51.7726	45.5289
14	41.7281	2.5363	13.5216	36.5922	15.6229	33.5431	21.8654	35.8845	29.6682	32.8326
15	42.8897	43.5736	4.1505	45.3139	11.4804	13.5281	48.6938	22.5328	46.0053	46.1073

Percentage Difference matrix based on the HSI function 'Entropy' and 'Deviation from mean metric'

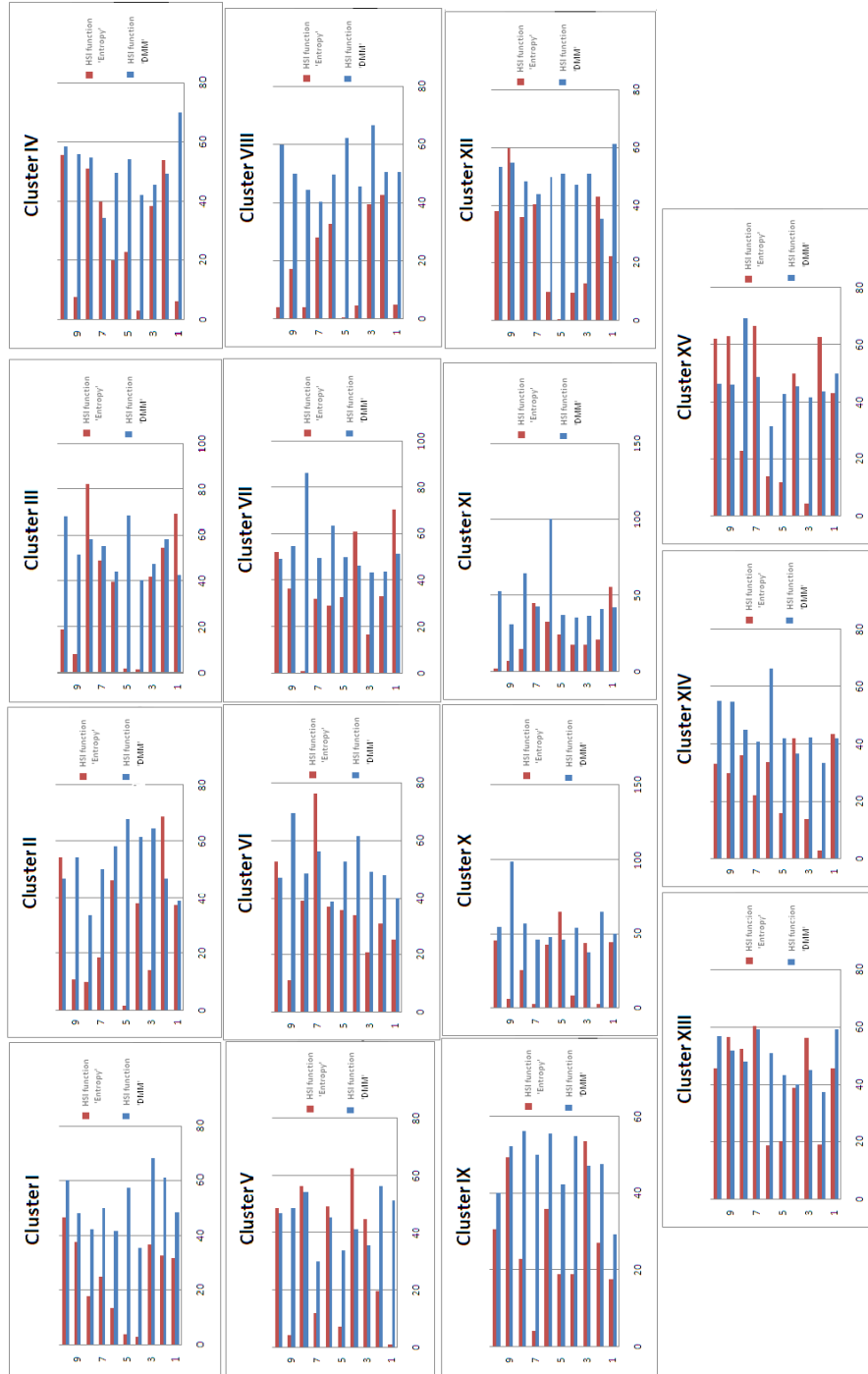


Figure 5.29: Bar graph representing the percentage difference matrix based on HSI function based on 'entropy' and 'deviation from mean metric' for the Patalganga Image. Color codes are: red- entropy and blue- deviation from mean metric.

Table 5.31: Classified feature index matrix based on dynamically decided HSI function for the Patalganga image. The feature index codes are 1-rocky, 2-snow and 3-vegetation.

Classified Feature Index based on dynamic HSI function for equivalence class (i) / Partition (z)	1	2	3	4	5	6	7	8	9	10
1	2	2	2	2	2	1	1	1	2	2
2	2	2	2	2	2	2	2	1	1	2
3	2	2	2	2	2	2	2	2	1	2
4	2	2	1	2	2	1	3	2	2	2
5	3	2	3	2	1	2	1	2	2	2
6	2	1	2	2	2	1	2	1	2	2
7	2	2	2	2	2	2	2	3	1	2
8	2	2	2	2	3	2	2	2	2	2
9	1	2	3	2	2	2	2	1	2	1
10	2	2	3	2	2	2	2	2	3	2
11	2	2	2	1	1	1	3	2	1	2
12	1	3	2	2	2	2	2	2	2	2
13	2	2	2	1	2	2	2	2	2	2
14	3	2	2	3	2	2	2	2	2	2
15	2	2	2	2	2	1	2	2	2	2

Table 5.32: HSI functions chosen dynamically for each equivalence class in the Patalganga image. The HSI function codes are 1-entropy and 2-deviation from mean metric.

Dynamic HSI function for equivalence class (i) / Partition (z)	1	2	3	4	5	6	7	8	9	10
1	2	2	2	2	2	2	2	2	2	2
2	2	1	2	2	2	2	2	2	2	1
3	1	2	2	2	2	2	2	1	2	2
4	2	1	2	2	2	2	1	2	2	2
5	2	2	1	1	2	1	2	1	2	1
6	2	2	2	2	2	2	1	2	2	1
7	1	2	2	1	2	2	2	2	2	1
8	2	2	2	2	2	2	2	2	2	2
9	2	2	1	2	2	2	2	2	2	2
10	2	2	1	2	1	2	2	2	2	2
11	1	2	2	2	2	2	1	2	2	2
12	2	1	2	2	2	2	2	2	1	2
13	2	2	1	2	2	2	1	1	1	2
14	1	2	2	1	2	2	2	2	2	2
15	2	1	2	1	2	2	1	2	1	1

Tables 5.33 and 5.34 present the error matrices for the original BBO classifier with the HSI function based on ‘deviation from mean metric’ and on ‘entropy’ respectively. Table 5.35 presents the error matrix for the proposed classifier. Table 5.36 represents the producer’s accuracy of classification from which it is reflected that vegetation pixels have been extracted perfectly followed by rocky pixels. Table 5.37 represents the user’s accuracy of classification. From the error matrix, the KHAT statistics of the proposed land cover feature extractor is calculated as 0.8014.

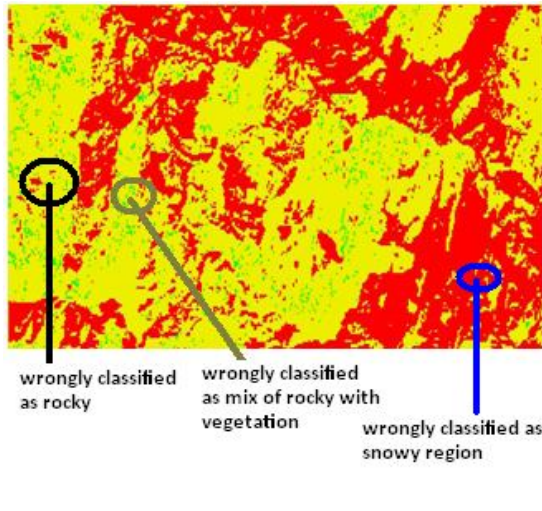


Figure 5.30: Final classified image of Patalganga region after applying BBO with the HSI function based on 'deviation from mean metric' for land cover feature extraction (Kappa coefficient=0.7248)

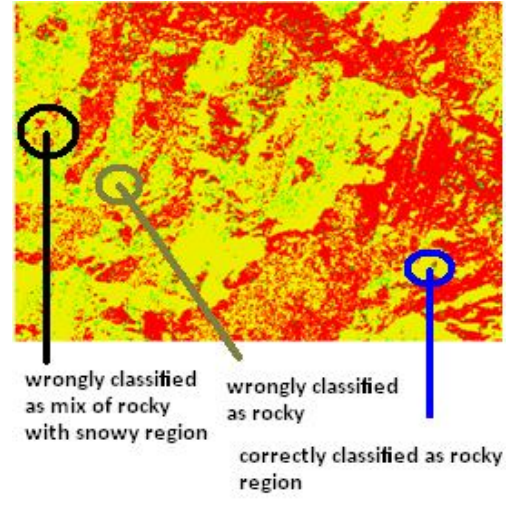


Figure 5.31: Final classified image of Patalganga region after applying BBO with the HSI function based on 'entropy' for land cover feature extraction (Kappa coefficient=0.7870)

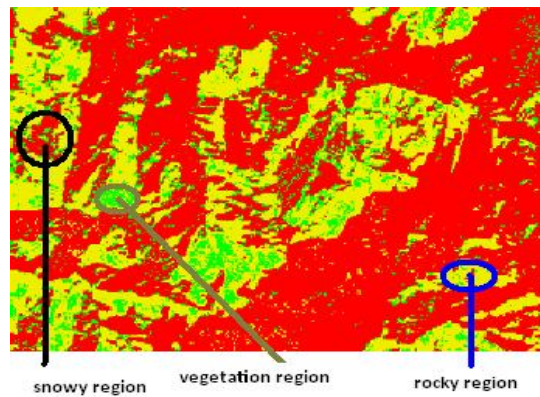


Figure 5.32: Final classified image of Patalganga region after applying extended BBO/EE based land cover feature extraction (Kappa coefficient=0.8014)

C. Classification comparison of BBO/EE based hybrid bio-inspired intelligent classifier with the other recent classifiers

In this section, the results of classification of the extended BBO/EE based hybrid classifier are presented since the hybrid ACO2/PSO/BBO classifier is the best known classifier developed till date [Goel, L., et. al., 2012A] and hence the result of substituting the original BBO with the proposed extended BBO/EE in the design of this hybrid bio-inspired intelligent classifier are studied. The results are compared with all the recent CI based classifiers developed, on the 7-band resourcesat satellite image of Alwar region in Rajasthan, India [Lillesand, T. M., et. al., 2007].

Table 5.33: Error matrix of BBO with the HSI function based on ‘deviation from mean metric’ on Patalganga region. (Kappa coefficient = 0.7248).

	Vegetation	Snow	Rocky	Total
Vegetation	89	7	7	103
Snow	48	181	4	224
Rocky	24	5	189	227
Total	161	200	200	561

Table 5.34: Error matrix of BBO with the HSI function based on ‘entropy’ on Patalganga region. (Kappa coefficient = 0.7870)

	Rocky	Snow	Vegetation	Total
Rocky	197,630	22,572	0	220,202
Snow	18,330	121,008	0	139,338
Vegetation	0	0	16,412	16,412
Total	215,960	143,580	16,412	375,952

Table 5.35: Error matrix of extended BBO/EE based feature extractor on Patalganga region. (Kappa coefficient = 0.8014)

	Rocky	Snow	Vegetation	Total
Rocky	197,730	22,472	0	220,202
Snow	18,130	121,208	0	139,338
Vegetation	0	0	16,412	16,412
Total	215,860	143,680	16,412	375,952

Table 5.36: Producer’s Accuracy

Feature	Accuracy	Producer’s
Rocky	197,730/215,860	91.6%
Snow	121,208/143,680	84.3%
Vegetation	16,412/16,412	100%

Table 5.37: User’s Accuracy

Feature	Accuracy Calculation	User’s Accuracy
Rocky	197,730/220,202	89.8%
Snow	121,208/139,338	86.9%
Vegetation	16,412/16,412	100%

From the discussion above, it is evident that the concept of dynamic HSI function in biogeography based optimization for the development of the proposed hybrid classifier produces better results than the other recent CI based as well as the traditional probabilistic classifiers such as the Minimum Distance to Mean Classification (MDMC) and the Maximum Likelihood Classifier (MLC) [(Ying, L., et. al., 2009), (Long, W. III, et. al., 2004)]. The satellite image classification results of the other recent CI based classifiers when applied on the Alwar region are shown below in Figure 5.33. Figures 5.33 (a) and (b) show the traditional MDMC and MLC classification of Alwar region which achieve a classification accuracy of 0.7364 and 0.7525 only [(Ying, L., et. al., 2009), (Long, W. III, et. al., 2004)]. Figures 5.33 (c) and (d) show the result of applying the fuzzy and the integrated rough-fuzzy classification of Alwar region which achieve the kappa – coefficients of 0.9134 and 0.9700 respectively [Panchal, V. K., et. al., 2008]. Figure 5.33(e) applies the cAntMiner (ACO2) algorithm on the Alwar region which has a kappa coefficient of 0.964 [(Blum, C., 2005), (Dorigo, M., et. al.,

2004), (Parpinelli, R. S., et. al., 2002)]. Figures 5.33(f) and (g) show the result of applying the hybrid ACO-BBO technique and the hybrid ACO2/PSO technique on the Alwar image which achieve kappa-coefficients of 0.96699 and 0.975 respectively [(Bansal, S., et. al., 2009), (Holden, N., et. al., 2005), (Hand, J., 1997), (Omkar, S. N., et. al., 2007)]. Figure 5.33 (h) presents the results of the semantic web based classifier on the image with a kappa coefficient of 0.9881 [Kumar, S., et. al., 2009]. Figure 5.33(i) presents the results of the hybrid ACO2/PSO/BBO classifier on the image with a kappa coefficient of 0.9818 [(Goel, L., et. al., 2010C), (Goel, L., et. al., 2012A)]. Figure 5.33(j) presents the results of the biogeography based feature extractor with non-linear immigration and emigration rates on the image with a kappa coefficient of 0.6912 [Gupta, S., et. al., 2011]. Figures 5.33 (k) and (l) are the result of applying the membrane computing and the bacterial foraging algorithms (hybrid FPAB/BBO) and achieve accuracy of 0.6818 and 0.6790 respectively [(Gupta, D., et. al., 2011), (Johal, N. K., et. al., 2010)]. Figure 5.33(m) shows the classification results after applying the proposed hybrid ACO2/PSO/(BBO/EE) classifier which achieves the classification efficiency of 0.9916 and hence enhances the accuracy provided by the original hybrid ACO2/PSO/BBO classifier (which had a kappa coefficient of 0.9818). *In fact, the hybrid ACO2/PSO/(BBO/EE) based on the proposed concept of dynamic HSI function in BBO achieves the highest classification efficiency of 99.16% and hence outperforms all the recent CI based classifiers developed till date.*

Table 5.38 presents the error matrix of the proposed classifier. Tables 5.39 and 5.40 present the producer's and the user's accuracies of this classifier. Table 5.41 summarizes and compares the kappa coefficients of the proposed hybrid ACO2/PSO/(BBO/EE) classifier with the other recent classifiers.

Table 5.38: Error matrix of Hybrid ACO2/PSO/(BBO/EE) on Alwar region. (Kappa coefficient = 0.9917)

	Vegetation	Urban	Rocky	Water	Barren	Total
Vegetation	142	0	0	0	0	142
Urban	1	190	0	1	0	192
Rocky	0	0	198	0	2	200
Water	0	0	0	69	0	69
Barren	0	0	1	0	163	164
Total	143	190	199	70	165	767

Table 5.39: Producer's Accuracy

Feature	Accuracy Calculation	Producer's Accuracy
Vegetation	142/143	99.3%
Urban	190/190	100.0%
Rocky	198/199	99.5%
Water	69/70	98.6%
Barren	163/165	98.8%

Table 5.40: User's Accuracy

Feature	Accuracy Calculation	User's Accuracy
Vegetation	142/14	100.0
Urban	190/19	99.0%
Rocky	198/20	99.0%
Water	69/69	100%
Barren	163/16	99.4%

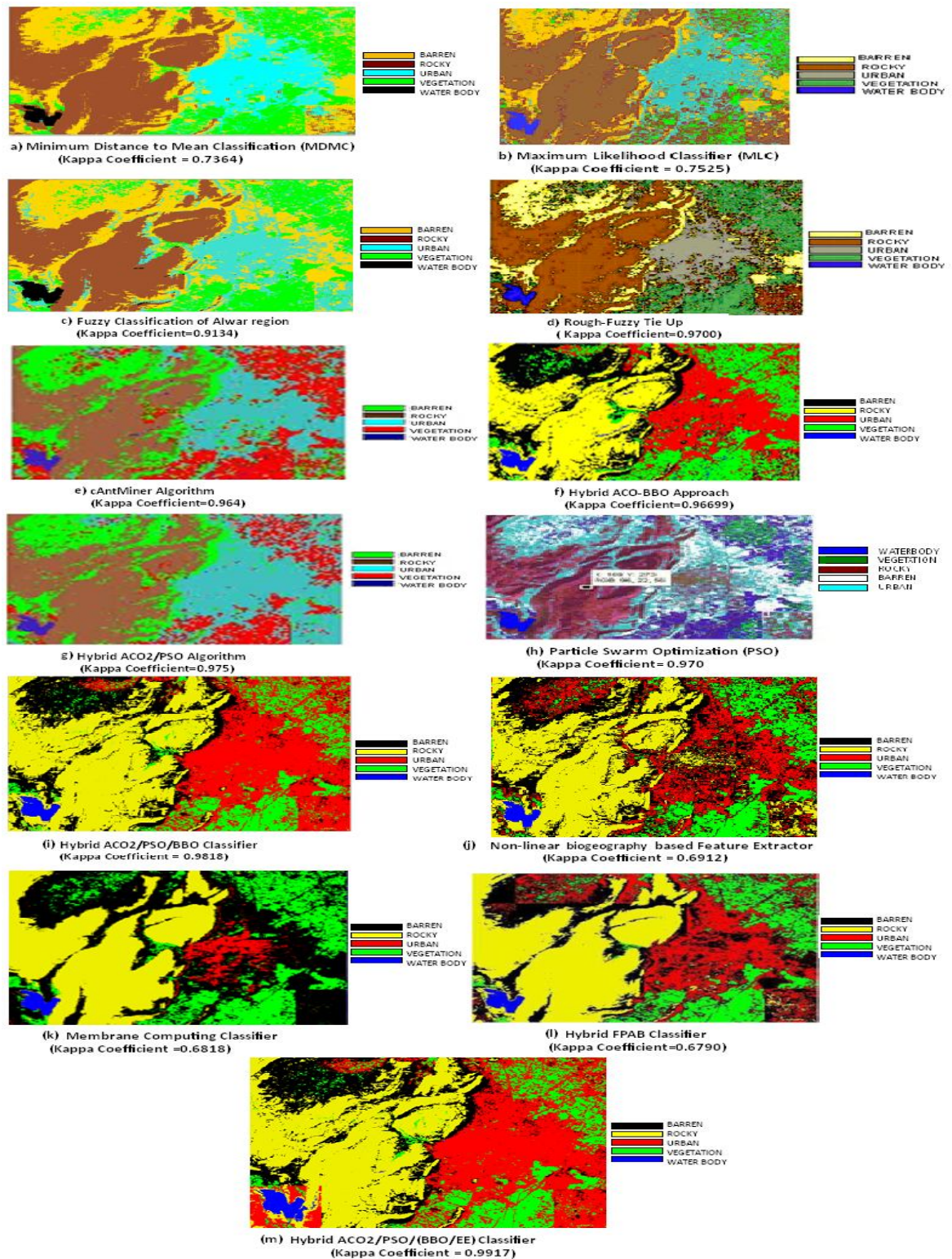


Figure 5.33: Classified images of Alwar region after applying various computing techniques

Table 5.41: Kappa coefficients of various recent CI based classifiers and probabilistic classifiers

MDMC	MLC	Fuzzy set	R-F Tie up	ACO	ACO/ PSO	Semantic Web Based	BBO	Hybrid ACO-BBO	Hybrid ACO/ PSO/ BBO	MC	Non-linear BBO	FPAB	Hybrid ACO/PSO/ (BBO/EE)
0.74	0.75	0.91	0.97	0.96	0.975	0.988	0.67	0.967	0.982	0.682	0.69	0.68	0.9917

5.3. Observations

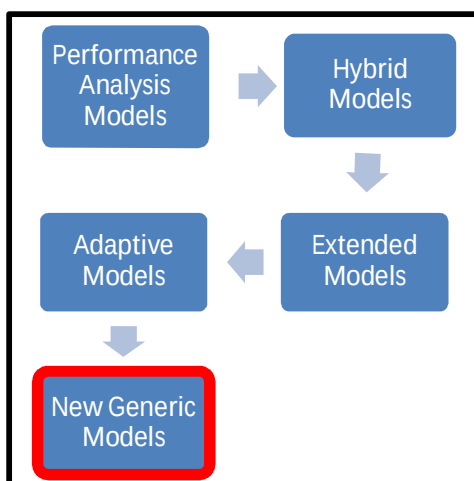
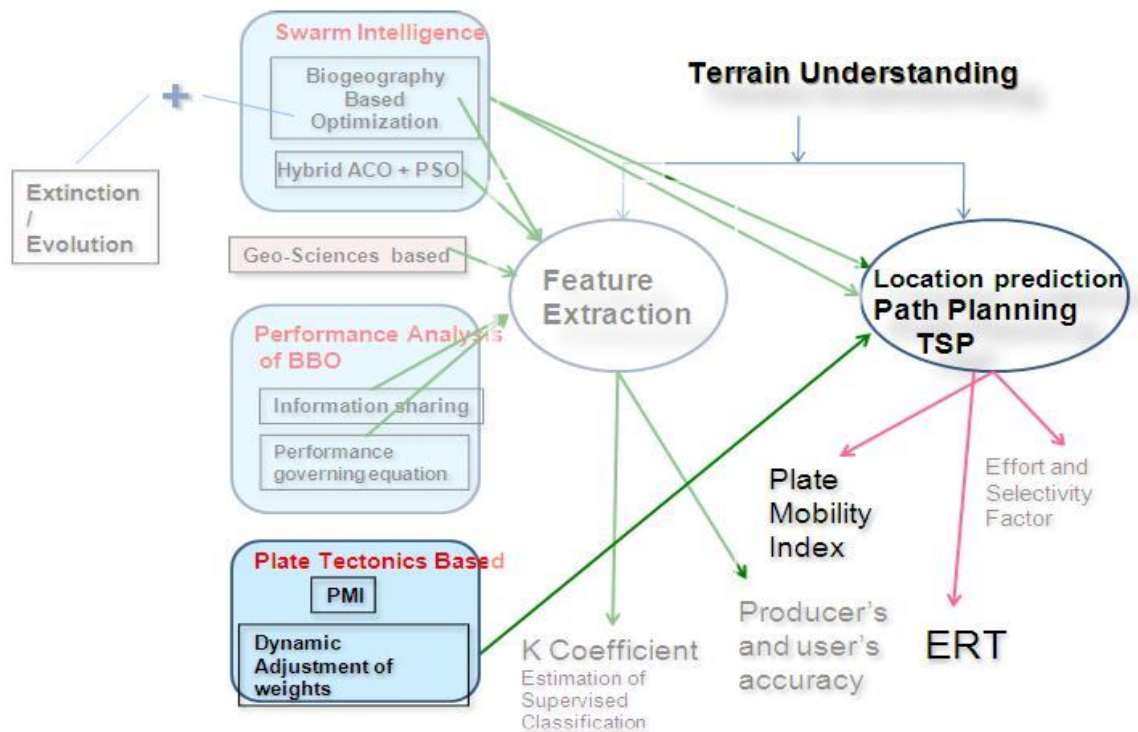
The chapter was an attempt to demonstrate the applicability of the proposed BBO/EE algorithm with HSI function modeling to practical engineering problems, such as in military, scientific and commercial applications. For the purpose, two major applications of location prediction and path planning; and land cover feature extraction were taken since these form the core of the terrain understanding applications targeted in the thesis. The BBO/EE algorithm was modified and factors of effort and selectivity were coined for the first time in the original BBO and hence the modified BBO/EE model was adapted for the military application of enemy base station prediction and path planning thereafter. The chapter also discussed the modification of the BBO/EE model to suit to the application of land cover feature extraction of satellite images by considering different definitions of HSI function for maximizing the classification efficiency each time.

Based on the results of the proposed two-phase recommender system, the commanders in the war theatre will have an increased battlefield awareness since they can now anticipate the destined enemy base station and can also select the shortest path to attack the enemy base station hence anticipating enemy's future actions. This will be of prime importance since in the modern warfare strategies, the ability of an army to position its troops at an effective geographical location (having predicted the location of the enemy base station), in the shortest possible time, will significantly gain a tactical advantage over the enemy and prove to be the deciding factor for winning. The proposed work can be further refined by modifying the existing SIVs and the addition of some more SIVs. Addition of new SIVs such as climate / weather conditions; state-liquid (water / petrol), solid (land / mountain), vapor (clouds), gaseous extracts (fog / smog); distance from roads (highways); water transportation modes; air strips supporting landing and flying of aircrafts (various sizes); railway network, land mine prone areas etc.; can help in building of a more versatile recommender system. Refinement of present SIVs by the inclusion of factors such as dimension (height / length / breadth), habitat (animals (friendly / dangerous), vegetation (edible / poisonous / allergic), surprise factor, etc. can further refine the recommender system to make it more efficient and robust. Couple of factors such as weapon capability and skills of the enemy are certain things which require human interpretation as these cannot be judged by the machine. Also the surprise factor contributes negatively to the comfort factor in other words it can be said that these are inversely proportional to each other. This would further help in determining additional information aiding in building a very refined recommender system that can be of great help in designing of the war / battlefield strategies. Also the above SIVs can be sampled round the year to generate very precise and accurate results.

From the results presented in the second section, it can be concluded that BBO/EE can be successfully applied for land cover feature extraction of satellite images. By taking the HSI function to be different and dynamic for each habitat, a lot of flexibility is brought in the choice of best solution. The performance of the extended model of biogeography based feature extraction technique has been demonstrated by running it on two different satellite images. The results indicate that the proposed optimizer is highly effective in extracting land cover features and when compared with the original BBO with the HSI function based on ‘deviation from mean metric’ and with the HSI function based on ‘entropy’ that was applied for land cover feature extraction [(Goel, L., et. al., 2011A), (Goel, L., et. al., 2013C), (Panchal, V. K., et. al., 2009)], it proves itself to be a much better classifier as shown in Table 5.38. A comparative study of the results of substituting the extended-BBO/EE classifier in the hybrid ACO2/PSO/BBO classifier with the other recent computational intelligence based classifiers is also presented wherein it is observed that the hybrid ACO2/PSO/(BBO/EE) classifier achieves the highest classification efficiency of 99.16% and hence outperforms the recent CI based classifiers developed till date. The future scope of the research includes proposing certain modification to the algorithm so that the kappa coefficient can be improved further. The current system is implemented using the simple heuristic techniques of deviation from mean metric and entropy; the system performance can be increased by using other heuristic functions for building a dynamic model.

The next chapter deals with exploring new natural phenomenon of plate tectonics, development of an analogy between the phenomenon and the optimization problem, deriving generic optimization model from the new methodology developed and hence adapting the generated optimization model for the terrain understanding application.

New Generic Models Of Nature Inspired Computational Intelligence



Chapter 6

- 6.1. Plate Tectonics
- 6.2. Plate Tectonics Based Optimization
- 6.3. Performance Analysis and Results
- 6.4. Observations

The chapter studies the geo-sciences based phenomenon of plate tectonics and demonstrates the application of the concepts and mathematics of plate tectonics as the basis for the development of a new optimization paradigm: **plate tectonics based optimization (PBO)**.

The theory of plate tectonics is a scientific theory that describes the large scale motions of Earth's lithosphere. The theory builds on the concepts of continental drift [(Iaffaldano, G., et. al., 2009), (Summeren, J. V., et. al., 2012)], developed during the first decades of the 20th century, when the concepts of seafloor spreading were developed in the late 1950s and early 1960s. According to Wegener [Wegener, A., 2011], the tectonics plate movement is thought to be driven by a combination of the motion of seafloor away from the spreading ridge (due to variations in topography and density of the crust that result in differences in gravitational forces) –ridge-push effect and drag, downward suction, at the subduction zones-slab-pull effect [(Summeren, J. V., et. al., 2012), (Matsuyama, T., et. al., 2010), (Richards, M., et. al, 2000), (Solomon, S. C., et. al., 1975)]. A different explanation lies in different forces generated by the rotation of the globe and tidal forces of the Sun and the Moon (external factors). One of the main points the theory proposes is that the amount of surface of the (continental and oceanic) plates that disappears in the mantle along the convergent boundaries by subduction is more or less in equilibrium with the new (oceanic) crust that is formed along the divergent margins by seafloor spreading. This is also referred to as the conveyor belt principle [(Summeren, J. V., et. al., 2012), (Richards, M., et. al, 2000), (Solomon, S. C., et. al., 1975)]. In this way, the total surface of the globe remains the same [(Solomon, S. C., et. al., 1975), (Thienen, P. V., et. al., 2004)].

Unlike the other evolutionary optimization algorithms like BBO [(Dawei, D.,et. al., 2009), (Simon, D., 2008), (Simon, D., 2011)], ACO [(Blum, C., 2005), (Dorigo, M., et. al., 2004)], PSO [(Clerc, M., 2006), (Eberhart, R., et. al., 2004)], DE [(Price, K., et. al, 1997), (Storn, R., 1999)], ES [(Kim, J. H., et. al.,1997), (Parmee, I., 2001), (Yao, X., et. al., 1999)], PBIL [Zhengra, H., et. al., 1999], GA [(Goldberg, D.,1989), (Michalewicz, Z., 1992)], Stud-GA [Khatib, W., et. al., 1998], etc. categorized as population based EAs [(Back, T., 1996), (Dasgupta, D., et. al., 2001), (Solomon, R., 1997)], PBO is not a population based EA since it does not involve reproduction or generation of children and can be categorized as a nature inspired or a geo-sciences based EA in the taxonomy of evolutionary optimization techniques since PBO is inspired from the geo-sciences phenomenon of plate tectonics which has never been used as a contributory domain to evolutionary computation.

The proposed PBO is capable of incorporating lower prioritized decision variables iteratively and achieve a better solution with each iteration and also incorporates the concept of dynamic adjustment of weights according to the conveyor-belt principle of plate tectonics. The

performance of PBO is analyzed on a set of 14 standard benchmark functions and also the performance results are compared with eight other evolutionary optimization algorithms. Based on the above characteristic strengths of PBO, its applicability is demonstrated on the travelling salesman problem on a 5-cityset and later extended to standard TSP problem instances. The performance of PBO is also compared with the twelve other recent evolutionary optimization techniques for the TSP problem for all the datasets taken.

The goals of this chapter are threefold. First, the chapter gives a general presentation of the new optimization method called PBO. This is done by first studying the concepts of plate tectonics, and then generalizing it to obtain a general-purpose optimization algorithm. Second, the chapter compares and contrasts PBO with other evolutionary optimization methods. This is done by looking at the commonalities and differences from an algorithmic point-of-view, and also by comparing their performances on a set of benchmark functions. Thirdly, the chapter demonstrates the applicability of PBO on engineering applications. For the purpose, PBO is applied on the travelling salesman problem first with a 5-city dataset for demonstration and then extended to five standard TSP problem instances considering dynamic environments. The chapter also compares and contrasts PBO with the other evolutionary optimization methods for the TSP application and found good results as compared to the other evolutionary algorithms.

The rest of the chapter is organized as follows: Section 6.1 reviews the ideas and mathematics of plate tectonics and formulates the equations for the governing forces towards plate motion and also presents the plate motion model. Section 6.2 discusses how plate tectonics can be used to formulate a general optimization algorithm. Section 6.3 provides the simulation results comparing PBO with other optimization methods on a set of standard benchmark functions. This section also adapts the PBO for the TSP problem. Section 6.4 concludes the work and discusses the observations made.

6.1. Plate Tectonics

The concepts of plate tectonics theory are modeled for the development of the plate tectonics based optimization (PBO) technique. Suppose that there is a problem and some candidate solutions analogous to the plate and the temperatures at different points on the plate characterized by the plate mobility index (generally termed as the fitness function in most evolutionary algorithms), respectively. PMI is the function of the contributing forces of ridge-push, slab-pull and the other external forces acting at different points on the plate: the ridge push force is analogous to the various decision variables which characterize the fitness (or the goodness) of a candidate solution and are problem specific, the slab-pull force is analogous to the associated weights (and dynamic adjustment of weights) of these decision variables and the external forces are analogous to the tolerance level of the solution. PMI is thus proportional to

the weighted sum of the current set of decision variables, the tolerance level is used to identify the higher priority decision variables. The PMI values are calculated for each solution using the set of higher prioritized decision variables those having weights greater than the tolerance level. The iterative refinement is next performed by iterative reduction in this tolerance level in order to incorporate lower prioritized decision variables and the weights of this newly created set of decision variables is adjusted relative to each other by considering the tradeoff between the current set of decision variables achieved through normalization of the weights of these variables in each iteration. The above process is repeated until the PMI becomes zero at which point the solution obtained is the optimal solution (which corresponds to the equilibrium state of the plate).

This section details the forces acting on the plate which contribute towards the formulation of the decision variable and hence PMI. The section is subdivided into four subsections where the first section describes the forces that act on a plate, the second section discusses the driving forces leading to plate movement and their proportionality relationship with the plate motion, third section formulates the equations governing these forces acting towards plate motion and the last section presents the plate motion model.

6.1.1. Forces acting on a plate

Plate tectonics is basically a kinematic phenomenon. The contributing forces of plate motion can be categorized into three major types: those related to mantle dynamics, those related to earth's rotation and those related to gravity as described below [(Iaffaldano, G., et. al., 2009), (Summeren, J. V., et. al., 2012), (Matsuyama, T., et. al., 2010)].

A. Forces related to mantle dynamics: the frictional force (force due to basal drag) and the gravitational force (force due to slab suction) [(Iaffaldano, G., et. al., 2009), (Summeren, J. V., et. al., 2012)].

B. Forces related to gravity: the ridge push force caused by the gravitational sliding away from a spreading ridge and the slab-pull force driven by the weight of cold, dense plates sinking into the mantle at trenches and thought to be the greatest force acting on the plates [(Iaffaldano, G., et. al., 2009), (Summeren, J. V., et. al., 2012), (Matsuyama, T., et. al., 2010)].

C. Forces related to Earth rotation: the tidal force, the pole flight force, force due to the coriolis effect and the other forces caused by the smaller deformation effects of the crust.

6.1.2. Governing Forces on a Plate

The actual vector of a plate's motion is a function of all the major forces acting upon the plate. Hence, the major contributing forces towards the plate movement and their relationships to the plate motion are modeled below.

A. Ridge Push force

Pressure build up under the plates causes the rifting of plates. Movement of plates occurs to minimize the pressure. Higher the pressure (P), greater is the velocity of movement (v) [(Richards, M., et. al, 2000), (Solomon, S. C., et. al., 1975), (Uyeda S., et. al., 1975)]. Hence,

$$v \propto P \quad (6.1)$$

B. Slab Pull force

Cooling of the landmass causes increase in the plate density resulting in sinking of the plates at the subduction zones. Greater is the density of plates ρ , the greater is the force of sinking (more gravitational pull) [(Richards, M., et. al, 2000), (Solomon, S. C., et. al., 1975)] i.e. greater plate movement. Hence,

$$v \propto \rho \quad (6.2)$$

C. Plate Drag (Basal Drag)

Friction with the lower layers causes the plate movement due to frictional drag. Greater the friction, nearer is the velocity of movement of the plate to the velocity of lower layer [(Richards, M., et. al, 2000), (Solomon, S. C., et. al., 1975)]. Let the velocity of lower layer is v_0 and the coefficient of friction is μ . Then the velocity of plate movement v is defined as

$$v = \mu v_0 \quad (6.3)$$

From, (6.1), (6.2) and (6.3),

$$v \propto \rho \cdot P \cdot \mu v_0$$

Since, v_0 is a constant,

$$\begin{aligned} v &\propto \rho \cdot P \cdot \mu \\ v &= k \rho P \mu \end{aligned} \quad (6.4)$$

where ρ is the density of the plate, P is the pressure build-up under the plate and μ is the coefficient of friction with the lower layer. Plate movement occurs to reach equilibrium state [Solomon, S. C., et. al., 1975] so that the net force acting on the plate becomes zero i.e.

$$\vec{F}_{RP} + \vec{F}_{SP} + \vec{F}_{PD} = 0 \quad (6.5)$$

where F_{RP} is the force on the plate due to ridge push effect, F_{SP} is the slab-pull force and F_{PD} is the force due to plate drag. The velocity of plate movement is governed by equation (6.4). As the net force on the plate approaches zero, the velocity on the plate also decreases and the plates settle into an equilibrium state.

6.1.3. Equations governing the forces

The force due to ridge-push effect is derived in the section below. The density (D) of a plate can be defined by equation (6.6) as given in [Richards, M., et. al, 2000].

$$\text{Density } (D) = 1000 \times \left(1 - \frac{(T + 288.94)}{508929.2 \times (T + 68.13)} \times (T - 3.98)^2 \right) \quad (6.6)$$

where T is the temperature in $^{\circ}\text{C}$, and D is the density in kg/m^3 as a function of temperature. For convenience, it is defined that $\alpha = 288.94$, $\theta = 3.98$, $\beta = 508929.2$ and $\gamma = 68.13$. Hence,

$$D = 1000 \times \left(1 - \frac{(T+\alpha)}{\beta \times (T+\gamma)} \times (T - \theta)^2 \right) \quad (6.7)$$

Differentiating D with respect to T ,

$$\begin{aligned} dD &= 1000 \left[\frac{0 - ((T + \alpha) \times 2(T - \theta)dT + dT(T - \theta)^2)(\beta(T + \gamma)) - \beta dT(T + \alpha)(T - \theta)^2}{\beta^2(T + \gamma)^2} \right] \\ &= \frac{\beta dT(T + \alpha)(T - \theta)^2 - \beta(T + \gamma)(T - \theta)(2(T + \alpha) + (T - \theta))dT}{\beta^2(T + \gamma)^2} \\ &= \frac{dT}{\beta(T + \gamma)^2} [(T + \alpha)(T^2 + \theta^2 - 2\theta T) - (T^2 + T(\gamma - \theta) - \theta\gamma)(3T + 2\alpha - \theta)] \\ &= \frac{dT}{\beta(T + \gamma)^2} [-2T^3 + (2\theta - 3\gamma - \alpha)T^2 + (4\theta\gamma - 2\alpha\gamma)T + (\alpha\theta^2 + 2\alpha\theta\gamma - \theta^2\gamma)] \end{aligned}$$

Let $I = -2$, $J = (2\theta - 3\gamma - \alpha)$, $K = (4\theta\gamma - 2\alpha\gamma)$ and $L = (\alpha\theta^2 + 2\alpha\theta\gamma - \theta^2\gamma)$

$$dD = \frac{dT}{\beta(T + \gamma)^2} [IT^3 + JT^2 + KT + L] \quad (6.8)$$

Now, the ridge push force F acting on the plate can be given by equation (6.9) where the force is defined as $\text{mass}(m) \times \text{acceleration due to gravity}(g)$ [(Richards, M., et. al, 2000), (Solomon, S. C., et. al., 1975)]. Also, $\text{mass} = \text{Volume} \times \text{Density}(D)$ and $\text{Volume} = \text{Area}(a) \times \text{Depth}(y)$ [(Richards, M., et. al, 2000), (Solomon, S. C., et. al., 1975)] where 'a' is the area of the plate under consideration.

$$\therefore F_{\text{Ridge-Push}} = mg = a.y.D.g$$

And, hence, expressing $F_{\text{Ridge-Push}}$ in differential terms,

$$dF_{\text{Ridge-Push}} = g.a.dy.dD$$

Substituting the value of dD from equation (6.8),

$$= g.a.dy \cdot \frac{dT}{\beta(T + \gamma)^2} (IT^3 + JT^2 + KT + L)$$

Now, $dT = k.dy$ or $dy = dT/k$ (see at the bottom of the page for proof), where 'k' is a constant.

$$dF_{\text{Ridge-Push}} = \frac{gA(dT)^2}{k\beta(T + \gamma)^2} (IT^3 + JT^2 + KT + L) \quad (6.9)$$

Now, integrating twice to obtain $F_{\text{Ridge-Push}}$,

$$F_{\text{Ridge-Push}} = \iint \frac{gA(dT)^2}{k\beta(T + \gamma)^2} (IT^3 + JT^2 + KT + L) \quad (6.10)$$

Let $T + \gamma = T'$

$$F_{Ridge-Push} = \int \frac{gA}{k\beta} \left(\frac{IT'^2}{3} + (J - 3I\gamma)T' + (3I\gamma^2 - 2\beta\gamma)\ln T' + \frac{I\gamma^3 + K\gamma - J\gamma^2 - L}{T'} \right) + K \quad (6.11)$$

Integrating (6.11),

$$F_{Ridge-Push} = \frac{gA}{k\beta} \left(\frac{IT'^3}{12} + \frac{(J - 3I\gamma)T'^2}{3} + T'(3I\gamma^2 - 2\beta\gamma)(\ln T' - 1) + (I\gamma^3 + K\gamma - J\gamma^2 - L)\ln T' \right)$$

Substituting the value of T' from (6.10), the ridge push force (in kg/ms^3) is given by equation (6.12) where T ranges from T_0 to T_m . T_0 and T_m are the temperatures at the surface and at the bottom of the molten material beneath the plate.

$$F_{Ridge-Push} = \frac{gA}{k\beta} \left(\frac{T_0}{T_m} \frac{I(T + \gamma)^3}{12} + \frac{(J - 3I\gamma)(T + \gamma)^2}{3} + (3I\gamma^2 - 2J\gamma)(T + \gamma)(\ln(T + \gamma) - 1) + (I\gamma^3 + K\gamma - J\gamma^2 - L)\ln(T + \gamma) \right) \quad (6.12)$$

Force due to frictional drag $F_{Plate Drag}$ (in kg/ms^3) is given by equation (6.13) [(Matsuyama, T., et. al., 2010), (Richards, M., et. al, 2000)].

$$F_{Plate Drag (Frictional Drag)} = \frac{\eta A v_w}{y_0} \quad (6.13)$$

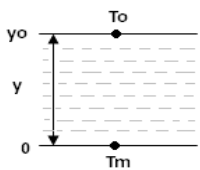
Where η is the coefficient of friction, A is the area of the plate under consideration in m^2 , v_w is the viscosity of the molten material beneath the plate responsible for frictional force in $Pa.S$ and y_0 is the depth of ocean at location of the plate in meters where the viscosity is given by equation (6.14) [Richards, M., et. al, 2000].

$$Viscosity (v_w) = a \times 10^{\frac{b}{(T-c)}} \quad (6.14)$$

where, $a = 2.414 \times 10^{-5} Pa.S$, $b = 247.8 K$ and $c = 140 K$.

Slab-pull force is given by equation (6.15). Force due to slab-pull effect is equivalent to the reduction in the buoyant force of water which is in turn equivalent to the reduction in the

Consider a liquid layer beneath the plate under consideration. Let the depth of the liquid be y with the initial and the final positions as 0 and y_0 respectively. Let the temperature at the surface and at the bottom be T_0 and T_m respectively as shown in the figure below. Then, it can be said that,



$$\frac{y-y_0}{y_0} = \frac{T-T_0}{T_0-T_m} \Rightarrow \frac{y}{y_0} = 1 + \frac{T-T_0}{T_0-T_m} \frac{y}{y_0} = \frac{T-T_m}{T_0-T_m}$$

$$dT = \frac{dy(T_0-T_m)}{y_0} \Rightarrow dT = k \cdot dy \text{ where } k = \frac{(T_0-T_m)}{y_0}.$$

weight of the water displaced [Solomon, S. C., et. al., 1975]. Hence, the slab-pull force (in kg/ms^3) is given by

$$\begin{aligned} \text{Slab - Pull force } (F_{\text{slab-pull}}) &= \text{Reduction in buoyant force of water} = \\ \text{Reduction in the weight of water displaced} &= (V_0 - V_F).D.g \end{aligned} \quad (6.15)$$

where V_0 is the initial velocity of the plate in m/s , V_F is the final velocity of the plate m/s , D is the density of the plate in kg/m^3 and g is the acceleration due to gravity in m/s^2 .

$$\therefore F_{\text{slab-pull}} = (V_0 - V_F).D.g.$$

$$\text{Since, } V_F = V_0(1 + \beta\Delta T) \quad (6.16)$$

where ΔT is the change in temperature in K and β is the contraction coefficient of the material of the plate [Solomon, S. C., et. al., 1975]. Substituting (6.16) in (6.15), the slab-pull force on the plate is derived in equation (6.17).

$$F_{\text{slab-pull}} = -V_0\beta.\Delta T.D.g \quad (6.17)$$

It is observed that the forces of slab-pull and ridge-push are mutually exclusive to each other which mean that only one force acts at a time on a plate (discussed in detail in the next section). Hence, at a given temperature, either the ridge-push force and the frictional drag force act on the plate or the slab-pull and the frictional drag force. The reason for the above is that as the temperature increases, the slab-pull force decreases (due to the decrease of density with temperature) and becomes zero at a temperature which is termed as the critical temperature T_c at which point the ridge-push force starts acting on the plate and increases thereafter with the increase of temperature. This is because with the increasing temperature, the pressure buildup beneath the plate increases and hence the ridge push force on the plate increases. However, the force due to frictional drag is an opposing force to the force exerted by either slab-pull or ridge push and hence, the net force acting on a plate is given by equations (6.18) and (6.19) respectively. Equation (6.18) represents the net force when the ridge-push force acts and equation (6.19) presents the net force when the slab pull force acts on the plate. If the temperature $T \leq T_c$, then the slab-pull force acts on the plate and the net force on the plate is given by equation (6.18) else if $T > T_c$, then the ridge-push force acts on the plate and the net force is given by equation (6.19).

If $(T > T_c)$,

Netforce (Ridge Push)

$$\begin{aligned} &= \text{Ridge Push force } (F_{\text{Ridge-Push}}) - \text{Frictional Drag } (F_{\text{Frictional Drag}}) \\ &= \frac{gA}{k\beta} \left(\frac{T_0}{T_m} \frac{I(T + \gamma)^3}{12} + \frac{(J - 3I\gamma)(T + \gamma)^2}{3} + (3I\gamma^2 - 2J\gamma)(T + \gamma)(\ln(T + \gamma) - 1) \right. \\ &\quad \left. + (I\gamma^3 + K\gamma - J\gamma^2 - L)\ln(T + \gamma) \right) - \frac{\eta A v_w}{y_0} \end{aligned} \quad (6.18)$$

else if $(T \leq T_c)$,

Netforce (Slab Pull) =

$$\text{Slab Pull force } (F_{\text{Slab-Pull}}) - \text{Frictional Drag } (F_{\text{Frictional Drag}}) = -V_0 \beta \cdot \Delta T \cdot D \cdot g - \frac{\eta A v_w}{y_0} \quad (6.19)$$

6.1.4. Plate Motion Model

This section presents the graph plots of the behavior of each of the contributing forces of plate movement i.e. ridge-push, slab-pull and frictional drag with respect to the temperature as presented by figures 6.1(b)-(e). The three major contributing forces towards plate motion are the ridge push, slab pull and the frictional drag the mathematical formulations of which were derived in the previous section in equations (6.12), (6.13) and (6.17). Also, the viscosity curve is presented in figure 6.1(a). Figures 6.1(e) and 6.1(f) present the graphs of the net force acting on the plate for the ridge push and the slab pull cases respectively. For studying the effects of varying temperatures on the three forces, the temperature ranges were taken from $(0 - 5000) K$ with an interval of $250 K$ as shown in Table 6.1 and 6.2. Table 6.1 is the table representing the values of viscosity, frictional drag, slab pull force and the net force ($net - force = slab\ pull - frictional\ drag$) values with the increasing temperature when the slab pull force acts between the temperature ranges from $0 - 500 K$ i.e. when $(T \leq T_c)$. Table 6.2 is the table representing the values of viscosity, frictional drag, ridge push force and the net force ($net - force = ridge\ push - frictional\ drag$) values with the increasing temperature when the ridge push force acts between the temperature range from $750 - 5000 K$ i.e. when $(T > T_c)$. Figure 6.1(a) is the graph plot of the viscosity v/s temperature since viscosity is the factor which governs the force due to frictional drag as given by equation (6.13). From the graph, it may be noted that the viscosity decreases with the increasing temperature and the viscosity curve represents a rectangular hyperbola. Figure 6.1(b) presents the graph plot of the frictional drag force v/s temperature and exhibits the same graph pattern as the viscosity curve since the frictional drag is directly proportional to the viscosity. Figures 6.1(c) and (d) present the ridge push and the slab-pull forces graph plots. The graph 6.1(c) depicting the effect of temperature on the ridge push force is an exponentially increasing curve with respect to temperature and hence, the increase of temperature leads to an exponential increase of the ridge-push force. This is because with the increase of temperature, the pressure buildup beneath the plate increases and causes the ridge-push force to increase exponentially and cause the movement of the plate as demonstrated by equation (6.4). However, the graph 6.1(d) depicting the effect of temperature on the slab-pull force is a vertical parabolic curve and the slab-pull curve is a decreasing curve which means that the slab-pull force decreases abruptly with the increasing temperature. This is because the slab-pull effect occurs by the

downward gravitational pull due to increased density of the plate with the cooling of the landmass and hence increase of temperature will have a reverse effect. The magnitude ranges of the ridge push and the slab pull forces indicated by the graph plots in figures 6.1(c) and (d) verify the fact that the slab pull is the greatest force acting on the plates. From the graphs that were plotted, it can be observed that the phenomenon of ridge push and slab pull are mutually exclusive to each other. This can be observed from the fact that the slabpull force acts on the plate as the temperature increases from $0 - 500\text{ K}$ when the slab-pull force on the plate is $6.86 \times 10^5 \text{ kg/ms}^3$ and abruptly decreases thereafter reaching zero at the temperature of 635 K as shown in Table 6.1. As the temperature is further increased, the ridge push force becomes predominant and acts between the temperature range of $750 - 5000\text{ K}$ increasing from $49.52 - 868.13 \text{ kg/ms}^3$ which is a substantial increase and follows an exponential curve as indicated by Table 6.2. The temperature range $500 - 750\text{ K}$ where the force acting on the plate changes from slab pull to ridge push is termed as the gray interval since in this interval neither of the two forces is predominant. The point at which these two forces reach an equilibrium state and deviating from this point would lead to one force becoming predominant is called as the critical temperature T_c . In the graph plots, the critical temperature is 635 K at which point the slab-pull force becomes zero and the ridge-push force acts thereafter. Figures 6.1(e) and 6.1(f) represent the net force acting on the plate due to ridge push and slab pull taking into account the opposing force of frictional drag the mathematical equations for which were derived in section 6.1.3 and presented in equations (6.18) and (6.19). Figure 6.1(e) represents the net force acting on the plate under the ridge-push condition. Since, the magnitude of the frictional drag force is small as compared to the ridge-push force acting on the plate which increases exponentially with the increase in temperature and the frictional force decreases with the increase in temperature (rectangular hyperbolic pattern), the resultant force acting on the plate increases exponentially with the increase of temperature which is also reflected from the graph in figure 6.1(e).

From the graph, it can be observed that the net force acting on the plate increases from $5.23 - 807.60 \text{ kg/ms}^3$ with the temperature change from $750 - 5000\text{ K}$ as shown by Table 6.2. Figure 6.1(f) represents the net force acting on the plate under the slab-pull condition. Since, the magnitude of the frictional drag force is small as compared to the slab-pull force acting on the plate and both the slab pull and the frictional drag forces decrease with the increase in temperature exhibiting a parabolic and a rectangular hyperbolic pattern, the resultant force acting on the plate also decreases with the increase of temperature and exhibits a parabolic pattern which is also reflected from the graph in figure 6.1(f). From the graph and the tabular results in Table 6.1, it can be observed that the net force acting on the plate decreases from $(1.96 \times 10^6) - (6.40 \times 10^5) \text{ kg/ms}^3$ with the temperature increase from $0 - 500\text{ K}$.

Figure 6.2 presents the combined graph plot of the net force acting on the plate v/s temperature for both the ridge-push effect and the slab-pull case. The net force acting on the plate under ridge-push condition ($net - force = ridge\ push - frictional\ drag$) and that under slab-pull condition ($net - force = slabpull - frictional\ drag$) can be given by equations (6.19) and (6.18) respectively. The color codes are blue for the net force v/s temperature plot under the slab-pull condition while red is for ridge push condition. From the graph, it can be observed that the slab-pull force acts on the plate between the temperature ranges from $0 - 500\ K$ i.e. when ($T \leq T_c$) after which the ridge-push force becomes predominant and acts between the temperature range from $750 - 5000\ K$ i.e. when ($T > T_c$). From the graph plot, it can be clearly observed that the phenomenon of slab-pull and ridge-push are mutually exclusive to each other.

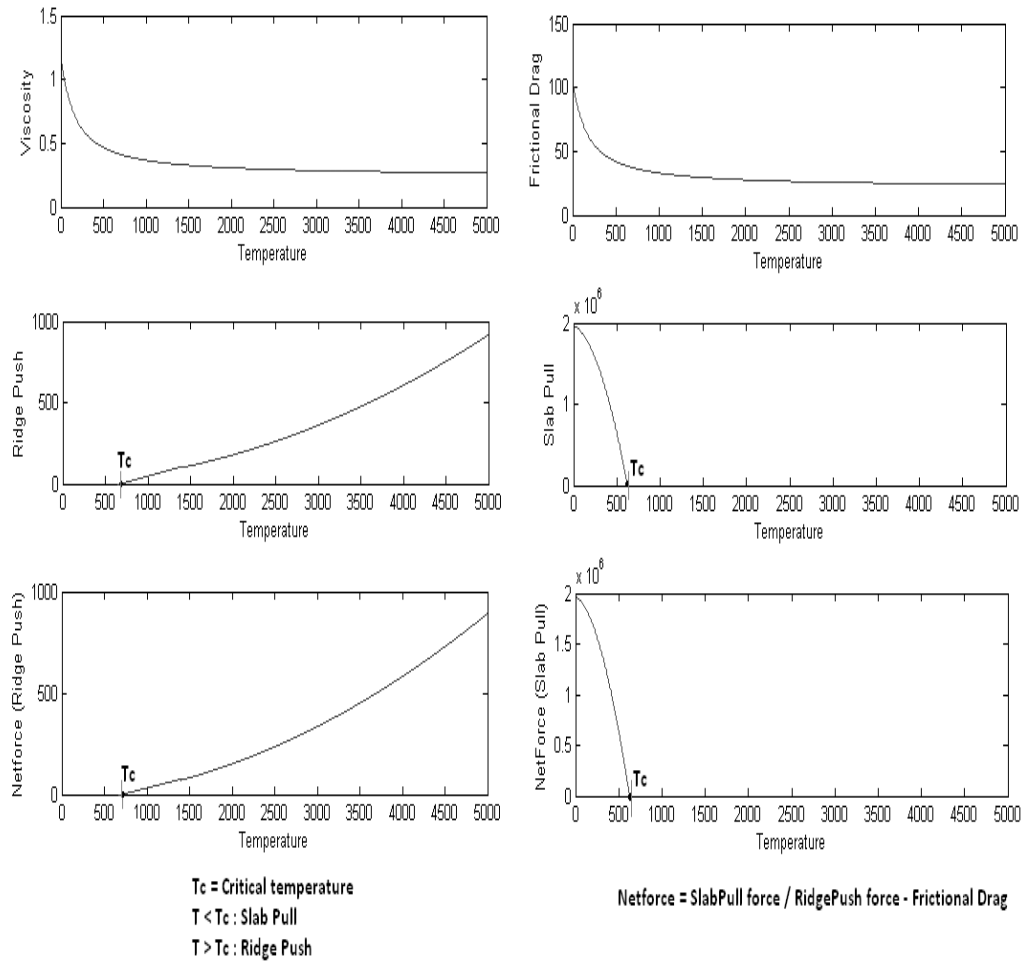


Figure 6.1(a)-(f): Graph plots of the behavior of each of the contributing entities in plate movement v/s temperature (T) (a) Viscosity (b) Frictional Drag (c) Ridge-push (d) Slab-pull (e) Net Force (due to Ridge-push) and (f)-Net Force (due to Slab-pull). If ($T \leq T_c$), $net - force = slab\ pull - frictional\ drag$ and if ($T > T_c$), $net - force = ridge\ push - frictional\ drag$.

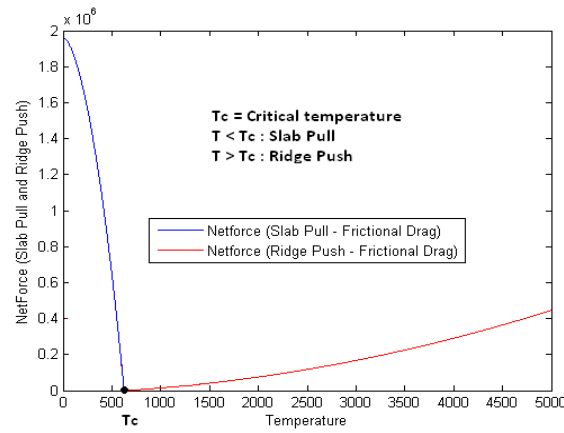


Figure 6.2: Combined graph plot of the net force acting on the plate v/s temperature for the slab-pull ($T \leq T_c$) and the ridge- push condition ($T > T_c$).

Table 6.1: Table representing the values of viscosity, frictional drag, slab pull force and the net force ($net - force = slab\ pull - frictional\ drag$) values with the increasing temperature when the slab pull force acts between the temperature ranges from 0 – 500 K i.e. when ($T \leq T_c$).

Temperature (< Tc) (K)	Viscosity (Pa. S)	Frictional Drag (kg. m/s ³)	Slab Pull (kg. m/s ³)	Net Force (Slab Pull – Frictional Drag) (kg. m/s ³)
0	1.16	100.00	1.96 E+06	1.96 E+06
250	1.17	105.54	1.59 E+06	1.56 E+06
500	0.63	61.19	6.86 E+05	6.40 E+05
Tc = 635 K				

Table 6.2: Table representing the values of viscosity, frictional drag, ridge push force and the net force ($net - force = ridge\ push - frictional\ drag$) values with the increasing temperature when the ridge push force acts between the temperature range from 750 – 5000 K i.e. when ($T > T_c$).

Temperature (> Tc) (K)	Viscosity (Pa. S)	Frictional Drag (kg. m/s3)	Ridge Push (kg. m/s3)	Net Force (Ridge push – Frictional Drag) (kg. m/s3)
Tc = 635 K				
750	0.48	44.29	49.52	5.23
1000	0.41	37.39	54.55	17.16
1250	0.37	33.69	76.56	42.85
1500	0.35	31.39	95.15	63.76
1750	0.33	29.82	124.83	82.02
2000	0.32	28.69	157.93	114.40
2250	0.31	27.84	195.04	150.55
2500	0.30	27.17	236.16	190.54
2750	0.30	26.63	281.30	234.43
3000	0.29	26.19	330.46	282.23
3250	0.29	25.83	383.62	333.99
3500	0.28	25.51	440.80	389.70
3750	0.28	25.24	501.99	449.39
4000	0.28	25.01	567.20	513.05
4250	0.28	24.81	636.41	580.70
4500	0.27	24.63	709.64	652.34
4750	0.27	24.47	786.88	727.97
5000	0.27	24.32	868.13	807.60

6.2. Plate Tectonics Based Optimization (PBO)

This section discusses how the concepts of plate tectonics demonstrated in the previous sections can be applied to formulate an optimization algorithm.

6.2.1. PBO Definitions

This section provides some definitions as a first step towards formalizing the PBO algorithm. R refers to the set of real numbers, Z refers to the set of integers, and $\emptyset$ refers to the empty set.

Definition 1: Candidate Solution: ρ_0 is the density of water at temperature T_0 . T_0 is the ambient temperature. T_m is the temperature at the point of spread and is different at each point of the plate analogous to the candidate solutions of the problem. Let N is the number of these points.

Definition 2: Decision variables: To each of the of N points where $N \in Z$, a *temperature* is associated $T_M \in Z$ such that at the i^{th} point where i ranges from $i = \{1, \dots, n\}$, the temperature is T_M^i . This T_M^i is characteristic of the solution i and is determined by several decision variables $\{c_1, c_2, c_3, \dots, c_p\}$ (analogous to the SIVs in BBO [Simon, D., 2008] and to the fitness function variables or the attributes in other EAs) which determine the fitness of the solution under consideration (in this case, fitness of the solution is analogous to the plate mobility index). The choice of the decision variables is problem dependent.

Definition 3: Weights: *Weight* (w_c) $\in R \{0, \dots, 1\}$ of each decision variable ranges from $w_{c_1}, w_{c_2}, w_{c_3}, \dots, w_{c_p}$. The adjustment of weights of the several decision variables is required when new decision variables are added iteratively to the set of already existing decision variables. This adjustment of weights is done through normalization by dividing these weights by the sum of the weights of all the currently considered decision variables.

Definition 4: The force f upon which the value of the plate mobility index (PMI) (defined in definition 5) depends, acts either in the same direction as *slabpull force* (S) or *ridgepush force* (R) or opposite to them. Consequently, it may support or weaken the R and S forces. These forces contribute towards the formulation of the PMI (or the fitness) of the candidate solution and are defined as:

[a] *Ridgepush force* (R) is the induced cause of movement of the plate and is modeled in the form of decision variables (attributes) c_j which characterize the given candidate solution. A decision variable c_j^α is defined which is the variable c_j for which the corresponding weight w_{c_j} is greater than or equal to the tolerance level α (defined in 4[c]) and are those decision variables which are included in the calculation of PMI in the current iteration.

[b] *Slabpull force* (S) is a driving force and therefore, S models the weight w_{c_j} of a particular decision variable such that, a change in the weights associated with the several decision variables specified is possible. This concept of alteration of the weights w_{c_j} through normalization for the newly generated set of decision variables in each iteration (controlled by the tolerance level as explained next), acts as the governing factor in the iterative refinement of PMI and **hence forms the basis of the proposed optimization process which is supported by the fact that the slab-**

pull force is thought to be the greatest force acting on the plates [(Summeren, J. V., et. al., 2012), (Matsuyama, T., et. al., 2010)].

[c] *External forces* acting on the plate such as the coriolis effect, tidal drag, pole flight force etc. are analogous to the secondary decision variables contributing to plate movement which affect the value of c_j and are assigned as the *tolerance level* (α). The tolerance level α can be initialized to a maximum value by selecting the maximum of the weights assigned to decision variables i.e. $\alpha_{init} = \max_{j=1}^p (w_{c_j})$. Thus, in the initial iteration, only the highest prioritized decision variable is considered in the current set of decision variables. The tolerance level is decremented at each iteration i.e. $\alpha = \alpha - \tau$ where the choice of τ is problem dependent.

Definition 5: Plate Mobility Index (PMI): $R \rightarrow R$ is a measure of the goodness of the candidate solution and is equivalent to the fitness function as generally termed in other optimization algorithms. The aim is to refine the value of the plate mobility index until the equilibrium state is reached (see definition (7)). Higher the value of *PMI*, higher is the value of T_M^i , and hence greater is the plate movement and lower is the value of the solution i . The PMI for a given candidate solution say i can be defined as given by equation (6.20) where T is the threshold level defined in definition (7).

$$PMI = \frac{\left| \sum_{j=1}^p w_{c_j}^{\alpha} \cdot c_j^{\alpha} - T \right|}{w_{c_j}^{\alpha_{init}} \cdot c_j^{\alpha_{init}}} \quad (6.20)$$

where, α_{init} is the initial value of the tolerance level (defined in 4(c)) such that the decision variables' set under consideration contains the highest prioritized decision variable in it with the highest weight i.e. $w_{c_j}^{\alpha_{init}} = w_{\max_{c_j} \sum_{j=1}^p (w_{c_j})} = 1$. The PMI value thus obtained is an index which

indicates the level of plate mobility in the current iteration. Since, the tolerance level is decremented with each iteration, the no. of decision variables contributing to the plate mobility index increases with each iteration thereby increasing the efficiency with each iteration as more and more decision variables are included after dynamic adjustment of weights thereby iteratively refining the results until the optimal solution is found.

Definition 6: From definitions 4(a), (b) and (c), the following relationship is established in equation (6.21).

$$\sum_{j=1}^p (w_{c_j}^{\alpha}) = 1 \quad (6.21)$$

This means that for T_M^i composed of the decision variables, the weights attached to the several decision variables are altered (dynamic adjustment of weights) and the next iteration is performed with these altered weights. The weights are altered by dividing each weight by the sum of the weights of all the decision variables **such that the sum of the normalized weights is**

always equal to 1 i.e. the weights are in equilibrium with each other. The above characteristic is modeled from the conveyor belt principle of the plate tectonics theory which states that the plate movement caused by the slab-pull (S) and the ridge-push forces (R) are in equilibrium with each other.

Definition 7: Equilibrium state: For finding an optimal solution to the problem which corresponds to the equilibrium state of the plate, a threshold T is defined. The equilibrium state of the plate is the temperature T_M^i at which the net force acting on the plate is zero which corresponds to the solution for which the plate mobility index is zero (≤ 0 for a minimization function and ≥ 0 for a maximization function). The threshold can be set with respect to a hypothetical ideal solution which takes into account the complete set of decision variables with their normalized weights and can be represented by $\sum_{j=1}^p \frac{w_{c_j}}{\sum_{j=1}^p (w_{c_j})} \cdot c_j$. Thus the threshold value

T can be set to $\sum_{j=1}^p \frac{w_{c_j}}{\sum_{j=1}^p (w_{c_j})} \cdot c_j \times (1 \pm (1 - \text{desired accuracy}))$ for minimization and maximization functions, respectively.

6.2.2. PBO Algorithm

The PBO algorithm is formally presented in figure 6.3 and summarized below.

1. Consider a plate where the temperatures at different points on the plate are considered as the candidate solutions. The temperatures associated with each of the candidate solutions is characteristic of the solution and is directly proportional to the plate mobility index.
2. The value of the PMI is calculated using equation (6.20) in definition 5 for the considered candidate solutions $i = \{1, \dots, n\}$ using the set of higher prioritized decision variables c_j^α i.e. those having weights greater than or equal to the tolerance level α . The tolerance level is decided using definition 4 (c).
3. This step is the iterative refinement step where the tolerance level is reduced by τ each time i.e. $\alpha = \alpha - \tau$ (analogous to the reduction in the effect of the external forces contributing towards PMI in order to bring the plate nearer to the equilibrium condition) in order to incorporate lower prioritized decision variables and hence formulate the new set of decision variables which better represent the PMI of the given candidate solution.
4. Next, the weights of this new set of decision variables are altered by converting them into normalized weights obtained by dividing each decision variable's weight by the sum of the weights of all decision variables such that the sum of their normalized weights equals 1. This step basically adjusts the weights of the newly created set of decision variables dynamically (in accordance with the conveyor-belt principle of plate tectonics). This step explains the working mechanism of the proposed algorithm. The algorithm basically considers the tradeoff between the several decision variables and based on this tradeoff, it adjusts their

weights which can be achieved by the normalization of weights which in turn refines the PMI value in each iteration.

5. Repeat the steps (2)-(4) until the PMI becomes zero at which point the corresponding solution is output as the optimal solution by the algorithm (analogous to the plate reaching the equilibrium condition). The optimal solution found suggests the optimal set of decision variables that are sufficient to obtain the desired efficiency (defined in terms of the threshold value) and hence the tolerance level (or the decision variables set) is also mentioned along with the solution which comprises the optimal solution.

The above process sets the plate mobility index to zero, the corresponding solution (termed as the associated temperature T_M^i) being the optimal solution found at the end of the algorithm. This means that the proposed PBO algorithm adapts to dynamic changes in the importance of the several decision variables used for assessing the goodness of the solutions by altering their weights in order to incorporate lower prioritized decision variables and hence achieves a better solution with each iteration. The algorithm for the plate tectonics based optimization is presented in figure 6.3.

-
1. Form an initial set of candidate solutions.
 2. Calculate the plate mobility index values for each solution,

$$PMI = \frac{\left| \sum_{j=1}^p w_{c_j}^\alpha \cdot c_j^\alpha - T \right|}{w_{c_j}^{\alpha_{init}} \cdot c_j^{\alpha_{init}}}$$

where $c_j^\alpha \rightarrow c_j$ for which $w_{c_j} \geq \alpha$. $\alpha_{init} = \max_{j=1}^p (w_{c_j})$. T is the predefined threshold value. $w_{c_j}^{\alpha_{init}} = 1$.

3. Reduce the tolerance level α by τ in order to incorporate lower prioritized decision variables and hence, formulate the new set of decision variables, c_j^α .

$\alpha = \alpha - \tau$; (iterative refinement)

4. Alter the weights corresponding to the new set of decision variables by replacing $w_{c_j}^\alpha$ by

$$\frac{w_{c_j}^\alpha}{\sum_{j=1}^p (w_{c_j}^\alpha)} \text{ for each } c_j^\alpha \text{ such that } \sum_{j=1}^p (w_{c_j}^\alpha) = 1 \text{ and the next iteration is performed with}$$

these altered weights. (dynamic adjustment of weights).

5. Return to step (2) until PMI reaches zero at which point the corresponding solution is the best fit solution.
-

Figure 6.3: Plate-tectonics Based Optimization (PBO)

6.2.3. Mathematical formulation of PBO

Figure 6.4 presents the approach for problem solving through PBO. The development of the PBO method begins with defining the ideal state of the overall objective, and specifies the complete set of decision variables with their ideal weights involved in the process. The ideal state of the objective is represented by the ideal model, the IDEAL. For practical purposes, the ideal good value for decision variables (or attributes) is defined as the best values which exist within the range of values of decision variables. The IDEAL, then, is simply the solution that has all the best values of decision variables. It is very unlikely that a certain solution has the best values for all attributes. Instead, a variety of alternatives may be used to simulate the ideal state. For this reason, the IDEAL is not to be considered as feasible alternatives, but it is used only as reference to which other alternatives are quantitatively compared. The numerical difference resulting from comparison represents the effectiveness of alternatives to achieve the ideal state of find a feasible solution which is as close as possible to the ideal point. The objective function for finding such a solution can be formulated as

$$\text{Minimize } \delta(SOL_i, IDEAL) \quad (6.22)$$

where SOL_i represents the candidate solution, and δ the difference from the ideal point, respectively. Thus the problem, and its solutions depend on the choice of ideal point, IDEAL, and the difference metric, the plate mobility index, used in the model. The PBO method determines the point in the H region which is “the closest” to the ideal point. To implement the above approach, it is assumed that a complete set of candidate solutions consisting of $1, 2, \dots, n$ solutions, and $1, 2, \dots, p$ decision variables is available. Hence, the decision variables matrix C_j can be represented by:

$$C_1 = \begin{bmatrix} c_{11} \\ c_{21} \\ \vdots \\ c_{n1} \end{bmatrix}, C_2 = \begin{bmatrix} c_{12} \\ c_{22} \\ \vdots \\ c_{n2} \end{bmatrix}, \dots, C_p = \begin{bmatrix} c_{1p} \\ c_{2p} \\ \vdots \\ c_{np} \end{bmatrix}.$$

The weights corresponding to the set of decision variables can be represented by the matrix W_{C_j} ,

$W_{C_j} = [w_{c_1} \ w_{c_2} \ \dots \ w_{c_p}]$. The matrix of normalized weights $Adj_W_{C_j}$, is then given by,

$$Adj_W_{C_j} = \begin{bmatrix} \frac{w_{c_1}}{\sum_{j=1}^p w_{c_j}} & \frac{w_{c_2}}{\sum_{j=1}^p w_{c_j}} & \dots & \frac{w_{c_p}}{\sum_{j=1}^p w_{c_j}} \end{bmatrix}$$

The tolerance levels can be summarized in the form of matrix,

$$\alpha = \begin{bmatrix} \max_{j=1}^p (w_{c_j}) & \text{Initial value} \\ \alpha - \tau & \vdots \\ \alpha - 2\tau & \vdots \\ \vdots & \vdots \\ \alpha - t\tau & \text{(Intermediate values)} \\ \vdots & \vdots \\ 0.1 & \vdots \\ 0 & \text{Ideal value} \end{bmatrix}$$

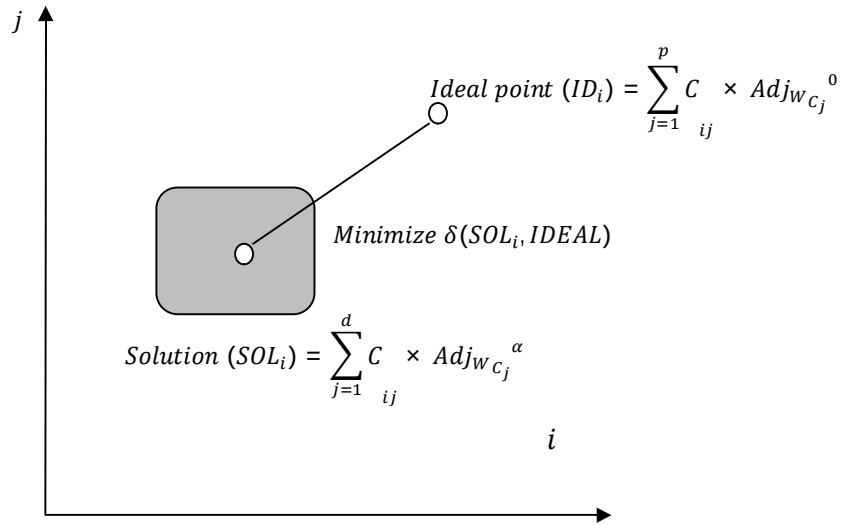


Figure 6.4: PBO Approach

The IDEAL solution with $\alpha = 0$ can hence be defined by the matrix ID_i ,

$$ID_i = \sum_{j=1}^p C_{ij} \times Adj_{w_{c_j}}^0 = \begin{bmatrix} c_{11} \cdot \frac{w_{c_1}}{\sum_{j=1}^p w_{c_j}} + \dots + c_{1p} \cdot \frac{w_{c_p}}{\sum_{j=1}^p w_{c_j}} \\ \vdots \\ c_{n1} \cdot \frac{w_{c_1}}{\sum_{j=1}^p w_{c_j}} + \dots + c_{np} \cdot \frac{w_{c_p}}{\sum_{j=1}^p w_{c_j}} \end{bmatrix}$$

The threshold level (T) for finding an optimal solution can be set as below where a is the desired accuracy percentage:

$$T = ID_i \times (1 \pm (1 - a)) \quad (6.23)$$

The INITIAL solution with $\alpha = \max_{j=1}^p (w_{c_j})$ can be expressed as the matrix $INIT_i$,

$$INIT_i = C_i \times Adj_{w_{c_1}}^{\max_{j=1}^p (w_{c_j})} = \begin{bmatrix} c_{11} \cdot 1 \\ \vdots \\ c_{n1} \cdot 1 \end{bmatrix}$$

The SOL_i value of the candidate solutions in a given iteration t can be expressed as given below.

The tolerance level will be taken as $\alpha - t \cdot \tau$ such that the no. of decision variables out of $j = 1, 2, \dots, p$ for which $w_{c_i} \geq \alpha - t \cdot \tau$ is taken as d . Then, the solution with $\alpha = \alpha_{init} - t \cdot \tau = \max_{j=1}^p (w_{c_j}) - t \cdot \tau$, can be derived as below:

$$SOL_i = \sum_{j=1}^d C_{ij} \times Adj_{w_{c_j}}^\alpha = \begin{bmatrix} c_{11} \cdot \frac{w_{c_1}}{\sum_{j=1}^d w_{c_j}} + \dots + c_{1d} \cdot \frac{w_{c_d}}{\sum_{j=1}^d w_{c_j}} \\ \vdots \\ c_{n1} \cdot \frac{w_{c_1}}{\sum_{j=1}^d w_{c_j}} + \dots + c_{nd} \cdot \frac{w_{c_d}}{\sum_{j=1}^d w_{c_j}} \end{bmatrix} \quad (6.24)$$

From (6.24), the PMI value of the candidate solutions in a given iteration t can be expressed as given below:

$$\begin{aligned}
PMI &= \frac{|SOL - T|}{INIT} = \frac{\sum_{j=1}^d C_{ij} \times Adj_{w_{c_j}}^\alpha - \sum_{j=1}^p C_{ij} \times Adj_{w_{c_j}}^0 \times (1 \pm (1 - a))}{C_{i1} \times Adj_{w_{c_1}}^{\max_{j=1}^p(w_{c_j})}} \\
&= \left| \frac{(c_{11} \cdot w_{c_1} + \dots + c_{1d} \cdot w_{c_d}) \times \left(\frac{1}{\sum_{j=1}^d w_{c_j}} - \frac{1 \pm (1 - a)}{\sum_{j=1}^p w_{c_j}} \right)}{c_{11} \cdot 1} \right. \\
&\quad \left. \frac{(c_{n1} \cdot w_{c_1} + \dots + c_{nd} \cdot w_{c_d}) \times \left(\frac{1}{\sum_{j=1}^d w_{c_j}} - \frac{1 \pm (1 - a)}{\sum_{j=1}^p w_{c_j}} \right)}{c_{n1} \cdot 1} \right|
\end{aligned} \tag{6.25}$$

The aim is to minimize the objective function such that the plate mobility index becomes zero. The PMI matrix can be set as below:

$$\text{Minimize } \delta(SOL_i, IDEAL)$$

such that,

$$PMI = \frac{SOL - T}{INIT} \leq 0$$

(≥ 0 in case of a maximization problem)

where,

$T = IDEAL \times (1 \pm (1 - a))$ and a is the desired accuracy / precision.

6.2.4. Differences between PBO and the other EAs

In this section, some of the differences of PBO are pointed out. Unlike the other evolutionary optimization algorithms like BBO [(Dawei, D., et. al., 2009), (Simon, D., 2008), (Simon, D., 2011)], ACO [(Blum, C., 2005), (Dorigo, M., et. al., 2004)], PSO [(Clerc, M., 2006), (Eberhart, R., et. al., 2004)], DE [(Price, K., et. al., 1997), (Storn, R., 1999)], ES [(Kim, J. H., et. al., 1997), (Parmee, I., 2001), (Yao, X., et. al., 1999)], PBIL [Zhengra, H., et. al., 1999], GA [(Goldberg, D., 1989), (Michalewicz, Z., 1992)], Stud-GA [Khatib, W., et. al., 1998], etc. categorized as population based EAs [(Back, T., 1996), (Dasgupta, D., et. al., 2001), (Solomon, R., 1997)], PBO is not a population based EA since it does not involve reproduction or generation of children and can be categorized as a nature inspired or a geo-sciences based EA in the taxonomy of evolutionary optimization techniques since PBO is inspired from the geo-sciences phenomenon of plate tectonics which has never been used as a contributory domain to evolutionary computation. PBO is in a way similar to ACO [(Blum, C., 2005), (Dorigo, M., et. al., 2004)] since it assigns a new set of decision variables as well as the weights to these decision variables based on the changes in their relative importance with respect to each other and hence generates a new set of solutions at each iteration as opposed to BBO [(Dawei, D., et. al., 2009), (Simon, D., 2008), (Simon, D., 2011)] in which the set of solutions are maintained from one iteration to the next, relying on migration to probabilistically adapt those solutions. However, PBO differs from

ACO in the sense that it is capable of incorporating lower prioritized decision variables in dynamic environments and achieve a better solution with each iteration thus leading to an iterative enhancement in the optimization efficiency achieved as discussed in the next section on the performance analysis of PBO. PBO can be contrasted with BBO in that BBO solutions are changed directly via migration from other solutions (islands). That is, BBO solutions directly share their attributes (SIVs) with other solutions however, in PBO, there is no sharing of attributes involved. The PBO solutions change with respect to the decision variables and the weights assigned to the decision variables that compose the solution. Also, unlike BBO, in which the old solutions are improved via migration, in PBO, the solutions are regenerated at each iteration by changing the set of decision variables (incorporating the lower prioritized decision variables) composing them and the weights are adjusted with respect to each other in order to reflect the effect of inclusion of additional variables on the optimization process since with this inclusion, there may be a tradeoff between the several decision variables and their weights used for establishing the goodness of the solution indicated by the PMI value.

PBO is similar to PSO [(Clerc, M., 2006), (Eberhart, R., et. al., 2004)] in the sense that PSO also models the change in its solution by representing each solution as a point in space, and representing the change over time of each solution as a velocity vector. In PBO, the weights assigned to the solutions are altered and lower prioritized decision variables that compose a solution are incorporated iteratively and the solutions are not changed directly. This is similar to PSO in which the solutions change indirectly via their velocities which in turn results in the position changes. PBO also differs from DE [(Price, K., et. al, 1997), (Storn, R., 1999)] in the sense that DE changes its solutions directly, but changes in a particular DE solution are based on differences between other DE solutions. The above discussion is summarized in Table 6.3. It is these differences between PBO and the other EAs that may prove to be its strength and provide the reason why PBO performs better than the other EAs on specific problems.

6.3. Performance Analysis and Results

This section looks at the performance of the proposed PBO as compared with the other evolutionary optimization methods by comparing their performances on a set of commonly used benchmark functions [(Finck, S. et. al., 2009), (Hansen, N., et. al., 2009), (Hansen, N., et. al., 2012), (Posik, P., et. al, 2012)]. The section also tests the proposed PBO on the Travelling Salesman problem (TSP) on a 5-cityset and extends the demonstration to a 50-city dataset. The section compares the performance of PBO with the ten other recent evolutionary optimization techniques for the TSP problem.

6.3.1. Performance analysis on benchmark functions using the minimal cost estimate

This section tests the PBO performance on some standard benchmark functions. It compares the performance of PBO with the other recent evolutionary optimization techniques such as BBO

[(Dawei, D., et. al., 2009), (Simon, D., 2008), (Simon, D., 2011)], ACO [(Blum, C., 2005), (Dorigo, M., et. al., 2004)], PSO [(Clerc, M., 2006), (Eberhart, R., et. al., 2004)], DE [(Price, K., et. al., 1997), (Storn, R., 1999)], ES [(Kim, J. H., et. al., 1997), (Parmee, I., 2001), (Yao, X., et. al., 1999)], PBIL [Zhengra, H., et. al., 1999], GA [(Goldberg, D., 1989), (Michalewicz, Z., 1992)], Stud-GA [Khatib, W., et. al., 1998] on the set of 14 standard benchmark functions in order to demonstrate the performance of the proposed PBO. The code that is modified and integrated with PBO for the performance analysis is taken from the work by Simon [(Dawei, D., et. al., 2009), (Simon, D., 2008), (Simon, D., 2011)] in order to demonstrate the performance of PBO on the same 14 benchmark functions as Simon [(Dawei, D., et. al., 2009), (Simon, D., 2011)]. The range and the granularity of each dimension are same as in [(Dawei, D., et. al., 2009), (Simon, D., 2008)].

The parameters that were used are: For PBO, the acceleration due to gravity in m/s^2 was taken as $g = 9.8$, the atmospheric temperature in Kelvin was taken as $T_0 = 298$, the surface area of the plate under consideration $A = 100$, the thickness of the plate was assumed to be

Table 6.3: Summary of differences between PBO and the other contemporary evolutionary optimization techniques

	PBO	BBO	ACO	PSO	DE	PBIL	GA	Stud-GA
Evolutionary Computation	Geo-sciences based	Population based	Population based	Population based	Population based	Population based	Population based	Population based
Optimization environment	Dynamic.	Static	Static.	Static.	Dynamic.	Static.	Static..	Static.
Solution generation	New solution generation (by changes in decision variables and their weights).	Probabilistic adaptation of the old solution by migration	New solution generation (based on pheromone update)	Old solution improved by information sharing by changing velocity and position.	Old solution is probabilistically adapted based on differences between the other solutions	New solution generation based on the probability distribution .	New solutions generated by global recombination by natural selection (reproduction)	New solutions are generated by global recombination (reproduction by crossover).
Biologically motivated	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Solutions change process	Solutions do not change directly (change via associated weights)	Solutions change directly via migration.	Solutions change directly based on the pheromone update.	Solutions do not change directly (change via velocities and position)	Solutions change directly based on differences b/w other DE solutions.	Solutions change directly by allele values from PBILs probability distribution	Solutions change directly via recombination and crossover.	Solutions change directly via crossover.
Sharing of attributes	No	Yes	No	Yes	Yes	No	Yes	Yes
Iterative improvement	By incorporating lower prioritized variables with normalized weights iteratively	By using migration to change existing sol ⁿ and elitism to retain the best solutions (in a static environment).	By generating new solutions using the concept of pheromone deposition of ants. (in a static environment).	By information sharing resulting in old solution improvement. (in a static environment).	By probabilistically adapting the old solutions using the differences between other DE solutions (in a static environment).	By updating probability distribution of solutions based on their learning rates, fixed / variable (in a static environment).	By global recombination and reproduction (in a static environment).	By global recombination and crossover (in a static environment).
Computational effort	Lower since doesn't create infeasible solutions	Higher due to long waiting period for feasibility check.	Higher due to long waiting period for feasibility check.	Higher due to long waiting period for feasibility check.	Higher due to long waiting period for feasibility check.	Higher due to long waiting period for feasibility check.	Higher due to long waiting period for feasibility check.	Higher due to long waiting period for feasibility check.

constant for all plates and was taken as $t = 2000$ for all plates, the depth of ocean at location of the plate in meters was assumed to be 10 km i.e. $y_0 = 10000$; the temperature in Kelvin beyond which ridge push will occur was taken as $T_{critical}(T_c) = 2000$, the cooling coefficient (β) = 0.000001 (the unit is $/K.cm^3$) and the time constant for slab pull plate cooling $\tau = 10$. For ACO, the initial pheromone value was taken as $1E - 6$, pheromone update constant as 20, global pheromone decay rate as 0.9, exploration constant as 1, local pheromone decay rate as 0.5, pheromone sensitivity as 1, and visibility sensitivity as 5 [(Blum, C., 2005), (Dorigo, M., et. al., 2004)]. For BBO, the habitat modification probability = 1, immigration probability bounds per gene = [0,1], step size for numerical integration of probabilities = 1, maximum immigration and migration rates for each island = 1, and mutation probability = 0 [(Dawei, D., et. al., 2009), (Simon, D., 2008), (Simon, D., 2011)]. For DE, a weighting factor of 0.5 and crossover constant of 0.5 was used [(Price, K., et. al, 1997), (Storn, R., 1999)]. For the ES, $\lambda = 10$ offspring in each generation were produced, and standard deviation was taken as $\sigma = 1$ for changing solutions [(Finck, S. et. al., 2009), (Parmee, I., 2001), (Yao, X., et. al., 1999)]. For the GA, a single point crossover with a crossover probability of 1, and a mutation probability of 0.01 was used [(Goldberg, D., 1989), (Michalewicz, Z., 1992)]. For PBIL, a learning rate of 0.05, 1 good population member and 0 bad population members, an elitism parameter of 1, and a 0 probability vector mutation rate was used [Zhengra, H., et. al., 1999]. For SGA, a single point crossover with a crossover probability of 1, and a mutation probability of 0.01 was used [Khatib, W., et. al., 1998]. For PSO, an inertial constant = 0.3, a cognitive constant = 1, and a social constant for swarm interaction = 1 was used [(Clerc, M., 2006), (Eberhart, R., et. al., 2004)]. As in [(Dawei, D., et. al., 2009), (Simon, D., 2008), (Simon, D., 2011)], each algorithm is considered to have a population size of 50, an elitism parameter of 2, and ran for 50 generations.

Tables 6.4 and 6.5 show the performance analysis results. Table 6.4 and 6.5 shows the average minima and the absolute best minima and also the CPU times found by each algorithm over 50 Monte Carlo simulations on 14 standard benchmark functions. From Table 6.4, it is seen that PBO performed the best (on average) on nine of the 14 benchmarks followed by BBO and ACO both of which performed the best on two of the standard benchmark functions each and PBIL which perform best on one benchmark function. From the table, it is observed that PBO outperforms DE, GA and Stud-GA on all the benchmark functions which is evident from the fact that the average and the absolute minimal cost found by PBO is much lower than that found by DE, GA and Stud-GA for all the 14 benchmark functions. From Table 6.4, it is observed that the CPU time for PBO is the lowest among all the optimization algorithms (CPU time = 0.31) which reflects the fact that the computational

complexity of PBO is the lowest despite of the significant decrease in the cost that it has provided over the other evolutionary optimization algorithms which require CPU times of 0.34, 0.39, 0.41, 0.42, 0.45, 0.47, 0.5 and 1.05 for PBIL, Stud-GA, ES, GA, BBO, PSO, DE and ACO respectively.

This can be explained by the fact that with the incorporation of the concept of reduction in tolerance level, the inclusion of additional constraints in the optimization process that may arise during the optimization process itself is made, and an optimal set of decision variables which can guarantee an optimal solution is found. Also, using the concept of dynamic adjustment of weights of these variables associated with a solution in each iteration, PBO is capable of incorporating lower prioritized criteria in dynamic environments and thus leading to an iterative enhancement in the optimization efficiency achieved. This is indicated by the decreasing value of the plate mobility index and hence, convergence to an optimal solution value is faster in the proposed model. This can also be reflected from the graph plots given in figures 6.5(a)-(n). The conceptual significance of this adjustment of weights for the newly created set of decision variables in each iteration is the distribution of weights based on the relative importance of the variables with respect to each other. The other nature-inspired optimization algorithms do not incorporate this concept of relative adjustment of weights used for measuring the goodness of a solution and hence may not provide optimum solution in dynamic environment ('dynamic' refers to the changing set of decision variables in each iteration) since they apply the same fitness function for a solution regardless of the change in the decision variables and the associated weights used for calculating the fitness function for minimal value found by the plate tectonics based model and hence faster iterative convergence to the minimal value. This can also be verified from the graph plots given in the figures 6.5(a)-(n) from which it is clearly reflected that the graph of PBO achieves the lowest minima and the candidate solution at each iteration. The above phenomenon leads to the decrease in the average value for almost all of the benchmark functions. From the above discussion and results, it can be concluded that PBO far outperforms the standard BBO and the other optimization algorithms as well and hence proposes an advanced optimization algorithm.

Based on the above characteristic strength of PBO, the applicability of PBO is demonstrated on engineering applications also. For the purpose, PBO is applied on the travelling salesman problem on a 5-cityset. From the results, it is observed that the cost achieved by PBO is the lowest among all the other evolutionary optimization algorithms for all the city sets in case of the TSP problem. Based on the results, it is observed that PBO far outperforms all the other evolutionary optimization algorithms and is capable of leading to an optimum solution faster and with minimum consumption of resources (i.e. the decision variables).

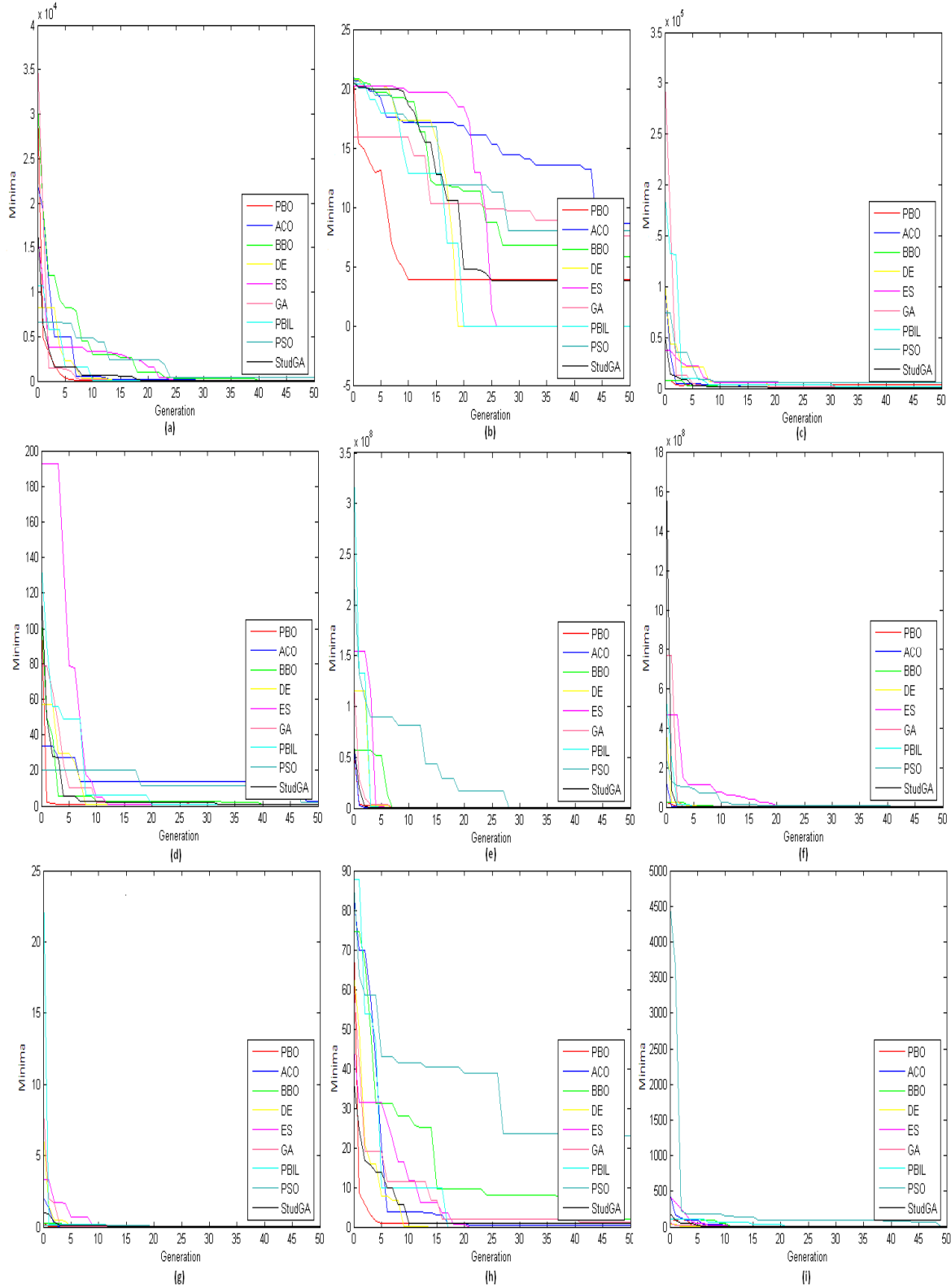


Figure 6.5(a)-(i): Graph plot of the minimal values for 50 generations for each of the nine optimization algorithms on the set of 14 benchmark functions. The benchmark functions are : Figure (a) – Step, Figure (b) - Ackley, Figure (c) - Fletcher, Figure (d) – Griewank, Figure (e) - Penalty #1, Figure (f) - Penalty #2, Figure (g) - Quartic, Figure (h) - Rastrigin, Figure (i) – Rosenbrock.

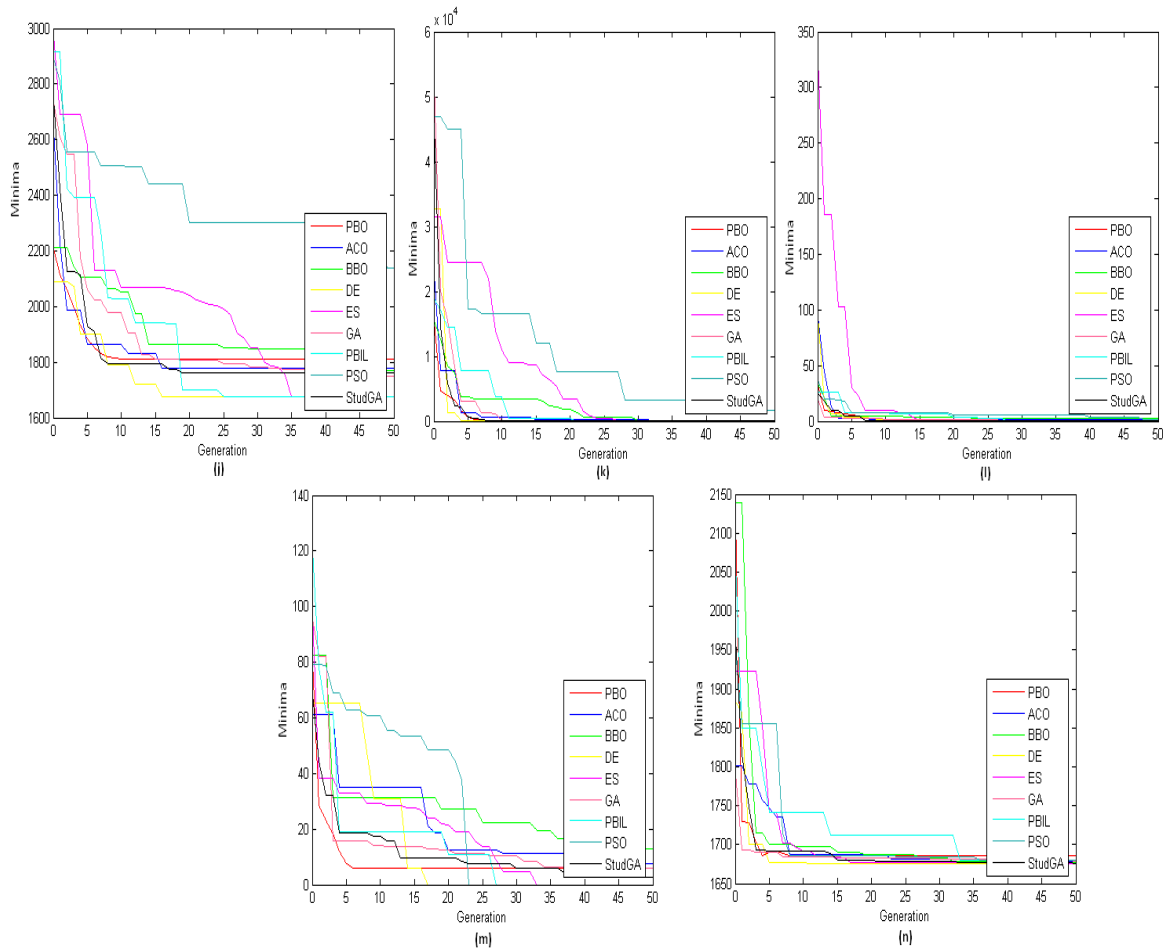


Figure 6.5(continued) (j) – (n): Graph plot of the minimal values for 50 generations for each of the nine optimization algorithms on the set of 14 benchmark functions. The benchmark functions are: Figure (j) - Schwefel 1.2, Figure (k) - Schwefel 2.21, Figure (l) - Schwefel 2.24, Figure (m) - Schwefel 2.26 and Figure (n) - Sphere.

Table 6.4: Mean normalized optimization results and CPU times on 14 benchmark functions. The numbers shown are average minimum function values found by the algorithms over 50 Monte Carlo simulations, and normalized so that the smallest number in each row is 100.

	PBO	BBO	ACO	DE	ES	GA	PBIL	STUDGA	PSO
Step	100	424	267	552	212	345	181	764	889
Ackley	100	290	121	174	212	107	172	233	211
Fletcher	218	116	319	455	283	448	100	515	102
Griewank	100	1405	550	1942	620	1219	372	1309	419
Penalty #1	241	100	5040	8886	1107	5808	459	3.07E+04	5990
Penalty #2	8.11E5	100	3.97E6	1.11E8	4.04E7	1.43E7	4.18E6	5.65E7	6.25E6
Quartic	100	230	130	115	700	145	400	410	455
Rastrigin	100	550	231	517	503	632	263	2949	1367
Rosenbrock	362	584	100	788	215	1031	238	6345	838
Schwefel 1.2	100	101	101	101	101	102	102	101	101
Schwefel 2.21	100	264	241	2043	390	661	181	3574	595
Schwefel 2.24	170	246	100	1460	120	443	118	695	366
Schwefel 2.26	100	279	186	199	181	160	139	349	354
Sphere	100	1613	345	1817	1256	1204	216	3084	770
CPU Times	0.31	0.45	1.05	0.5	0.41	0.42	0.34	0.39	0.47

Table 6.5: Best normalized optimization results on 14 benchmark functions. The numbers shown are the absolute best minima found after 50 Monte Carlo simulations of each algorithm, and normalized so that the smallest number in each row is 100.

	PBO	BBO	ACO	DE	ES	GA	PBIL	STUDGA	PSO
Step	100	112	459	102	139	133	184	722	208
Ackley	103	227	180	244	199	183	100	212	153
Fletcher	279	249	461	393	305	513	100	384	130
Griewank	100	280	1411	493	190	640	190	212	550
Penalty #1	148	100	585	348	396	1293	215	439	433
Penalty #2	100	180	405	3.34E8	1.91E8	5.26E8	1.77E6	2.44E7	2.05E6
Quartic	100	4504	1.75E4	2189	482	8031	3333	4070	445
Rastrigin	100	125	812	420	150	1196	125	2770	225
Rosenbrock	241	752	100	186	198	252	257	395	561
Schwefel 1.2	100	103	101	103	101	107	102	102	110
Schwefel 2.21	100	1879	1653	1247	218	1.18E4	173	3.97E4	592
Schwefel 2.24	100	226	145	450	238	338	575	436	300
Schwefel 2.26	244	448	244	349	236	432	100	1506	512
Sphere	191	625	100	591	140	1.60E4	10	1449	171

Thus, it can be concluded that the PBO algorithm finds major use for path planning where an iterative improvement of the solution is observed which is the characteristic property of PBO and proves to be its strength as discussed in the next section.

6.3.2. Plate Tectonics Based Optimization for Travelling Salesman Problem

This section demonstrates the performance of PBO on the travelling salesman problem (TSP). Figure 6.6 presents the algorithm for solving the TSP using PBO.

In the case of TSP, the set of decision variables (j where j ranges from 1 to 5) for optimization are defined as the distance between the two cities (D), the mobility factor of the path (MF), the transportation network availability on the path (N) and the terrain characteristics specifically the degree of water (W) and vegetation (V) on the path for deciding the feasibility of a path for travelling by the salesman. The corresponding weights attached to each of these decision variables are taken as $w_{c_j} = \{0.6, 0.5, 0.3, 0.2, 0.1\}$ based on their relative importance in deciding the best feasible path. The S value is calculated for each from the current city in order to decide the best path from the current city until the initial city is reached. At this stage, the PMI value is calculated by subtracting the threshold value from this and dividing it by the initial value. The initial value is the value of the factor S with only the highest prioritized decision variable (i.e. distance) in the decision variables set whose weight is assigned as 1. The initial value of the tolerance level is set to $\alpha_{init} = \max_{j=1}^p (w_{c_j}) = 0.6$, $\tau = 0.1$. The desired accuracy is taken as 95%. The threshold level is set with respect to the ideal solution on a 5-city dataset (figure 6.7) as given below where the numbers in brackets represent the values of the decision variables. (The values of all the decision variables are considered as percentage of the maximum value of the distance

-
1. Form an initial set of candidate paths from the given initial city. Choose it as the current city.
 2. Initialize the decision variables set c_j i.e. $c_j = \{\text{distance (D)}, \text{mobility factor (M}_t\text{)}, \text{transportation network availability (N)}, \text{terrain activities – degree of water (W)}, \text{terrain activities - degree of vegetation (V)}\}$; $w_{c_j} = \{0.6, 0.5, 0.3, 0.2, 0.1\}$
 3. Calculate the factor S (initialize $S = 0$) for each path from the current city,

$$S = S + \sum_{j=1}^{p=5} w_{c_j} \cdot c_j^\alpha$$

where $c_j^\alpha \rightarrow c_j$ for which $w_{c_j} \geq \alpha$. Initially $\alpha_{init} = \max_{j=1}^p (w_{c_j}) = 0.6$, $\tau = 0.1$.

4. Choose the path with the lowest S value and assign the next node of this path as the current city. Repeat the steps (3) - (4) until the initial city is reached. The PMI value is defined as $PMI = \frac{|S-T|}{w_{c_j}^{\alpha_{init}} \cdot c_j^{\alpha_{init}}}$ where $w_{c_j}^{\alpha_{init}} = 1$ and T is the predefined threshold value.
 5. Reduce the tolerance level α by τ and hence, formulate, c_j^α i.e. $\alpha = \alpha - \tau$;
 6. Alter the weights corresponding to the new set of decision variables by replacing $w_{c_j}^\alpha$ by $\frac{w_{c_j}^\alpha}{\sum_{j=1}^p (w_{c_j}^\alpha)}$ for each c_j^α such that $\sum_{j=1}^p (w_{c_j}^\alpha) = 1$ and the next iteration is performed with these altered weights.
 7. Repeat the steps until PMI becomes zero at which point the corresponding path is the best path found and the PMI is the optimal PMI value.
-

Figure 6.6: TSP problem solving by PBO.

variable and hence the values of the other decision variables are calculated by multiplying each by the maximum value of the decision variable, distance i.e. 190).

The ideal solution has the lowest possible tolerance level i.e. 0.1 since it considers all the 5 decision variables with their normalized weights obtained as $\{0.35, 0.29, 0.18, 0.12, 0.06\}$. Using the above set of ideal weights, the best path predicted by PBO will be ABCEDA with the S value as 339.70 and hence the threshold value can be set as $340 \times (1 \pm (1 - 0.95)) = 357$. In each iteration, the best path predicted by PBO and its corresponding PMI value is refined by reducing the tolerance level and hence incorporating lower prioritized decision

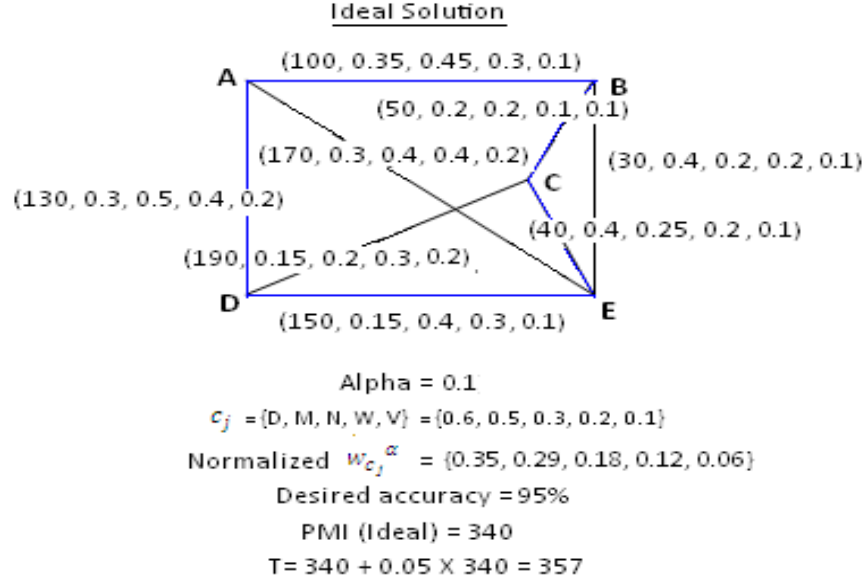
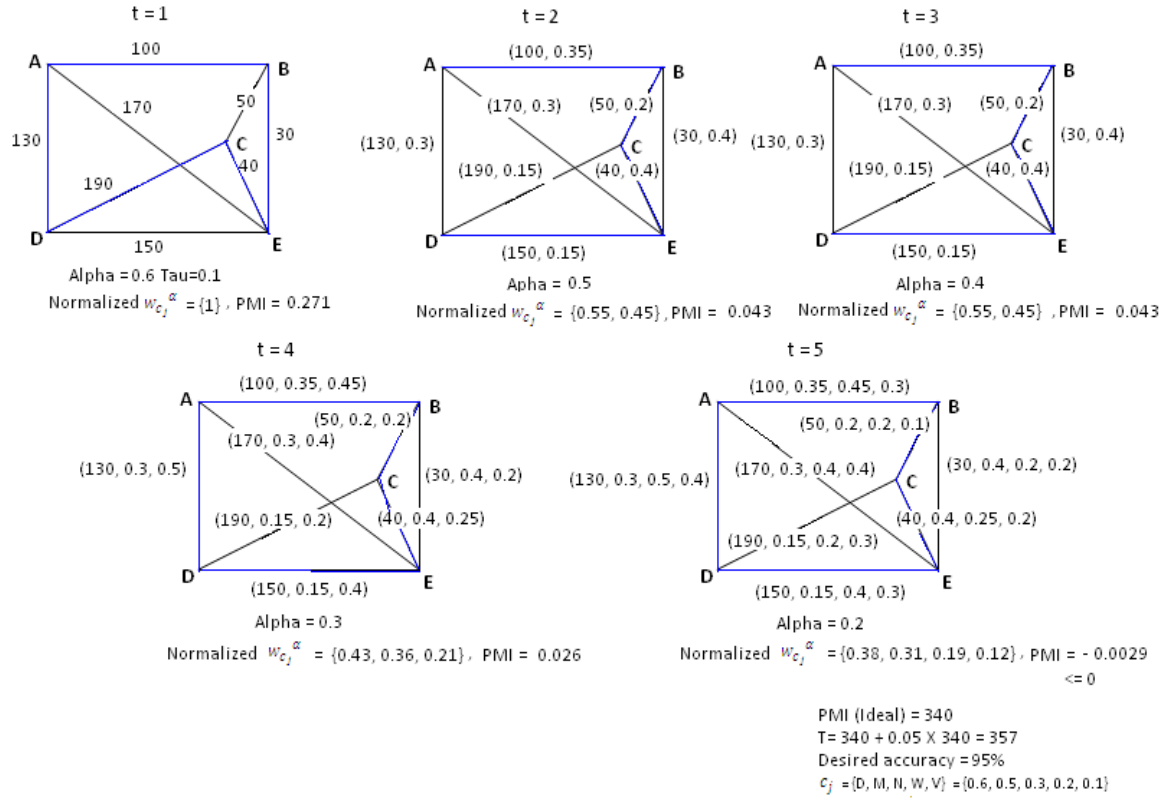


Figure 6.7: Ideal solution for TSP problem solving through PBO on a sample 5-city dataset

variables after adjustment of their weights dynamically through normalization until the PMI value reaches zero at which point the optimal solution is reached. The output of the algorithm also mentions the best path predicted by PBO along with the tolerance level indicating the optimal set of decision variables required for predicting the optimal path.

The graphs in figures 6.8(a)-(e) represent five iterations of the TSP problem solving on a 5-city dataset where the numbers in brackets represent the values of the decision variables considered in the current iteration. Figure 6.8(a) represents the initial stage when only the highest weighted decision variable (distance) is considered for PMI calculation (the tolerance level being 0.6 initially), the weight assigned to it is thus 1. Hence, A being the initial city, the S values of the paths AB, AE and AD are 100, 170 and 130 respectively, and hence the next city to be traversed is B which is set as the current city. Proceeding in the similar way, the best path predicted by PBO in the first iteration is $A \rightarrow B \rightarrow E \rightarrow C \rightarrow D \rightarrow A$ with $S = (100 + 30 + 40 + 190 + 130) = 490$ and the PMI value as $(490 - 357) / 490 = 0.270$. Since, the PMI value is not zero, the iterative refinement process is started by reducing the tolerance level by 0.1. Now, the tolerance level becomes 0.5 and hence, the $c_j^\alpha = \{D, M_F\}$ with the normalized adjusted weights as $\left\{\frac{0.6}{1.1}, \frac{0.5}{1.1}\right\} = \{0.55, 0.45\}$ as shown by figure 6.8(b). Using the above set of weights for the currently considered variables, the best path predicted by PBO changes to $A \rightarrow B \rightarrow C \rightarrow E \rightarrow D \rightarrow A$ the best path predicted by PBO remains the same but the S value reduces further i.e.



Figures 6.8(a)-(e): Solution graph for TSP problem solving by PBO for 5 iterations (t = 1 to 5). The figures in brackets are the values of the decision variables, {Distance, Mobility factor, Network, degree of water, degree of vegetation} for the given path between the two cities X and Y. The best predicted path by PBO at each iteration is shown in blue and is **ABECDA** → **ABCEDA** → **ABCEDA** → **ABCEDA** → **ABCEDA** with the PMI values as **0.270** → **0.043** → **0.043** → **0.026** → **-0.0029** (≤ 0).

$S = (84.925 + 44.6 + 56.2 + 95.325 + 97.15) = 378.2$ with the PMI value as $(378.2 - 357) / 490 = 0.043$.

In the next iteration, reducing the tolerance level to 0.4 does not change the set $c_j^\alpha = \{D, M_F\}$ and hence the PMI remains constant as depicted by figure 6.8(c). On further reducing the tolerance level to 0.3, the network availability variable is also considered i.e. $c_j^\alpha = \{D, M_F, N\}$ and the adjusted weights become $\left\{\frac{0.6}{1.4}, \frac{0.5}{1.4}, \frac{0.3}{1.4}\right\} = \{0.43, 0.36, 0.21\}$ based on which $S = (84.895 + 43.16 + 54.54 + 90.72 + 96.37) = 369.7$ and the $PMI = \frac{369.7 - 357}{490} = 0.026$. Further reducing the tolerance level to 0.2 makes $c_j^\alpha = \{D, M_F, N, W\}$ with adjusted weights as $\left\{\frac{0.6}{1.6}, \frac{0.5}{1.6}, \frac{0.3}{1.6}, \frac{0.2}{1.6}\right\} = \{0.38, 0.31, 0.19, 0.12\}$ and with $S = (81.7 + 40.28 + 52.345 + 87.12 + 94.24) = 355.7$ which makes $PMI \leq 0$ (minimization function). At this stage, the optimal solution is reached and the algorithm outputs the best path as ABCEDA with the tolerance level as 0.2 (optimal decision variables set is $\{D, M_F, N, W\}$). This means that the proposed PBO algorithm also considers the optimal set of decision variables that are sufficient to guarantee an optimal solution and hence reduces the computational complexity of the solution.

Table 6.6: Summarizing the values of PMI and the best path predicted by the PBO algorithm in each

		t = 1	t = 2	t = 3	t = 4	t = 5
$T = 357$ <i>Desired accuracy</i> $= 95\%$ $\alpha_{init} = 0.6, \tau = 0.1$ <i>PMI (ideal)</i> $= 340$. C_j $= \{D, M_F, N, W, V\}$ w_{c_j} $= \{0.6, 0.5, 0.3, 0.2, 0\}$	α	0.6	0.5	0.4	0.3	0.2
	$\frac{w_{c_j}^\alpha}{\sum_{j=1}^p (w_{c_j}^\alpha)}$	{1}	{0.55, 0.45}	{0.55, 0.45}	{0.43, 0.36, 0.21}	{0.38, 0.31, 0.19, 0.12}
	PMI $= \frac{ S - T }{w_{c_j}^{\alpha_{init}} \cdot c_j^{\alpha_{init}}}$	$\frac{490 - 357}{490}$ $= 0.270$	$\frac{378.2 - 357}{490}$ $= 0.043$	$\frac{378.2 - 357}{490}$ $= 0.043$	$\frac{368.7 - 357}{490}$ $= 0.026$	$\frac{355.7 - 357}{490}$ $= -0.0029$ (< 0)
	<i>Best path</i>	ABECDA	ABCEDA	ABCEDA	ABCEDA	ABCEDA

Table 6.6 summarizes the values of the tolerance level, the decision variables of the current iteration, their normalized weights, the S and PMI values along with the initial parameters. From the table, it is observed that with the increasing iterations, the PMI values decrease and the solutions refine until the optimal solution is reached. Comparing the optimal PMI value with the initial value, there is an improvement of almost 28% in the efficiency of the predicted solution. This iterative enhancement is not possible with other popular techniques used for TSP problem solving like ACO which will find the best path in a static environment since these do not consider the lower prioritized variables and also the concept of dynamic adjustment of weights in the calculation of the cost for the choice of the best path is not made. Using ACO, the best path chosen in the first time instant will remain the same and will be the output of ACO (referring to figure 6.8(a), the output of ACO will be 490 and the best path chosen will be ABECDA. The above discussion was done on a 5-city dataset just for demonstration of the working mechanism of PBO. Next, the proposed algorithm is tested on standard TSP problem instances and the results are compared with the other approaches of Hybrid GA with GCX [(Whitley, D., et. al., 2010), (Zhifeng, H., et. al., 2008)], Chained-LK [(Whitley, D., et. al., 2010), (Zhifeng, H., et. al., 2008)], ACO [(Blum, C., 2005), (Dorigo, M., et. al., 2004)], SGA [Khatib, W., et. al., 1998], ES [(Back, T., 1996), (Afaq, H. et. al., 2011)], SOFM [Afaq, H. et. al., 2011], FI [Afaq, H. et. al., 2011], PSO [(Clerc, M., 2006), (Eberhart, R., et. al., 2004), (Afaq, H. et. al., 2011)], GA [(Goldberg, D., 1989), (Michalewicz, Z., 1992)], BBO [(Simon, D., 2008), (Song, Y., et. al., 2010)], PBIL [(Afaq, H. et. al., 2011), (Zhengra, H., et. al., 1999)] and DE [(Sauer, J. G., et. al., 2008), (Mei, M., et. al., 2010), (Storn, R., 1999)] that have been used till date for TSP problem solving.

Table 6.7 presents the results of the simulation for symmetric TSP problem solving on standard TSP problem instances where lower prioritized factors are also introduced in the cost factor calculation after adjustment of weights. From the table, it is observed that the cost factor achieved by PBO is the lowest among all the other evolutionary optimization

Table 6.7: Comparison of PBO with other nature-inspired algorithms on random instances of the symmetric TSP. Comparison on the cost factor ($\times 10^2$) obtained on standard problem instances.

Problem Name	PBO	Hybrid GA-GCX	Chained-LK	ACO	SGA	ES	SOFM	FI	PSO	GA	BBO	PBIL	DE
rand500	48.7	51.135	53.083	58.6	58.8	59.8	60.6	60.3	56.7	59.8	56.2	57.4	57.8
att532	53.52	57.27	57.8	64.37	63.95	64.16	66.5	66.8	62.24	64.27	60.44	62.98	63.52
nrw1379	127.7	157.07	151.96	153.62	155.83	157.21	160.80	161.34	150.87	157.48	148.11	150.59	151.41
rand1500	142.2	173.48	177.75	171	174.3	175.8	176.1	178.8	165.3	175.8	165.3	167.40	168.6
u1817	186.42	303.87	341.15	224.22	230.03	235.85	243.48	243.84	222.40	235.85	222.04	219.49	221.31

algorithms for all the datasets. Thus, from the results, it is reflected that PBO far outperforms the other evolutionary optimization algorithms that have been used till date for TSP problem solving and is capable of leading to an optimum solution faster. Unlike the other approaches which considered only the highest weighted variable i.e. distance and assigned it as the weight 1, PBO considers a tradeoff between the higher and the lower priority decision variables by distributing the weights among the variables according to the relative importance of a decision variable in contributing towards the choice of the best solution which is achieved through normalization of weights such that the total is always 1. It is to be noted that even if the complete set of variables is considered by other algorithms, the variables must be considered relative to each other which would require normalization of weights and not simply picking the given weight value of a decision variable independently.

6.4. Observations

The chapter has shown how plate tectonics, the study of the large scale motions of continental plates and the driving forces leading to the plate motion can be used to derive algorithms for optimization. This new family of algorithms is called as PBO. PBO was also applied to benchmark functions and to the TSP, and shown that it provides performance on a par with the other evolutionary-optimization methods. It was noted that CPU time is critical to the implementation of many population-based evolutionary optimization algorithms. If an algorithm does not converge rapidly, it will be of little use since the optimality condition would not be satisfied within the prescribed time limit. From the performance analysis results of PBO, it is observed that PBO is the fastest among all the optimization algorithms analyzed and hence PBO is the least computationally complex algorithm among all the other evolutionary optimization algorithms compared in this chapter (see Table 6.3).

It cannot be concluded that PBO is universally better than other algorithms, or vice versa. However, it may be possible in future work to improve and quantify the performance of PBO relative to other algorithms for applications with specific features. The good performance of PBO on TSP provides some evidence that PBO theory can be successfully applied to

practical problems especially when working in constantly changing environments where PBO iteratively enhances the solution as a continuous optimizer. Owing to the above characteristic property of PBO, PBO also removes the disadvantage of the population based optimization algorithms, one that is related to computational effort, i.e. the problem of creating infeasible solutions. In BBO as well as in the other evolutionary optimization techniques developed till date, the feasibility check has to wait until after the new solution is already complete. This procedure may result in creating too many infeasible solutions and may slow down the algorithm considerably. However, PBO incorporates mechanisms to ensure feasibility during solution generation by considering the effect of changes in the environment in which the solution is generated and hence deals with the bottleneck of most of the optimization algorithms used till date.

This chapter is the basic work proposing a new optimization technique which opens up a wide range of possibilities for future research [Dasgupta, D., et. al., 2001]. It would be interesting to prove the expressions derived in Section 6.1 about the net force required in the ridge-push and the slab-pull case. The plate movement model that highlights the relevance of the critical temperature and the proposed fact that the phenomenon of ridge-push and slab-pull being mutually exclusive to each other has a very special structure that may provide a new dimension to the nature, type and the relationship of the forces acting on the plate with each other causing motion which is termed as plate tectonics phenomenon in literature. The properties of plate tectonics (such as the reduction in the effect of the external factors with time and the conveyor belt principle [(Thienen, P. V., et. al., 2004), (Turcotte, D. L., et. al., 2002)]) could have important implications for the behavior of PBO with respect to stability, convergence, equilibrium, and other issues. PBO can prove to be significantly useful when applied to the optimization of problems with dynamic fitness landscapes. This is achieved in the other evolutionary algorithms by using optimal filters to estimate solution fitness similar to what has been suggested for GAs [(Goldberg, D.,1989), (Michalewicz, Z., 1992)]. The details of the plate motion model in Fig. 6.2 could be adjusted to improve optimization performance. The contributing forces of ridge-push and slab-pull were used as the major factors governing the plate motion and were modeled in the PBO keeping all the other contributing forces as the external factors analogous to the tolerance level in PBO, however, these other forces like the tidal drag, coriolis effect, rotational effect forces etc. can be further analyzed and modeled into PBO. This means that the plate motion model can be further refined by considering the other driving forces of plate motion.

The plate tectonics literature is very rich and hence there are many possibilities for further work. Also the continental drift theory (or the plate tectonics theory) has implications for

biogeography too [Briggs, J. C., 1987] since the plate tectonics theory helps biogeographers to explain the disjunct biogeographic distribution of present day life found on different continents but having similar ancestors. In future, the performance of PBO can be quantified with respect to other algorithms for problems with specific features and hence an extension to this work could be finding an integration between biogeography and plate tectonics thereby leading to the development of a hybrid algorithm which could be applied for finding better solutions to the problems on which PBO or BBO are applied individually [Goel, L., et. al., 2012A].

This chapter has demonstrated the applicability of PBO towards engineering applications. From the results, it is observed that the cost factor achieved by PBO is the lowest among all the other evolutionary optimization algorithms for all the city sets in case of the TSP problem. Based on the results, it is observed that PBO far outperforms all the other evolutionary optimization algorithms and is capable of leading to an optimum solution in dynamic environments. However, almost every problem in engineering (and in life) can be interpreted as an optimization problem [(Back, T., 1996), (Dasgupta, D., et. al., 2001)] and hence, the new optimization algorithm of PBO can be applied for other applications as well for example, the satellite image classification, sensor selection for aircraft engine health estimation, enemy base station prediction, battlefield planning, etc. to name a few. Thus, the technique of PBO opens up promising avenues of productive research in varying applications.

Conclusion and future scope of the work

Chapter 7

7.1. Conclusion

7.2. Research Outcome and Future Scope of the work

7.1 Conclusion

Nature inspired computing has grown to be a popular heuristic for problem solving in diverse domains which creates a need for enriching the taxonomy of nature inspired computational intelligence techniques. One of the very important application domains has been targeted in this thesis that of terrain understanding based on remote sensing images. The datasets chosen are multi-spectral satellite datasets provided courtesy of DRDO for testing of the various models of nature inspired intelligence developed and explored in the thesis.

Chapter 1 has formulated the problem of developing novel optimization technique based on nature inspired computational intelligence which will work efficiently for terrain understanding applications that have very sophisticated datasets. The flow of the work done in the thesis to achieve the goal is summarized in figure 7.1 below. The thesis begins by developing performance analysis model of biogeography based feature extractor and establishes the concept of information sharing in swarm intelligence techniques. Based on the analytical model, two hybrid models are developed, the hybrid ACO2/PSO/BBO and the improved ACO2/PSO(BBO-GS) model. Based on the study of the above models, the thesis attempts to extend the original model of biogeography and proposes an extended BBO/EE optimization model which is hence adapted for both of the terrain understanding applications of the thesis. *Finally, the thesis develops new generic plate tectonics based optimization model based on the never explored domain of geo-sciences.*

The applications targeted included military, commercial and scientific applications of high impact and critical importance in real world. Also, the thesis successfully establishes the fact

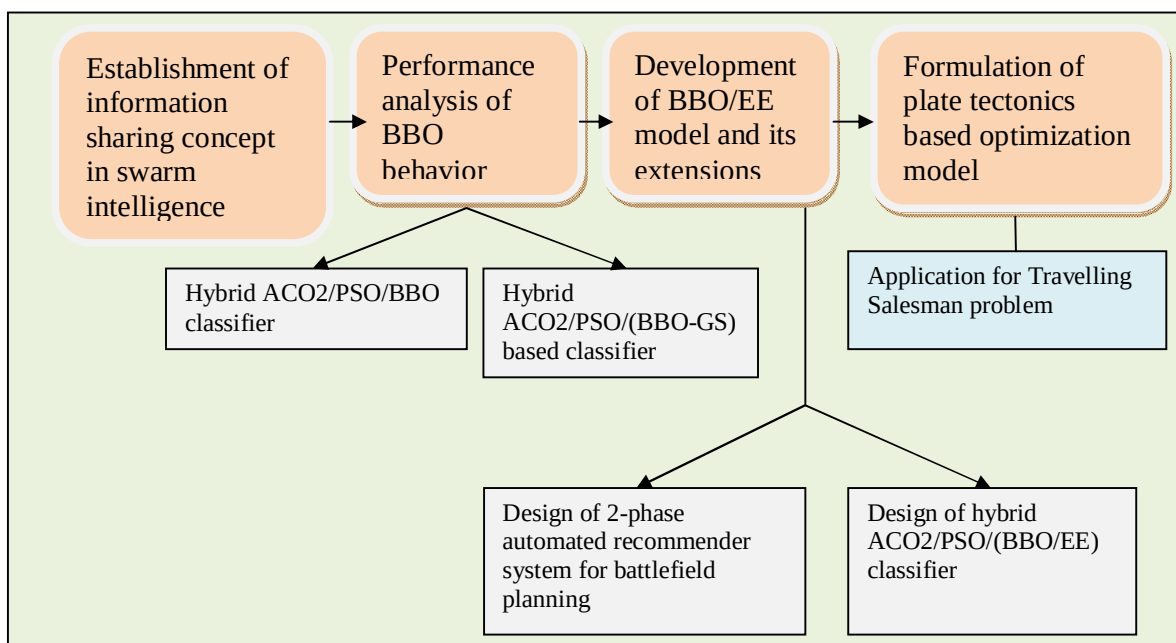


Figure 7.1: Summary of work done in the thesis

that nature inspired CI techniques are much efficient as compared to the traditional and human cognition based techniques.

The conclusion of the research work is as follows:

Chapter 2 has developed a mathematical framework for the behavioral analysis of the BBO technique over natural terrain features by means of a combined tabular representation of the percentage of classification efficiency, the degree of entropy and the similarity index for each land cover feature for each of the multi-spectral bands taken on two different datasets of Alwar and Patalganga. Based on the results, the performance governing equation of BBO has been established which states that the classification efficiency of BBO is inversely proportional to the degree of entropy and directly to the similarity index on a given land cover feature. Finally, the concept of information sharing has been established in swarm intelligence techniques of ACO, PSO, BBO and BFO.

Chapter 3 has designed two hybrid models based on the performance governing model of BBO, the first being an integration of the BBO technique with the ACO and PSO technique is able to classify the desirable features more efficiently than others based on a similarity pattern analysis of the training set data distribution graph. This classifier achieved a very high classification efficiency of 0.9818. The second hybrid model improved the first one by incorporating the geo-sciences concepts in BBO to design the hybrid ACO2/PSO/(BBO-GS) based classifier and is able to classify the homogeneous regions of the image very accurately. This classifier achieved a kappa coefficient of 0.987 which is one of the best classifier amongst the CI classifiers developed till date for feature extraction. This concludes that the performance of hybrid models is a significant improvement over the individual techniques applied for the same problem.

Chapter 4 developed the biogeography based optimization with extinction and evolution (BBO/EE) model which incorporated the concepts of extinction and evolution in the species abundance model of BBO. Six extended migration models were also developed. These models work very well on non-separable problems where the knowledge about the problem structure is required for problem solving. Also, these models simplify the process of migration in the original BBO and refine the results leading to faster convergence towards optimal solutions with reduced complexity indicated by the results where BBO/EE performed the best (on average) on seven of the 14 benchmark functions and outperformed the standard BBO on all the benchmark functions. Also, the CPU time for the BBO/EE reduced significantly (CPU time = 0.61) over the original BBO model which had a CPU time of 2.4.

In Chapter 5, the extended models were adapted based on the modified BBO/EE for both the terrain understanding applications taken up in the thesis. A two - phase recommender system

architecture was designed for intelligent preparation of the battlefield. The developed system is a milestone in the history of recommender system design and can be used by commanders in the battlefield for improved decision making. The chapter also demonstrated the applicability of BBO/EE for geo-spatial feature extraction and compared the performance of the BBO/EE based hybrid feature extractor with the original hybrid ACO2/PSO/BBO classifier and twelve other optimization methods that have been adapted till date for feature extraction application. The designed hybrid classifier is very effective in extraction of land cover features and outperformed the previously developed CI based classifiers as indicated by the results where its kappa coefficient was found to be 0.992 which is the highest value amongst all the CI classifiers developed till date for feature extraction.

Chapter 6 developed a novel optimization framework based on the entirely new geo-sciences based phenomenon of plate tectonics. The developed technique is able to utilize an optimal set of decision variables until all the problem constraints are satisfied and is able to adjust the weights of the decision variables dynamically according to the conveyor belt principle of plate tectonics. The performance of the proposed optimization algorithm was analyzed on a set of 14 standard benchmark functions and the results were compared with the other state of the art evolutionary optimization methods where it was found that the algorithm performed the best (on average) on 9 of the 14 benchmarks and the CPU time required was the lowest (0.31 sec) among all the optimization algorithms. The chapter also demonstrated its applicability on the TSP problem on standard TSP problem instances. The results indicate that PBO approach produces much better results than the other contemporary EAs applied on these TSP instances.

Thus, the goal of developing novel optimization techniques inspired from nature and natural phenomenon that can be very well suited to real world terrain understanding applications of high impact has been successfully attempted in this thesis.

7.2 Research Outcome and Future Scope

The research outcome of the thesis is extension of the taxonomy of nature inspired CI techniques [Engelbrecht, A. P., 2007] to include the techniques inspired from the concepts of geo-sciences. This is represented in figure 7.2. Thus, the revised taxonomy of computational intelligence based techniques will consist of the human cognition based, the artificial immune system based; swarm intelligence and the extended geo-sciences based intelligent techniques as the fourth pillar of computational intelligence techniques with the latter three categorized under nature inspired intelligence as given in the figure (proposed extension in red block).

This extended geo-sciences domain covers the fields of science like the physical geography which covers the aspects of geomorphology, oceanography, climatology and biogeography;

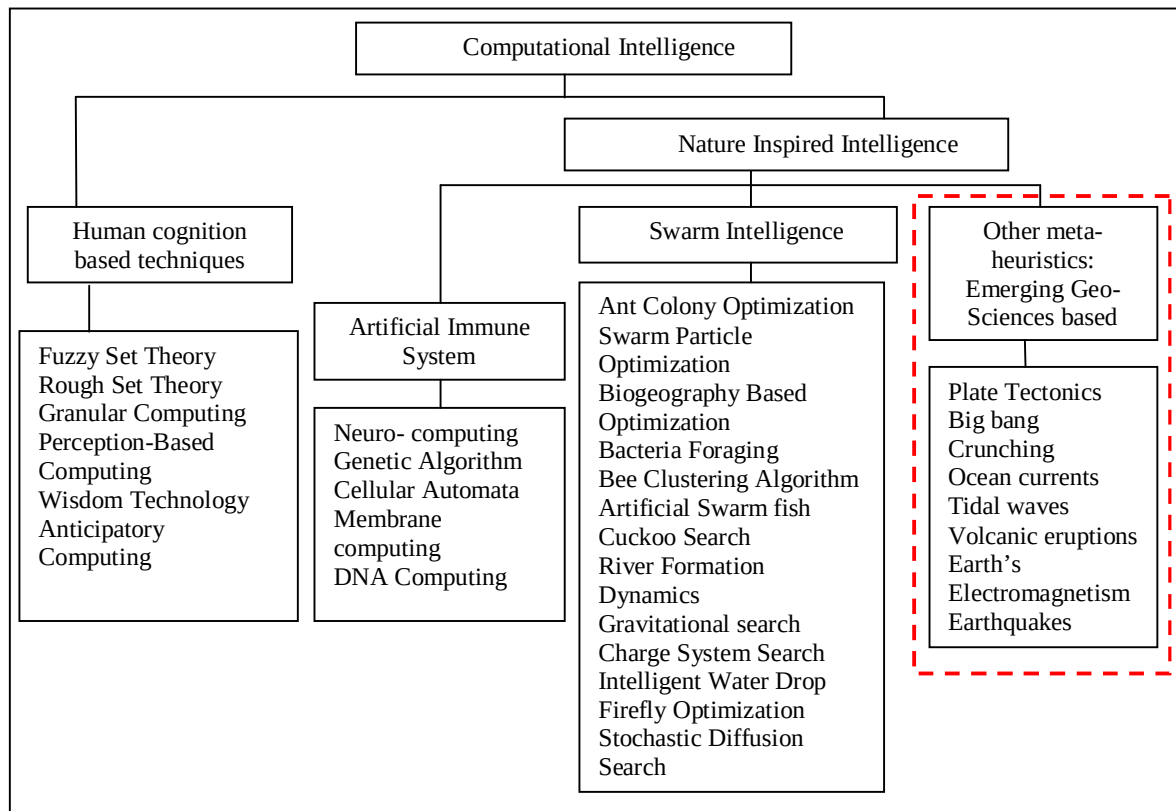


Figure 7.2: Taxonomy of computational Intelligence techniques with the proposed extension in dashed lines in red.

geophysics and geodesy which investigate the shape of the Earth, its reaction to forces and its magnetic and gravity fields; geophysicists which explore the earth's core and mantle as well as the tectonic and seismic activity of the lithosphere; the soil sciences which covers the outermost layer of the Earth's crust that is subject to soil formation processes; oceanography and hydrology which describe the marine and freshwater domains of the watery parts of the Earth; glaciology which covers the icy parts of the Earth and atmospheric sciences which cover the gaseous parts of the Earth between the surface and the exosphere. Based on the above discussion, the following nature inspired phenomenon are categorized as the basis of developing intelligent techniques under geo-sciences as listed below [(Gorny, V. I., 2002), (Briggs, J.C., 1987), (Moresi, L., et. al., 2000)]:

- Plate Tectonics: Based on energy transformations of earth's crust.
- Big bang Crunching: Based on energy transformations in the universe.
- Oceanic currents: Based on energy transformations in the watery parts of the earth.
- Tidal waves: Based on energy transformations in the sun and moon.
- Volcanic eruptions: Based on energy transformations of earth's crust.
- Earth's Electromagnetism: Based on energy transformations in the magnetic and gravity fields of earth.
- Earthquakes: Based on energy transformations of earth's crust.

The above identified categorizations under the geo-sciences domain can be very well used for deriving computational intelligence algorithms since nature always demonstrates its ability to reach an equilibrium condition analogous to an optimal state in problem solving. The techniques based on geo-sciences phenomenon are complex and involve non-linear dynamics which may lead to exponential time complexities however, guarantee optimal solutions in both static and dynamic environments and can be used for modeling complex systems in diverse domains like those of biology, physics, engineering, economy and management.

The new directions that this research has explored are discussed in the paragraphs below:

The thesis integrated BBO with ACO2/PSO to build a hybrid model for land cover feature extraction. In future, BBO can be integrated with newer techniques of artificial bee colony optimization, cuckoo search [Yang, X. S. et. al., 2010], bacterial foraging optimization, improvements of BBO like LBBO, BBO/RE, BBO/ES/RE, membrane computing, stochastic diffusion search, gravitational search [Rashedi, E., et. al., 2010], charged system search [Kaveh, A., et. al., 2010], support vector machines etc. for building highly autonomous and very efficient classifiers that can prove to be very close to becoming an ideal classifier. These classifiers will be able to support the analyst's decisive capabilities and truly enhance the process of algorithm selection for achieving maximum efficiency on different natural terrain features.

Furthermore, the thesis also studied complex biogeography on islands and added more factors to the definition of species count on a habitat taking hints from the population dynamics concepts and theories. The above factors extended the original model of biogeography. With the above work, an important extension to it can be incorporating other factors of biogeography into the proposed model in order to further sophisticate the species abundance model of biogeography proposed by us. These factors can be the dependencies among species, the habitat characteristics, the species migratory behavior, population cycles, etc.

The thesis applied BBO/EE for the design of recommender system for intelligent battlefield preparation and the application of feature extraction of satellite images. Other applications of the proposed biogeography based optimization with extinction and evolution algorithm can be explored in various terrain understanding domains such as mixed pixel resolution, travelling salesman problem, robot navigation, etc. Also, other domains such as face recognition, pattern recognition, sensor selection, economic load dispatch, etc. can be the future domains in which the applicability of the proposed approach can be effectively explored and successfully tested over the original BBO.

The thesis studied the phenomenon of plate tectonics of earth's dynamics and viewed it as the phenomenon which can be used to design an optimization algorithm. The phenomenon of plate tectonics can be studied further and the slab-pull and the ridge-push forces of plate motion can be further analyzed so as to determine closely the factors leading to the equilibrium condition of the plate. The external forces that are treated as analogous to secondary decision variables in the currently proposed model can be further refined and incorporated into the mainstream model to refine the model and strengthen the role of these factors over the equilibrium condition of the plate. The plate tectonics concept has been tested on standard instances of the TSP problem, however, in future, the algorithm can show a good applicability to the path planning applications on real world datasets as well. Further, other natural phenomenon in the proposed taxonomy of nature inspired computational intelligence such as the big-bang crunching, the earthquake mechanism, the volcanic eruption conditions, etc. can be studied closely and newer optimization algorithms can be developed taking inspiration from these.

Based on the performance governing model of biogeography based land cover feature extractor, the thesis developed an application of integrating the biogeography based extractor with geo-sciences concept of entropy which is one of the governing factors of the efficiency of biogeography based extractor. In future, other applications can be targeted where entropy can be calculated in any other spectral band in case of applications where the dataset is terrain multi-spectral images for example, mixed pixel resolution, design of hybrid classifiers, etc. or simply formulating the entropy function according to its standard definition and evaluating the fitness function for applications where the dataset are normal images, for example, in case of face recognition application, pattern recognition, etc. The thesis establishes the concept of information sharing in swarm intelligence techniques and analyzes four techniques of ACO, PSO, BBO and bacterial foraging optimization. Furthermore, newer techniques like ABC, gravitational search, etc. may be explored and added to this categorization in order to fully establish the concept of information sharing as a powerful paradigm which serves as the basis of the mechanism of feature extraction.

REFERENCES

- [1] Aamodt, A., Plaza, E., “*Case Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches*”, Artificial Intelligence Communications, IOS Press, vol. 7(1), pp. 39-59, 1994.
- [2] Accad, A. and Neil, D.T., “*Lowland Rainforest Structural Vegetation Communities of Northeastern Australia: Spatial Response To Predicted Climate Change*”, In: 18th World IMACS / MODSIM Congress, Cairns, Australia: Proceedings, pp. 2028-2034, 2009.
- [3] Afaq, H. and Saini, S., “*On the Solutions to the Travelling Salesman Problem using Nature Inspired Computing Techniques*”, IJCSI International Journal of Computer Science Issues, vol. 8(4), pp. 326-334, 2011.
- [4] Alpaydin, E., “*Introduction to Machine Learning*”, MIT Press, 2004.
- [5] Anderson, J. R., Hardy, E. E., Roach, J. T. and Witmer, R. E., “*A Land Use and Land Cover Classification System for Use with Remote Sensor Data*”, Geological Survey Professional Paper 964, pp. 28, 1976.
- [6] Back, T., “*Evolutionary Algorithms in Theory and Practice*”, Oxford, U.K.: Oxford Univ. Press, 1996.
- [7] Baluja, S., “*Population-Based Incremental Learning: A Method for Integrating Genetic Search Based Function Optimization and Competitive Learning*”, Technical Report (Pittsburgh, PA: Carnegie Mellon University) (CMU-CS-94-163), 1994.
- [8] Banerjee, S., Bharadwaj, A., Gupta, D., Panchal, V.K., “*Remote Sensing image Classification using Artificial Bee Colony Algorithm*”, International Journal of Computer Science and Informatics, vol. 2(3), pp. 67-72, 2012.
- [9] Bansal, S., Gupta, D., Panchal, V.K., Kumar, S., “*Swarm Intelligence Inspired Classifiers in Comparison with Fuzzy and Rough Classifiers: A Remote Sensing Approach*”, Contemporary Computing, Communications in Computer and Information Science, vol. 40, U.S.A.: Springer, pp. 284-294, 2009.
- [10] Bargiela, A. and Pedrycz, W., “*Granular Computing*,” Kluwer, Boston, 2003.
- [11] Bargiela, A. and Pedrycz, W., “*The roots of granular computing*,” Proceedings of the 2006 IEEE Int. Conf. on Granular Computing (GrC’06), pp. 806-809, 2006.
- [12] Beattie, P.D. & Bishop, J.M., “*Self-Localisation in the 'Senario' Autonomous Wheelchair*”, Journal of Intelligent and Robotic Systems, vol. 22, pp 255–267, Kluwer Academic Publishers, 1998.
- [13] Beyer, H., “*The Theory of Evolution Strategies*”, New York: Springer, 2001.
- [14] Bhattacharya, A., Chattopadhyay, P., “*Application of biogeography-based optimization for*

- solving multi-objective economic emission load dispatch problems*”, Electric Power Components and Systems, Taylor and Francis, vol. 38(3), pp. 340-365, 2010.
- [15] Bhattacharya, A., Bhattacharya, A., Datta, S., Basu, M., “*Gravitational Search Algorithm Optimization for short-term hydrothermal scheduling*”, In: Proceedings of International Conference on Emerging Trends in Electrical Engineering and Energy Management (ICETEEEM), IEEE Publications, Chennai, pp. 216-221, 2012.
 - [16] Bishop, J.M., “*Stochastic Searching Networks*”, In: Proceedings of 1st IEEE International Conference on Artificial Neural Networks, pp 329–331, London, 1989.
 - [17] Bishop, Y., Fienberg, S. and Holland, P., “*Discrete Multivariate Analysis | Theory and Practice*”, MIT Press, Cambridge, 1975.
 - [18] Blum, C., “*Ant colony optimization: Introduction and recent trends*,” Physics of Life Reviews, Elsevier, vol. 2(4), pp. 353–373, 2005.
 - [19] Bonabeau, E., Dorigo, M. and Theraulaz, G., “*Swarm Intelligence from Natural to Artificial System*”, First Edition, Oxford University Press, U.S.A., pp.1-24, 1999.
 - [20] Bratton and Kennedy, J., “*Defining a Standard for Particle Swarm Optimization*”, In: proceedings of the IEEE Swarm Intelligence Symposium, Honolulu, Hawaii, USA, pp. 120-127, 2007.
 - [21] Briggs, J.C., “*Biogeography & Plate Tectonics*”, Elsevier Science Publications, 1987.
 - [22] Brumby, S. P., Theiler, J., Perkins, S. J., Harvey, N., Szymanski, J. J., Bloch, J. J., Mitchell, M., “*Investigation of Image Feature Extraction by a Genetic Algorithm*”, Proc. SPIE vol. 3812, Applications and Science of Neural Networks, Fuzzy Systems, and Evolutionary Computation II, Bosacchi, Bruno, Fogel, David B., Bezdek, James C.; Eds., pp. 24-31, 1999.
 - [23] Burchardt, H. and Salomon, R., “*Implementation of Path Planning using Genetic Algorithms on Mobile Robots*”, IEEE Congress on Evolutionary Computation, pp. 1831-1836, 2006.
 - [24] Clerc, M., “*Particle Swarm Optimization*”, ISTE Publishing, Amsterdam, The Netherlands, 2006.
 - [25] Congalton, R.G., “*Assessing Landsat classification accuracy using discrete multivariate analysis statistical techniques*”, Photogrammetric Engineering and Remote Sensing, vol. 49(12), pp. 1671-1678, 1983.
 - [26] Cohen, J., “*A coefficient of agreement for nominal scales*”, Educational and Psychological Measurement”, vol. 20(1), pp. 37-40, 1960.
 - [27] Currie, David J., “*Global Ecology and Biogeography*”, U.K.: Blackwell Publishing Ltd, 2012.

- [28] Das S., Biswas A., Dasgupta S., Abraham A., “*Bacterial Foraging Optimization Algorithm: Theoretical Foundations, Analysis and Applications*”, SpringerLink, Foundations of Computational Intelligence, vol. 3, SCI 203, pp 23-55, 2009.
- [29] Dasgupta, D., Michalewicz, Z., Eds., “*Evolutionary Algorithms in Engineering Applications*”, Springer, New York, 2001.
- [30] Dasgupta, D., Yu, S., Nino, F., “*Recent Advances in Artificial Immune Systems: Models and Application*”, Applied Soft Computing, Elsevier, vol. 11, pp 1574-1587, 2011.
- [31] Dawei D., Simon, D, Ergezer, M., "Biogeography-Based Optimization Combined with Evolutionary Strategy and Immigration Refusal", In: IEEE International Conference on Systems, Man and Cybernetics (SMC), San Antonio, TX, pp. 997-1002, 2009.
- [32] Demetris, S. and Anthanassios, V., “*Comparison of computational Intelligence based Classification Techniques for Remotely Sensed optical Image Classification*”, IEEE Transactions on Geosciences and Remote Sensing , vol. 44(8), pp. 2305 - 2318, 2006.
- [33] Dorigo, M., Gambardella, L., Middendorf, M., Stutzle, T., Eds., “*Special section on ant colony optimization*”, IEEE Transactions on Evolutionary Computing, vol. 6(4), pp. 317–365, 2002.
- [34] Dorigo, M. and Stuetzle, T., “*Ant Colony Optimization*”, MIT Press, 2004.
- [35] Duan, H., Liu, S., Lei, X., “*Air Robot Path Planning Based on Intelligent Water Drops Optiimization*”, IEEE International Joint Conference on Neural Networks (World Congress on Computational Intelligence), Hong Kong, pp. 1397-1401, 2008.
- [36] Dusan, T., and Dell'orco, M., “*Bee colony optimization: A cooperative learning approach to complex transportation problems*”, In: Proceedings of 16th Mini-EURO Conference on Artificial Intelligence Methods in Transportation, Poznan, Poland, pp. 51-60, 2005.
- [37] Eberhart, R., Shi, Y., Kennedy, J., “*Swarm Intelligence*”, San Mateo, CA: Morgan Kaufmann, 2001.
- [38] Eberhart, R. and Y. Shi, “*Special issue on particle swarm optimization,*” IEEE Transactions on Evolutionary Computation, vol. 8(3), pp. 201–228, 2004.
- [39] Engelbrecht, A. P., “*Computational Intelligence: An Introduction*”, 2nd Edition, John Wiley and Sons, 2007.
- [40] Ergezer, M., Simon, D., Du, D.W., “*Oppositional biogeography-based optimization*”, in: Proceeding of IEEE the Conference on Systems, Man, and Cybernetics, San Antonio, TX, pp. 1035–1040, 2009.
- [41] Fausett, L., “*Fundamentals of Neural Networks*”, New York: Prentice Hall, 1994.
- [42] Finck, S., Hansen, N., Ros, R., and Auger, A., “*Real-parameter black-box optimization benchmarking 2009: Presentation of the noiseless functions*”, Technical Report 2009/20, Research Center PPE, 2009.

- [43] Foody, G.M., “*Status of land cover classification accuracy assessment*”, Remote Sensing of Environment, vol. 80, pp. 185-201, 2002. --- (A)
- [44] Foody, G.M. and Atkinson P. M., “*Uncertainty in Remote Sensing and GIS*”, John Wiley & Sons, England, 2002. --- (B)
- [45] Foody, G.M. “*The evaluation and comparison of thematic maps derived from remote sensing*”, In: Caetano, M. and Painho, M. (eds), Proceedings of the 7th International Symposium on Spatial Accuracy Assessment in Natural Resources and Environmental Sciences, 2006, Lisboa, Instituto Geográfico Português, 2006.
- [46] Franklin, J., “*Predictive vegetation mapping: Geographic modelling of biospatial patterns in relation to environmental gradients*”, Progress in Physical Geography, vol. 19(4), pp. 474-499, 1995.
- [47] Freitas, A.A., “*A Genetic Programming Framework for Two Data Mining Tasks: Classification and Generalized Rule Induction*”, In: Proceedings of the 2nd Annual Conference on Genetic Programming, Stanford University, CA, USA, Morgan Kaufmann, page 96 - 101, 1997.
- [48] Gao, Y., Chen, Y., “*The optimization of water utilization based on artificial fish-swarm algorithm*”, In: Sixth International Conference on Natural Computation (ICNC), Yantai, Shandong, vol. 8, pp. 441-4419, 2010.
- [49] Goel, L., Gupta, D., Panchal, V.K., “*Hybrid ACO-BBO Approach for predicting the deployment strategies of enemy troops in a military terrain application*”, in: 4th International Multi-Conference on Intelligent Systems & Nanotechnology (IISN), pp. 118-122, 2010. --- (A)
- [50] Goel, L., “*Land cover Feature Extraction using Hybrid Swarm Intelligence Techniques-A Remote sensing perspective*”, ACEEE International Journal on Signal & Image Processing, vol. 1(3), pp. 14-16, 2010. --- (B)
- [51] Goel, L., Gupta, D., Panchal, V.K., “*Embedding Expert Knowledge to Hybrid Bio-Inspired Techniques- An Adaptive Strategy Towards Focused Land Cover Feature Extraction*”, International Journal of Computer Science and Information Security, vol. 8(2), pp. 244-253, 2010. --- (C)
- [52] Goel, L., Gupta, D., Panchal, V.K., “*Performance Governing Factors of BBO for Land Cover Feature Extraction: An Analytical Study*”, In: IEEE World Congress on Information and Communication Technologies (WICT), IEEE Publications, Mumbai, India, pp. 165-170, 2011. ---(A)
- [53] Goel, L., Gupta, D., Panchal, V.K., “*Information Sharing in Swarm Intelligence Techniques : A Perspective Application for Natural Terrain Feature Elicitation*”, International Journal of Computer Applications, vol. 32(2), pp. 34-40, 2011. --- (B)

- [54] Goel, L., Gupta, D., Panchal, V.K., “ *Biogeography and Plate Tectonics based Optimization for Water body Extraction in Satellite Images*”, In: International Conference on Soft Computing and Pattern Recognition (SocProS), Advances in Soft Computing series of Springer, Roorkee, India: Springer Publications, vol. 131, pp. 1-10, 2011. --- (C)
- [55] Goel, L., Gupta, D., Panchal, V.K., “*Hybrid bio-inspired techniques for land cover feature extraction: A Remote Sensing Perspective*”, Applied Soft Computing Journal, Elsevier Publications, vol. 12(2), pp. 832-849, 2012. --- (A)
- [56] Goel, L., Gupta, D., Panchal, V.K., “*Dynamic model of blended biogeography based optimization for land cover feature extraction*”, In: International Conference on Contemporary Computing (IC3), Communications in Computer & Information Sciences (CCIS-LNCS), Springer Publications, vol. 306, pp. 8-19, 2012. --- (B)
- [57] Goel, L., Gupta, D., Panchal, V.K., “*Biogeography based Anticipatory Computing Framework for Intelligent Battlefield Planning*”, IEEE International Conference on Hybrid Intelligent Systems (HIS), Pune, pp. 41-46, 2012. --- (C)
- [58] Goel, L., Gupta, D., Panchal, V.K., “*Extended species abundance models of biogeography based optimization*”, IEEE conference on Computational Intelligence, Modelling and Simulation (CIMSIm), IEEE Xplore and CSDL, Kuantan, Malaysia, pp. 7-12, 2012. ---(D)
- [59] Goel, L., Gupta, D., Panchal, V.K., “*Biogeography and Geo-Sciences Based Land Cover Feature Extraction: A Remote Sensing Perspective*”, Applied Soft Computing, Elsevier Publications, vol. 13(10), pp. 4194 – 4208, 2013. --- (A)
- [60] Goel, L., Gupta, D., Panchal, V.K., “*Land Cover Feature Extraction of Multi-spectral Satellite Images Based on Extended Species Abundance Model of Biogeography*”, Transactions on Computational Science, Springer Publications, vol. 21, pp. 90-118, 2013. --- (B)
- [61] Goel, L., Gupta, D., Panchal, V.K., “*Performance Analysis of Biogeography Based Land Cover Feature Extractor for Building Hybrid Intelligent Models*”, International Journal of Applied Evolutionary Computation, IGI Global Publications, vol. 4(4), pp. 1-26, 2013. --- (C)
- [62] Goel, L., “*Nature Inspired Intelligence: A Modern Tool for Warfare Strategic Decision Making*” in the book titled “Case Studies in Intelligent Computing: Achievements and Trends”, **Taylor and Francis Publications, CRC Press**, pp. 226-244, 2014.
- [63] Goel, L., Panchal, V. K., “*Feature Extraction through Information Sharing in Swarm Intelligence Techniques*”, in the book titled "Knowledge based processes for software development", **IGI Global Publishers**, pp. 151-175, 2014.
- [64] Goel, S., Sharma, A., Panchal, V.K., “*Biogeography Based Optimization, an, Evolutionary Technique or a Swarm Based Approach?*”, IISN, Jagadhri, India, 2011.

- [65] Goldberg, D., "*Genetic Algorithms in Search, Optimization, and Machine Learning*", Reading, MA: Addison-Wesley, 1989.
- [66] Gornyi, V.I., "*The Mantle Convection and the Drift of Euro-Asian Plate (According the Remote Geothermal Method Data)*", IEEE International Geo-science and Remote Sensing Symposium, IGARSS '02, vol. 4, pp. 2029-2035, 2002.
- [67] Gupta, D., Das, B. and Panchal, V. K., "*A Methodical Study for the Extraction of Landscape Traits Using Membrane Computing Technique*", In: Proceedings of GEM' (WORLDCOMP), 2011.
- [68] Gupta, D., Goel, L., Abhishek, "*An Efficient Biogeography Based Face Recognition Algorithm*", CSE, Atlantis Press, pp. 64-67, 2013. --- (A)
- [69] Gupta, D., Goel, L., Aggarwal, V., "*A Hybrid Biogeography Based Heuristic for the Mirrored Traveling Tournament Problem*", Sixth International Conference on Contemporary Computing (IC3), IEEE Publications, Catalog no. CFP 1381U-CDR, pp. 325-330, 2013. --- (B)
- [70] Gupta, D., Das, B., Panchal, V.K., "*Applying case based reasoning to Cuckoo Search for the Expedition of Groundwater Exploration*", In: Proceedings of Seventh International Conference on Bio-Inspired Computing: Theories and Applications (BICTA 2012), Advances in Intelligent Systems and Computing, Springer Publications, vol. 201, pp 341-353, 2013. --- (C)
- [71] Gupta, N., Singh, V., Panchal, V.K., Das, B., "*Membrane Computing for water body area extraction in Satellite Image*", IISN, Jagadhri, India, 2011. --- (A)
- [72] Gupta, N., Panchal, V.K., "*Swarm Intelligence for mixed pixel resolution*", In: Computers & Simulations in Modern Science, U.S.A.: WSEAS press, vol. 5, pp. 190-194, 2011. --- (B)
- [73] Gupta, S., Arora, A., Panchal, V.K. and Goel, S., "*Extended Biogeography based Optimization for Natural Terrain Feature Classification from Satellite Remote Sensing Images*", In: Communications in Computer and Information Science (LNCS), Springer Publications, vol. 168, pp. 262-269, 2011.
- [74] Hand, J., "*Construction and Assessment of Classification Rules*", Wiley, 1997.
- [75] Hansen, N., Finck, S., Ros, R. Auger., and A., "*Real-parameter black-box optimization benchmarking 2009: Noiseless functions definition*", Technical Report RR-6829, INRIA, 2009 (updated February 2010).
- [76] Hansen, N., Auger, A., Finck, S. and Ros., R., "*Real-parameter black-box optimization benchmarking 2012: Experimental setup*", Technical Report RR-7215., INRIA, ISSN 0249-6399, 2012.
- [77] Harald, N., "*Introduction to Cellular Automata*", Seminar: Organic Computing, SS2006, 2006.

- [78] Haykin, S., “*Neural Network: A comprehensive foundation*”, 2nd Edition, Prentice-Hall, 1999.
- [79] Heerman P.D. and Khazenie, N., “*Classification of multispectral remote sensing data using a back propagation neural network*,” IEEE Transactions on Geosciences and Remote Sensing, vol. 30(1), pp.81-88, 1992.
- [80] Holden, N. and Freitas, A.A., “*A hybrid particle swarm/ant colony algorithm for the classification of hierarchical biological data*”, In: Proceedings of IEEE Swarm Intelligence Symposium (SIS-05), pp. 100-107, 2005.
- [81] Iaffaldano, G. and Bunge, H., “*Relating rapid plate-motion variations to plate-boundary forces in global coupled models of the mantle/lithosphere system: Effects of topography and friction*”, Tectonophysics, Elsevier Publications, vol. 474(1-2), pp. 393–404, 2009.
- [82] Jaiswal, R. K., Mukherjee, S., Raju, D., Saxena, R., “*Forest Fire Risk Zone Mapping from Satellite Imagery and GIS*”, International Journal of Applied Earth Observation and Geoinformation, Elsevier Publications, vol. 4(1), pp. 1-10, 2002.
- [83] Johal, N. K., Singh, S. and Kundra, H., “*A hybrid FPAB/BBO Algorithm for Satellite Image Classification*”, International Journal of Computer Applications (0975 – 8887), vol. 6(5), pp. 31-36, 2010.
- [84] Johnson C.G., “*Teaching Natural Computation*”, IEEE Computational Intelligence Magazine, vol. 4(1), pp. 24-30, IEEE Press Piscataway, NJ, USA , 2009.
- [85] Kamkar I., Akbarzadeh, T., Mohammad, R., Yaghoobi, M., “*Intelligent Water Drops a new optimization algorithm for solving the Vehile Routing Problem*”, IEEE International Conference on Systems Man and Cybernetics (SMC), Istanbul, pp. 4142 – 4146, 2010.
- [86] Kang, F., Li, J., Ma, Z., “*Rosenbrock artificial bee colony algorithm for accurate global optimization of numerical functions*”, Information Sciences, Elsevier Publications, vol. 181(16), pp. 3508-3531, 2011.
- [87] Kang, F. Li, J., Xu, Q., “*Damage detection based on improved particle swarm optimization using vibration data*”, Applied Soft Computing, Elsevier Publications, vol. 12(8), pp. 2329-2335, 2012.
- [88] Kang, F., Li, J., Li, H., “*Artificial bee colony algorithm and pattern search hybridized for global optimization*”, Applied Soft Computing, Elsevier Publications, vol. 13(4), pp. 1781-1791, 2013. --- (A)
- [89] Kang, F., Li, J., Ma, Z., “*An artificial bee colony algorithm for locating the critical slip surface in slope stability analysis*”, Engineering Optimization, Taylor and Francis Publications, vol. 45(2), pp. 207-223, 2013. --- (B)
- [90] Karaboga, D., “*An Idea Based On Honey Bee Swarm for Numerical Optimization*”, Technical Report-TR06, 2005.

- [91] Karaboga, D. and Bahriye, B., “*A powerful and efficient algorithm for numerical function optimization: artificial bee colony algorithm*”, Journal of Global Optimization, vol. 39, pp. 459-471, Springer, 2007.
- [92] Karaboga, D. and Bahriye, A., “*A comparative study of Artificial Bee Colony algorithm*”, Applied Mathematics and Computation, vol. 214(1), pp. 108–132, 2009.
- [93] Kari, L., Rozenberg, G., “*The Many Facets of Natural Computing*”, Communication of the ACM Magazine, New York, USA, vol. 51(10), pp. 72-83, 2008.
- [94] Kaur, P., Kaur, S., Aulakh, G. S., Panchal, V.K., Gill, I. S., “*Hybrid approach using PSO and ACO for Cross-Country Path Planning*”, In: Proceedings of International Conference on Artificial Intelligence and Pattern Recognition, Florida, U.S., pp. 1-6, 2010.
- [95] Kaveh, A., Talatahari, S., “*A novel heuristic optimization method: charged system search*”, Acta Mechanica, Springer, vol. 213, pp 267-289, 2010.
- [96] Kennedy, J., Eberhart, R., “*Particle Swarm Optimization*”, Proceedings of IEEE International Conference on Neural Networks, Perth, WA, vol. 4, pp. 1942–1948, 1995.
- [97] Khatib, W., and Fleming, P., “*The stud GA: A mini revolution?*,” In: Proceedings of the 5th International Conference on Parallel Problem Solving from Nature, London, UK: Springer, pp. 683-691, 1998.
- [98] Kim, J.H., Myung H., “*Evolutionary programming techniques for constrained optimization problems*”, IEEE Transactions on Evolutionary Computation, vol. 1(2), pp. 129–40, 1997.
- [99] Kim, B.N., Kwon, O.S., Kim, K.J., Lee, E.H., Hong, S.H., “*A Study On Path Planning for Mobile Robot Based on Fuzzy Logic Controller*”, In: Proceedings of IEEE TENCON, pp. 1-6, 1999.
- [100] Kumar, S., Gupta, D., Panchal, V.K., Kumar, S., “*Enabling Web Services For Classification of Satellite Images*”, In: International Conference on Semantic Web and Web Services (SWWS'09), Nevada, USA, CSREA Press, pp. 89-94, 2009.
- [101] Kundra, H., Panchal, V.K. and Singh, G., “*Biogeography Based Groundwater Exploration*”, International Journal of Computer Applications, vol. 1(8), pp. 87–91, 2010.
- [102] Lewin, D. I., “*DNA computing*”, Computing in Science & Engineering, vol. 4(3), pp. 5–8, 2002.
- [103] Lillesand, T.M., Kiefer, R.W. and Chipman, J.W., “*Remote Sensing and Image Interpretation*”, Fifth Edition, Wiley & Sons Ltd., England, pp.586-592, 2007.
- [104] Long, W., III, Srihann, S, Geoscience and Remote Sensing Symposium, Unsupervised and supervised classifications, “*Land cover classification of SSC image: unsupervised and supervised classification using ERDAS Imagine*”, In: IGARSS '04. Proceedings, vol.4, pp. 20-24, 2004.

- [105] Lunetta, R., Lyon, S., John, G., “*Remote Sensing and GIS Accuracy Assessment*”, CRC Press, U.S., 2004.
- [106] Ma, H., Suhong, N., and Man, S., “*Equilibrium Species Counts and Migration Model Tradeoffs for Biogeography based Optimization*”, IEEE Conference on Decision and Control, pp. 3306-3310, 2009.
- [107] Ma, H., “*An analysis of the equilibrium of migration models for biogeography-based optimization*”, Information Sciences, Elsevier Publications, vol. 180(18), pp. 3444–3464, 2010.
- [108] Ma, H., Simon, D., “*Blended Biogeography based optimization for constrained optimization*”, Engineering Applications of Artificial Intelligence, vol. 24(3), pp. 517-525, 2011.
- [109] Matsuyama, T., Iwamori, H., “*Analysis on the origin of toroidal plate motion and its application to the Philippine Seaplate*”, American Geophysical Union, Fall Meeting 2010.
- [110] Mei, M., Xue, H., Zhong, M., Gu, Y., “*An Improved Differential Evolution Algorithm for TSP Problem*”, International Conference on Intelligent Computation Technology and Automation, vol. 1, pp.544-547, 2010.
- [111] Michalewicz, Z., “*Genetic Algorithms, Data Structures + Evolutionary Programming*”, New York: Springer, 1992.
- [112] Momm, H.G., Easson, G., Joel, K., “*Evaluation of the use of spectral and textual information by an evolutionary algorithm for multi-spectral imagery classification*”, Journal of Computers, Environment and Urban Systems, Elsevier Publications, vol. 33(6), pp 463-471, 2009.
- [113] Moresi, L., Gurnis, M., Shijie, Z., “*Plate tectonics and convection in the earth’s mantle: toward a numerical simulation*”, International Journal of Computing in Science and Engineering, IEEE Publications, vol. 2(3), pp. 22-33, 2000.
- [114] Muhlenbein, H., Schlierkamp-Voosen, D., “*Predictive models for the breeder genetic algorithm- I. Continuous parameter optimization*”, Evolutionary Computation, vol. 1(1), pp. 25–49, 1993.
- [115] Niall, M., Woods, D., “*The computational power of membrane systems under tight uniformity conditions*”, Natural Computing, Springer, vol. 10, pp. 613-632, 2011.
- [116] Niu Q., Zhou, F., Zhou, T., “*Quantum Genetic Algorithm for Hybrid Flow Shop Scheduling Problems to minimize Total Completion Time*”, LSMS/ICSEE 2010, LNCS 6329, Springer, pp 21-29, 2010.
- [117] Óhrn, A. and Komorowski, J., “*A Rough Set tool kit for analysis of data*”, In: proceedings of 3rd International Joint Conference on Information Sciences, Durham, NC, pp. 403-407, 1997.

- [118] Omkar, S.N., Kumar, M., Mudigere, D. and Muley, D., “*Urban Satellite Image Classification using Biologically Inspired Techniques*”, In Proceedings of IEEE International Symposium on Industrial Electronics, Vigo, Spain, pp. 1767 – 1772, 2007.
- [119] Otero, F. E. B., Freitas, A. A. and Johnson, C. G., “*cAnt-Miner: An Ant Colony Classification Algorithm to Cope with Continuous Attributes*”, In: ANTS 2008, LNCS 5217, pp. 48-59, Springer-Verlag Berlin Heidelberg, 2008.
- [120] Ovreiu, M., Simon, D., “*Biogeography-based optimization of neuro-fuzzy system parameters for diagnosis of cardiac disease*”, In: Proceedings of Genetic and Evolutionary Computation Conference (GECCO), Portland, Oregon, pp. 1235-1242, 2010.
- [121] Pal, S.K. and Skowron, A., “*Rough Fuzzy Hybridization: A new Trend in Decision Making*”, First Edition, Singapore: Springer-Verlag publishers, 1999.
- [122] Panchal, V.K, Singhal, N., Kumar, S., Bhakna, S., “*Rough-Fuzzy Sets Tie-Up for Geospatial Information*”, In: Proceedings of International Conference on Emerging Scenarios in Space Technology and Applications (ESSTA2008), Chennai, India, vol. 1, 2008.
- [123] Panchal, V.K., Singh, P., Kaur, N., Kundra, H., “*Biogeography based satellite image classification*”, International Journal of Computer Science and Information Security, vol. 6(2), pp. 269–274, 2009.
- [124] Panchal, V.K., Bhugra, D., Goel, S. and Singhania, V., “*Study of the behavior of BBO over natural terrain features*”, In: 3rd International Conference on Electronics Computer Technology (ICECT), Kanyakumari, India, vol. 4, pp. 28 – 32, 2011.
- [125] Pappula, L., “*Application of real coded genetic algorithm for target sensing*”, In: Sixth International Conference on Sensing Technology (ICST), Kolkata, India, pp. 69 – 72, 2012.
- [126] Parmee, I., “*Evolutionary and Adaptive Computing in Engineering Design*”, New York: Springer, 2001.
- [127] Parpinelli, R.S., Lopes, H.S. and Freitas, A.A., “*Data Mining with an Ant Colony Optimization Algorithm*”, IEEE Transactions on Evolutionary Computation, Special issue on Ant Colony algorithms, vol. 6(4), pp. 321-332, 2002.
- [128] Passino, K.M., “*Biomimicry of Bacterial Foraging for Distributed Optimization and Control*”, IEEE Control Systems Magazine, vol. 22(3), pp. 52 – 67, 2002.
- [129] Patrick, C. H., “*The Generation of Situational Awareness within Autonomous Systems – A Near to Mid Term Study*”, Issues by Defence Systems Analysis Division Information Sciences Laboratory, DSTO-TR-1896, DSTO Information Sciences Laboratory, Edinburgh, Australia, 2006.
- [130] Patterson, D., “*Artificial Neural Networks*”, Singapore: Prentice Hall, 1996.

- [131] Paun, G., Rozenberg, G., Salomaa, A., “*DNA Computing - New Computing Paradigms*”, Series: Texts in Theoretical Computer Science, Springer-Verlag, 1998.
- [132] Paun, G., “*Membrane Computing: An Introduction*”, Series: Natural Computing Series, Springer, 2002.
- [133] Pawlak, Z., “*Rough Sets*”, International Journal of Computer and Information Science, Springer, vol. 11(5), pp. 341-356, 1982.
- [134] Pawlak, Z., “*Rough set approach to knowledge-based decision support*”, European Journal of Operational Research, vol. 99(1), pp. 48-57, 1997.
- [135] Pawlak, Z., “*Rough Set Theory and its Applications to Data Analysis*”, Cybernetics and Systems, Taylor and Francis, vol. 29(7), pp. 661-688, 1998.
- [136] Peddle, D. R., White, H. P., Soffer, R.J., Miller, J. R., LeDrew, E. F., “*Reflectance processing of remote sensing spectro-radiometer data*”, Computers & Geo-sciences, Elsevier Publications, vol. 27(2), pp. 203-213, 2001.
- [137] Piatrik, T. and Izquierdo, E., “*Image Classification Using an Ant Colony Optimization Approach*”, SAMT 2006, LNCS 4306, pp. 159 – 168, 2006.
- [138] Plaza, A., Chein-I, C., “*High-Performance Computer Architectures for Remote Sensing Data Analysis: Overview and Case Study*”, In: Antonio Plaza and Chein-I Chang (Ed.), High Performance Computing in Remote Sensing, FL, US: Chapman and Hall/ CRC, Taylor and Francis Publications, pp. 9-43, 2008.
- [139] Pošík P, Huyer W, Pál L., “*A Comparison of Global Search Algorithms for Continuous Black-Box Optimization*”, Evolutionary Computation Journal, MIT Press, vol. 20(4), pp. 509-541, 2012 (doi: 10.1162/EVCO_a_00084).
- [140] Price, K., and Storn, R., “*Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces,*” Journal of Global Optimization, Springer, vol. 11(4), pp. 341–359, 1997.
- [141] Rabanal, P., Rodríguez, I., Rubio, F., “*Applying River Formation Dynamics to the Steiner Tree Problem*”, In: Proceedings of 9th IEEE International Conference on Cognitive Informatics (ICCI), Beijing, pp. 704-711, 2010.
- [142] Rarick, R., Simon, D., Villaseca, F.E., Vyakaranam, B., “*Biogeography-based optimization and the solution of the power flow problem*”, In: Proceedings of the IEEE Conference on Systems, Man, and Cybernetics, San Antonio, TX, pp. 1003–1008, 2009.
- [143] Rashedi E., Nezamabadi-pour, H., Saryazdi, S., “*BGSA: binary gravitational search algorithm*”, Natural Computation, Springer, vol. 9(3), pp 727-745, 2010.
- [144] Richards, M., Gordon, R. and Hilst, R., “*The History and Dynamics of Global Plate Motions*”, Published by the American Geophysical Union as part of the Geophysical Monograph Series, vol. 121, pp. 5–46, 2000.

- [145] Rogan, J., Miller, J., Stow, D., Franklin, J., Levien, L. and Fischer, C., “*Land-cover change Monitoring with classification trees using Landsat TM and ancillary data*”, Photogrammetric Engineering and Remote Sensing, vol. 69(7), pp. 793-804, 2003.
- [146] Rosenfield, G. and Fitzpatrick-Lins, K., “*A coefficient of agreement as a measure of thematic classification accuracy*”, Photogrammetric Engineering and Remote Sensing, vol. 52(2), pp. 223-227, 1986.
- [147] Sauer, J. G. and Coelho, L., “*Discrete Differential Evolution with Local Search to Solve the Traveling Salesman Problem: Fundamentals and Case Studies*”, 2009, In: Proceedings of 7th IEEE International Conference on Cybernetic Intelligent Systems, London, pp. 1-6, 2008.
- [148] Simon, D., “*Biogeography Based Optimization*”, IEEE Transactions on Evolutionary Computation, vol. 12(6), pp. 702-713, 2008.
- [149] Simon, D., Du, D. and Ergezer, M., “*Biogeography - Based Optimization Combined with Evolutionary Strategy and Immigration Refusal*”, In: proceedings of IEEE International Conference on System, Man and Cybernetics, San Antonio, TX, U.S.A., pp. 997-1002, 2009.
- [150] Simon, D., “*A Dynamic System Model of Biogeography based Optimization*”, Applied Soft Computing, Elsevier Publications, vol. 11(8), pp. 5652-5661, 2011.
- [151] Simon, D., Omran, M., and Clerc, M., “*Linearized Biogeography-Based Optimization with Re-initialization and Local Search*,” Information Sciences, Elsevier Publications, vol. 267, pp. 140–157, 2014.
- [152] Sinha, S., Bhola, A., Panchal, V.K., Singhal, S., Abraham, A., “*Resolving Mixed Pixels by Hybridization of Biogeography Based Optimization and Ant Colony Optimization*”, In: Proceedings of IEEE World Congress on Computational Intelligence (WCCI), Brisbane, Australia, pp. 126-131, 2012.
- [153] Solomon R., “*Evolutionary algorithms and gradient search: similarities and differences*”, IEEE Transactions on Evolutionary Computation, vol. 2(2), pp. 45–55, 1997.
- [154] Solomon, S. C., Sleep, N. H. and Richardson, R. M., “*On the Forces Driving Plate Tectonics: Inferences from Absolute Plate Velocities and Intra-plate Stress*”, Geophysical Journal of the Royal Astronomical Society, vol. 42(2), pp. 769–801, 1975.
- [155] Solomon, Z. , Dobrowski, H., Safford, D., Cheng, Y. B., Ustin, S. L., “*Mapping mountain vegetation using species distribution modeling, image-based texture analysis, and object-based classification*”, Applied Vegetation Science, Wiley, vol. 11(4), pp. 499–508, 2008.
- [156] Song, Y., Liu, M., Wang, Z., “*Biogeography-based Optimization for the Travelling Salesman Problems*”, Third International Joint Conference on Computational Science and Optimization (CSO), Huangshan, Anhui, China, vol. 1, pp. 295-299, 2010.

- [157] Storn, R., “*System design by constraint adaptation and differential evolution*,” IEEE Transactions on Evolutionary Computation, vol. 3(1), pp. 22–34, 1999.
- [158] Summeren, J. V., Conrad, C. P., Bertelloni, C. L., “*The importance of slab pull and a global asthenosphere to plate motions*”, Geochemistry Geophysics Geosystems, Wiley, vol. 13(2), 2012.
- [159] Thienen, P. V., Vlaar, N.J., Berg, A.P., Van, D., “*Plate tectonics on the terrestrial planets*”, Physics of the Earth and Planetary Interiors, Elsevier Publications, vol. 142(1-2), pp. 61–74, 2004.
- [160] Timmis, J., Hone, A., Stibor, T., Clark, E., “*Theoretical advances in artificial immune systems*”, Journal of Theoretical Computer Science, Elsevier, vol. 403(1), pp 11-32, 2008.
- [161] Turchin, P., “*Complex Population Dynamics: a Theoretical/Empirical Synthesis*”, Princeton, NJ: Princeton University Press, 2003
- [162] Turcotte, D. L. and Schubert, G., “*Geodynamics*”, Wiley 2nd edition, 2002.
- [163] Tzu, S., “*The Art of War, a chinese military treatise*”, Chapter 3, 6th century BC.
- [164] USGS, “*The spatial data transfer standard*”, United States Geological Survey, Draft, January 1990.
- [165] Uyeda S. and Forsyth, D., “*On the relative importance of the driving forces of plate motion*”, Geophysical Journal of the Royal Astronomical Society, Wiley, vol. 43(1), pp. 163–200, 1975.
- [166] Varshney, T., Goel, L., Gupta, D., Panchal, V. K., “Efficient mixed pixel resolution using bio-inspired heuristics”, **IEEE Publications**, International Conference on Optimization, Reliability, and Information Technology (ICROIT), Faridabad, Haryana, pp. 364-368, 2014.
- [167] Velayutham P., Chandrasekaran S., Thulukkanam S., Jitendra K., Jain, D., “*Membrane Computing Model for Software Requirement Engineering*”, In: 2nd IEEE International Conference on Computer and Network Technology, Bangkok, pp. 487 – 491, 2010.
- [168] Industrial Application, Wuhan, vol. 1, pp. 566-570, 2008.
- [169] Wang, D., Wu X. B., “*Particle Swarm Intelligence Classification Algorithm for Remote Sensing Images*”, In: IEEE Pacific-Asia Workshop on Computational Intelligence and Wegener, A., “*The Origin of Continents and Oceans*”, Dover Publications, 2011.
- [170] Whitley, D., Hains, D., Howe, A., “*A Hybrid Genetic Algorithm for the Traveling Salesman Problem using Generalized Partition Crossover*”, In: proceedings of the 11th international conference on parallel problem solving from nature (PPSN XI), Part I, LNCS 6238, pp. 566-575, 2010.
- [171] Williams, K. J., et al., “*Which environmental variables should I use in my biodiversity model?*”, International Journal of Geographical Information Sciences, Special Issue:

- Second Special Issue on Spatial Ecology, Taylor and Francis, vol. 26(11), 2009-2047, 2012.
- [172] Wong, L. P., Yoke, M., Low, H., Chong, C. S., “*Bee Colony Optimization with Local Search for Traveling Salesman Problem*”, In: Proceedings of 7th IEEE International Conference on Industrial Informatics (INDIN), Daejeon, pp. 1019 – 1025, 2008.
 - [173] Wulder, M. A., Franklin, S. E., White, J. C., Linke, J. and Magnussen, S., “*An accuracy assessment framework for large-area land cover classification products derived from medium-resolution satellite data*”, International Journal of Remote Sensing, Taylor and Francis, vol. 27(4), pp. 663-683, 2006.
 - [174] Xiaoping, L., Xia, L., Lin, L., Jinqiang, H. and Bin, A., “*An Innovative Method to classify Remote-Sensing Images using Ant Colony Optimization*”, IEEE transactions on geosciences and remote sensing, vol. 46(12), pp. 4198-4208, 2008.
 - [175] Yang, X.S., “*Nature-inspired meta-heuristic algorithm*”, UK: Luniver Press, 2008.
 - [176] Yang, X. S., Deb, S., “*Engineering Optimization by Cuckoo search*,” International Journal of Mathematical Modeling and Numerical Optimization, Inderscience Publications, vol. 1(4), pp 330-343, 2010.
 - [177] Yang, X.S. “*Firefly algorithm, stochastic test functions and design optimisation*”, International Journal of Bio-inspired Computation, vol. 2(2), pp. 78–84, 2010.
 - [178] Yao, X., Liu, Y. and Lin, G., “*Evolutionary programming made faster*,” IEEE Transactions on Evolutionary Computation, vol. 3(2), pp. 82–102, 1999.
 - [179] Yasar, G. S. and Tuncay, E., “*Detection of Hidden Hostile/Terrorist Groups in Harsh Territories by Using Animals as Mobile Biological Sensors*”, In: Sensors 2008, vol. 8(7), pp. 4365-4383, 2008.
 - [180] Ying L., Cheng, B., “*An Improved k-Nearest Neighbor Algorithm and Its Application to High Resolution Remote Sensing Image Classification*”, In: 17th IEEE International Conference on Geoinformatics, Fairfax, VA, pp. 1-4, 2009.
 - [181] Zadeh, L.A., “*Fuzzy sets*,” Information and Control, vol. 8(3), pp.338- 353, 1965.
 - [182] Zhengra, H., Chengjian, W., Bingyao, J., Wenjiang, P. and Luxi, Y., “*A New Population-based Incremental Learning Method for the Travelling Salesman Problem*”, In: Proceedings of the 1999 Congress on Evolutionary Computation (CEC), vol. 2, pp. 1152-1156, 1999.
 - [183] Zhifeng, H., Huang, H. and Cai, R., “*Bio-inspired Algorithms for TSP and Generalized TSP*”, Book: Travelling Salesman Problem, ITech, Vienna, Austria, ISBN 978-953-7619-10-7, pp. 202, 2008.
 - [184] Zhou, G., Wei, D., “*Survey and Analysis of Land Satellite Remote Sensing Applied in Highway Transportations Infrastructure and System Engineering*”, In: IEEE International

- on Geoscience and Remote Sensing Symposium (IGARSS), Boston, MA: IEEE, vol. 4, pp. 479 – 482, 2008.
- [185] Zimmermann, N. E., Edwards Jr., Thomas C., Graham, C. H., Pearman, Peter B. and Svenning, Peter B., “*New trends in species distribution modeling*”, Ecography, Special issue on Species Distribution Modeling, Wiley, vol. 33(6), pp. 985 – 989, 2010.
- [186] Open Source Tool for Hybrid ACO2/PSO rule extraction
"<http://www.PSOACO2.sourceforge.net.com/>"

Biography of Author



- Languages known: **English, Hindi, German.**
- Professional Experience: **Oracle India Pvt. Ltd.**, Member Technical Staff, July 2010-January 2011 at a very high pay package (awarded as the ***best campus placement***).
- Academic Qualifications: **Ph.D.** with **first rank** in PhD coursework.
- **M.E.** (Computer Engg.) from DCE with **Distinction** and 82% marks (**University Topper**)
- **B.Tech** (Computer Sc. & Engg.) from JSSATE, UPTU with 78% marks (**Batch Topper**)
- **XIIth** (ISC Board) with 90% marks (**Second Topper and Gold Medalist with 100% marks in Mathematics and highest % in Chemistry**)
- **Xth** (ICSE Board) with 89.2% marks.

Research Experience

- **Hold a practical research experience** working at Defense & Terrain Research Lab (DTRL), Defense & Research Development Organization (DRDO), for a duration of **an year** while in Masters and **3.5 years** since joining Ph.D. program, a total of **4.5 years** as a research trainee (**January 2011 – June 2014 + August, 2009 – July, 2010**).
- Undergone a practical training for a period of 2 months in the field of NETWORKING in BHEL, Haridwar (**June, 2005 - August, 2005**).

Teaching Experience

Hold a teaching experience of 4.5 years during **Ph.D.** in January 2011-June 2014 and after that as an **Assistant Professor** till date. Taught various subjects: Artificial Intelligence, Soft Computing, Digital Image Processing and Fundamentals of Programming in c++ to B. Tech Final year and third year students in Delhi Technological University on recommendation by the Head of Department, Dept. of Computer Engineering, DTU. Also, during M.E., she has taken practical classes of courses like Operating Systems, Compiler Design, Java, Software Engineering and C++.

Awards and achievements

As presented in Section 2(A) in ‘PUBLICATIONS AND AWARDS FROM THE THESIS’.

Lectures delivered and invited talks

As presented in Section 2(B) in ‘PUBLICATIONS AND AWARDS FROM THE THESIS’.

Publications and projects undertaken

As presented in Section 1(A-E) in ‘PUBLICATIONS AND AWARDS FROM THE THESIS’

Summary

Lavika Goel is an Assistant Professor at SRM University, NCR Campus and worked as Assistant Professor (contractual) at Delhi Technological University (DTU). She worked as a research scholar in the Department of Computer Science & Engineering at DTU, formerly Delhi College of Engineering, Delhi, India. She worked at **Oracle India Private Ltd.** for six months through campus recruitment and then left it to pursue research in the field of Nature Inspired Intelligence, her interest area. Securing a **GATE percentile of 99% and an AIR rank of 242**, she did her M.E. (Master of Engineering with *Distinction*) in Computer Technology & Applications from Delhi College of Engineering, Delhi and was the **top rank holder of M.E. Batch 2008-2010**. She was offered six months internship from **ST Microelectronics** and a dual campus recruitment by **Oracle India Pvt. Ltd.** as well as **Aricent Technologies**. She did her B-Tech (with *Honours*) in Computer Science & Engineering from JSS Academy of Technical Education, Noida and also achieved the highest percentage of marks there. She was offered recruitment by **HCL Technologies** as a part of B.Tech Campus Recruitment Program and had also cleared the **Microsoft** Off-Campus recruitment written examination. She was the **Second topper** and **Gold medalist** with **100% marks** in **Mathematics** and **highest % in Chemistry** in 12th Std. ISC board.

Holding varied research interests, she has published numerous research papers in international journals and conferences including **Springer, IEEE, Elsevier, Taylor and Francis**, etc. She holds student membership of the **Machine Intelligence & Research (MIR) Labs** and serves as reviewer to many reputed international journals including the **Elsevier Journal of Information Sciences (Impact factor 3.7)** and the **Elsevier Journal of Applied Soft Computing (Impact factor 2.5)**. Her research interest is in the area of Artificial Intelligence particularly dedicated to the areas- nature inspired intelligence, soft computing, hybrid intelligent systems, cryptography and remote sensing. She is currently working on the development of optimization techniques inspired from nature inspired intelligence.