Lecture Notes in Electrical Engineering 696

Navid Razmjooy
Mohsen Ashourian
Zahra Foroozandeh *Editors*

Metaheuristics and Optimization in Computer and Electrical Engineering

Springer

Lecture Notes in Electrical Engineering

Volume 696

The book series *Lecture Notes in Electrical Engineering* (LNEE) publishes the latest developments in Electrical Engineering - quickly, informally and in high quality. While original research reported in proceedings and monographs has traditionally formed the core of LNEE, we also encourage authors to submit books devoted to supporting student education and professional training in the various fields and applications areas of electrical engineering. The series cover classical and emerging topics concerning:

- Communication Engineering, Information Theory and Networks
- Electronics Engineering and Microelectronics
- Signal, Image and Speech Processing
- Wireless and Mobile Communication
- Circuits and Systems
- Energy Systems, Power Electronics and Electrical Machines
- Electro-optical Engineering
- Instrumentation Engineering
- Avionics Engineering
- Control Systems
- Internet-of-Things and Cybersecurity
- Biomedical Devices, MEMS and NEMS

For general information about this book series, comments or suggestions, please contact leontina.dicecco@springer.com.

To submit a proposal or request further information, please contact the Publishing Editor in your country:

China

Jasmine Dou, Associate Editor (jasmine.dou@springer.com)

India, Japan, Rest of Asia

Swati Meherishi, Executive Editor (Swati.Meherishi@springer.com)

Southeast Asia, Australia, New Zealand

Ramesh Nath Premnath, Editor (ramesh.premnath@springernature.com)

USA, Canada:

Michael Luby, Senior Editor (michael.luby@springer.com)

All other Countries:

Leontina Di Cecco, Senior Editor (leontina.dicecco@springer.com)

**** Indexing: Indexed by Scopus. ****

More information about this series at http://www.springer.com/series/7818

Navid Razmjooy · Mohsen Ashourian ·
Zahra Foroozandeh
Editors

Metaheuristics and Optimization in Computer and Electrical Engineering

Springer

Editors
Navid Razmjooy
Tafresh University
Tafresh, Iran

Zahra Foroozandeh
Faculty of Engineering
University of Porto
Porto, Portugal

Mohsen Ashourian
Department of Electrical Engineering
Islamic Azad University (IAU)
Majlesi Branch
Isfahan, Iran

ISSN 1876-1100 ISSN 1876-1119 (electronic)
Lecture Notes in Electrical Engineering
ISBN 978-3-030-56688-3 ISBN 978-3-030-56689-0 (eBook)
https://doi.org/10.1007/978-3-030-56689-0

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Optimization is an important and decisive activity in science and engineering. Engineers will be able to produce better designs when they can save time and decrease the problem complexity with optimization methods. Many engineering optimization problems are naturally more complex and difficult to solve by conventional optimization methods such as dynamic programming. In recent years, more attention has been paid to innovative methods derived from the nature that is inspired by the social or the natural systems, which have yielded very good results in solving complex optimization problems. Metaheuristic algorithms are a type of random algorithm which is used to find the optimal solutions. Metaheuristic algorithms are able to find good (near-optimal) solutions for complicated optimization problems in a short time. Metaheuristics are one of the approximate types of optimization algorithms that can better escape from the local optimum points and can be used in a wide range of problems in engineering.

Recently, several categories of metaheuristic algorithms have been developed. To date, there has been no clear and comprehensive definition of the term metaheuristic, and various definitions have been proposed for this term. But in short, the main characteristics of metaheuristic methods can be expressed as follows:

- The main purpose of these methods is to effectively and efficiently search for the solution space instead of simply finding the optimal or near-optimal solutions.
- Metaheuristic algorithms are policies and strategies that guide the search process.
- Metaheuristic algorithms give approximate and often random solutions.
- These methods may use mechanisms to prevent the search process from being trapped in local optimizations.
- Metaheuristic algorithms are not dependent on the type of problem; in other words, they can be used to solve a wide range of optimization problems.

In short, it can be said that metaheuristic algorithms are advanced and general search strategies and suggest steps and criteria that are very effective in escaping from the trap of local optimizations. Therefore, the main contribution of this book is to indicate the advantages and importance of metaheuristics in computer and electrical engineering applications.

Tafresh, Iran Navid Razmjooy

Contents

Preface

Navid Razmjooy

Abstract Meta-heuristics are one of the most significant decision-making methods concerning especially in the fields of computer and electrical engineering. Most of the real-world optimizations are highly multimodal and nonlinear, under various complex constraints. Different objectives are often conflicting. Even for a single objective, sometimes, optimal solutions may not exist at all. In general, finding an optimal solution or even sub-optimal solutions is not an easy task.

Keywords Optimization · Meta-heuristics · Bio-inspired algorithms · Computer and electrical systems

Optimization techniques are one of the most important decision-making areas concerning industrial, economics, business, and financial systems. Most real-world optimizations are highly multimodal and nonlinear, under various complex constraints. Different objectives are often conflicting. Even for a single objective, sometimes, optimal solutions may not exist at all. In general, finding an optimal solution or even sub-optimal solutions is not an easy task. Metaheuristic methods are a crucial aspect of any complex algorithm, and a very active area of research is the design of nature-inspired metaheuristics. Many recent metaheuristics, especially evolutionary algorithms, are inspired by natural systems to solve complex engineering problems in a reasonable amount of time. Computer and electrical systems analysis and operation are no exception to this. Whoever has implemented a complex engineering system, knows well that the selection of the initial guess and the convergence criterion are based on heuristics. The objective of the book is to explore the emerging meta-heuristics optimization algorithms and methods in computer and electrical engineering applications. The prospective audience of the book is engineers, researchers, scientists, as well as industrialists. Chapters 1–15 are each devoted to a separate algorithm—and they each start with a brief literature review of the development of the algorithm, and then explain how it has been used for a real-life problem.

N. Razmjooy (✉)
Tafresh, Iran
e-mail: navid.razmjooy@tafreshu.ac.ir

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_1

Jaya Algorithm and Applications: A Comprehensive Review

Essam H. Houssein, Ahmed G. Gad, and Yaser M. Wazery

Abstract All the population-based optimization algorithms are probabilistic algorithms and require only a few control parameters including number of candidates, number of iterations, elite size, etc. Besides these control parameters, algorithm-specific control parameters are required by different algorithms. Either the computational effort is increased or the local optimal solution is yielded as a result of the improper tuning of algorithm-specific parameters. Thus, the algorithm which works without any algorithm-specific parameters such as Jaya Algorithm (JA) is widespread among the optimization applications and researchers. JA can be easily implemented based on not having to tune any algorithm-specific parameters. In order to prove that JA could be applied to every problem arising in practice, this study presents a comprehensive review of the advances with JA and its variants. On the other hand, the performance of JA was evaluated to solve the Himmelblau function against five well-known optimization algorithms. Therefore, this study is expected to highlight the JA's capabilities and performances especially for those researchers who are eager to explore the algorithm.

Keywords Population-based optimization algorithms · Jaya algorithm · Himmelblau function · Optimization · Algorithm-specific parameters

1 Introduction

Optimization methods aims to find the maximum or minimum optimal value for a given function. To optimize a function, it must have its own behavior, domain, and a specific number of variables, named design variables. Indeed, some functions with local minima can lead to the difficulty of obtaining an absolute optimal value [1–3].

E. H. Houssein (✉) · Y. M. Wazery
Faculty of Computers and Information, Minia University, Minia, Egypt
e-mail: essam.halim@mu.edu.eg

A. G. Gad
Faculty of Computers and Information, Kafrelsheikh University, Kafrelsheikh, Egypt

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_2

Deterministic and heuristic approaches are the two main categories of optimization methods. A function's analytical properties are exploited in deterministic approaches [4]. Obtaining the global optimum may become a very complex task in non-convex or large-scale optimization problems [5, 6]. Therefore, heuristic approaches should be used due to their flexibility and efficiency over deterministic ones as well as the high ability to reduce the total computational time required to determine the optimum [7, 8].

Recently, metaheuristic optimization methods are mainly inspired by the evolution theory or a swarm behavior. Parameter-tuning like number of iterations, population size, elite size, etc. is required by all evolutionary or swarm-based algorithms. Moreover, various algorithms may need tuning of their own algorithm-specific parameters. The improper tuning of these parameters may adversely affect the performance of these algorithms. Tuning the number of iterations and population size is also required [9, 10].

Over the past few years, different metaheuristic optimization algorithms [11] have been used in several approaches, such as the Particle Swarm Optimization (PSO) [12], Genetic Algorithm (GA) [13], Artificial Bee Colony (ABC) algorithm [14], Bat Algorithm (BA) [15], Differential Evolution (DE) [16], Whale Optimization Algorithm (WOA) [17], Elephant Herding Optimization (EHO) [18], Grasshopper Optimization Algorithm (GOA) [19], Harris Hawk Optimizer (HHO) [20–22], and Henry gas solubility optimization [23, 24]. These algorithms have achieved a high performance by effectively optimizing the system parameters. However, the computational cost optimization of metaheuristic algorithms is still an ongoing research point that needs further improvement in many areas.

A recently developed algorithm called Jaya, first proposed by Rao [25], takes the advantage of simplicity and ease to apply in addition to not requiring any algorithm-specific control parameters. As a result, it has been successfully adapted to solve benchmark global optimization functions, power flow problems [26] as well as travelling salesman's problem.

Being parameter-less is the major feature of the JA. Due to this single attribute, a user will not need parameter-tuning to get acceptable output. However, controlling the algorithm-specific control parameters of JA does not look easy. For example, it was found that tuning the parameters in each iteration is not easy and time-consuming [27]. Furthermore, because JA is a newly developed algorithm, it is limited in its application. It stays to be perceived how the algorithm contributes to diverse fields of research [28].

Jaya is a population-based heuristic algorithm that operates similarly to other optimization techniques [29, 30]. The foremost goal of this study is to introduce an updated review of research of JA hybridization, improved, and variants. Also, this review determines the ability of JA to competitively solve the Himmelblau function as compared to other five different optimization algorithms. Detailed and comprehensive comparisons can be carried out using this test suite because of the huge amount of data available in literature. To improve the convergence rate, the original JA formulation is exploited to reduce the number of iterative evaluations required in the optimization.

The results of the JA are found competitive to the best solutions available in other cutting-edge metaheuristic computational methods including variants of BA, GA, DE, PSO, And CS according to their efficient optimization task. The performance of JA is investigated in terms of minimum weight, standard deviation and number of iterative evaluations required in the optimization process. Over independent optimization runs, a statistical analysis shows superior convergence of the proposed algorithm in terms of the average, best, worst optimized weights and corresponding standard deviations. With no shadow of doubt, results proved the robustness, efficiency, and reliability of the JA as compared to other metaheuristic methods.

The rest of the paper is organized as follows: an overview of JA variants in various fields of engineering optimization is presented in Sect. 3. The structure of JA is generally described in Sect. 2. Experiments and corresponding results are discussed in Sect. 4. Finally, Sect. 5 concludes this review.

2 Jaya Algorithm (JA)

For the purpose of solving constrained and unconstrained optimization problems, Rao has developed a novel swarm-based heuristic; Jaya algorithm [31]. Jaya is a Sanskrit word meaning "**Victory**". The algorithm is working based on that the searching agents or the particles continuously attempt to get closer to the target (i.e., victory) by keeping away from the worst solution (i.e., failures) in each iteration. Depending on this mechanism, the searching agent updates basically its position by keeping only the best position as well as precisely ignoring all worst positions [25]. Hence, by updating all candidates of JA in each iteration, all solutions resulting from iteration are higher than the previous worst solution. In JA, every proposed solution or searching agent is known as a particle ($X_{j,k,i}$), every particle is seeking the most effective solution (J_{Best}) and keeps away from the worst solution (J_{Worst}) of the objective or cost function (J), in the search area. Figure 1 shows the flowchart of the JA.

This is achieved to optimize the objective function (J), supposing 'n' of candidates (i.e., $k = 1, 2, ..., n$) and 'm' of design variables or generators (i.e., $j = 1, 2, ..., m$). The positions of the particles are mathematically updated as (1) [25]:

$$X'_{j,k,i} = X_{j,k,i} + r_{1,j,i}(X_{j,best,i} - |X_{j,k,i}|) - r_{2,j,i}(X_{j,worst,i} - |X_{j,k,i}|) \quad (1)$$

where, $X_{j,k,i}$ represents jth variable for the kth candidate during the ith iteration. $r_{1,j,i}$ and $r_{2,j,i}$ are random numbers with values in the range [0, 1]. The best and worst candidate values are $X_{j,best,i}$ and $X_{j,worst,i}$, respectively. $X'_{j,k,i}$ are the new value of $X_{j,k,i}$. $X'_{j,k,i}$ value is kept if it returns the best fitness value compared to $X_{j,k,i}$. At the end of each iteration, the cost function values are maintained, and the next iteration considers these values as input for computation until the optimal solution is obtained.

The pseudocode of JA, outlined in Algorithm 1, is simple to understand.

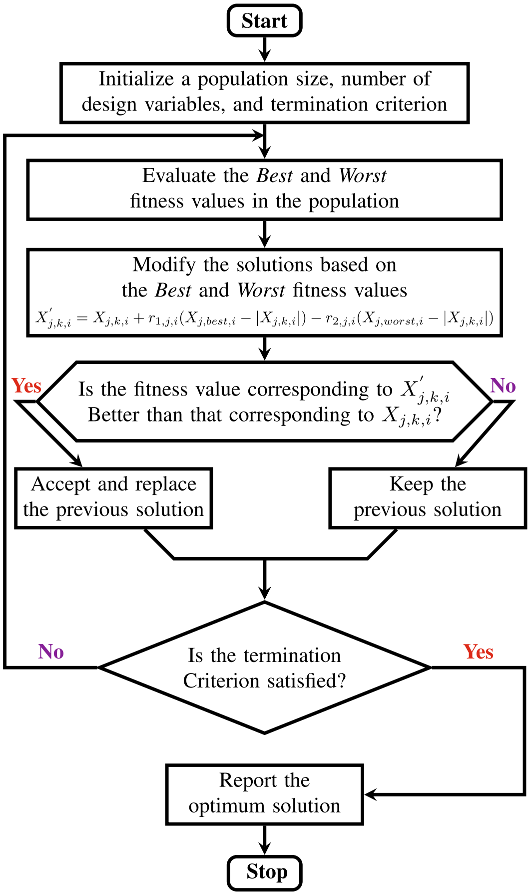


Fig. 1 JA flowchart

3 Variants of Jaya Algorithm

Teaching Learning Based Optimization (TLBO), a powerful heuristic algorithm that is free from algorithm-specific parameters, was introduced by Rao et al. [32]. Optimization researchers have widely used the TLBO algorithm because of its effectiveness and wide popularity [33, 34]. In [35], mathematical models were optimized by Rao et al. optimized in terms of input-output process parameters of die casting, continuous casting, and squeeze casting processes using the recently developed TLBO algorithm. The TLBO algorithm has achieved a big success that led to another parameter-less algorithm that has been very recently presented by Rao [25]. However, the newly presented algorithm is mono-phase and affordable to apply. The new algorithm is defined as Jaya, and a range of constrained and unconstrained benchmark functions has been effectively solved using it [25]. All the case studies considered in

Algorithm 1 JA pseudo code.

Input:
n – Population size or number of candidates
m – Number of design variables or generators
Z – Maximum number of iterations
X – Population

Output:
J_{Best} or J_{min} – Best solution of the cost or fitness function (J)
J_{Worst} – Worst solution of the objective or fitness function (J)
X_{best} – Best candidate of the fitness function
X_{worst} – Worst candidate of the fitness function

Start
  for $k := 1$ to n **do** ▷ (i.e., population size)
    for $j := 1$ to m **do** ▷ (i.e., design variables)
      Initialize population ($X_{i,k}$)
    end for
  end for
  Evaluate $X_{best,1}$ and $X_{worst,1}$ ▷ (Based on the solution of the fitness function)
  Set $i = 1$ ▷ (Initialize iteration number)
  while maximum number of iterations (Z) is not met **do**
    for $k := 1$ to n **do** ▷ (i.e., population size)
      for $j := 1$ to m **do** ▷ (i.e., design variables)
        Set $r_{1,j,i}$ = a random number from [0, 1]
        Set $r_{2,j,i}$ = a random number from [0, 1]
        Update Eq. 1
      end for
      if solution $X'_{j,k,i}$ is better than $X_{j,k,i}$ **then** ▷ (Based on the solution of the fitness function)
        Set $X_{j,k,i+1} = X'_{j,k,i}$
      else
        Set $X_{j,k,i+1} = X_{j,k,i}$
      end if
    end for
    Set $i = i + 1$
    Update $X_{best,1}$ and $X_{worst,1}$
  end while ▷ maximum iterations or termination criterion satisfied
End

diverse casting processes has proven the higher convergence speed of JA as compared to the TLBO algorithm [36].

The literature is rich enough in addressing variety methods of JA. Generally, the optimal solution of the search optimization problems is obtained by tuning the system parameters in the flexible intelligence algorithms [36–39]. Both usual non-linearity and non-differentiability of practical systems (i.e., power system, aerospace, robotics, etc.,) make them highly complex [38–40]. In these scenarios, to boost the performance of searching and obtaining the optimal solution, the researchers have utilized algorithm-specified parameters in the optimization algorithms. The inertia weight parameter in PSO [41] and temperature parameter in simulated annealing [42] are such few examples. In complex problems, the absence of tuning the algorithm-specific parameters may cause JA not to converge faster.

Rao et al. [43] developed Multi-Objective Jaya (MO-Jaya) algorithm that: (1) does not need the algorithm-specific parameters to be tuned by the user, and (2) is implemented simply since, in single phase, a single equation is used to update the solutions with low computational cost. To give a high convergence speed to MO-Jaya, simultaneously, the best and worst solutions are considered in the current population to avoid trapping into local optima. Based on the fact that the solutions are relationally non-dominant and with the aid of the ranking mechanism, MO-Jaya algorithm guides the search process towards the Pareto-optimal set and maintains the good solutions in every generation. Accordingly, the process engineer can use the Pareto-efficient solutions provided by MO-Jaya algorithm as ready reference in order to set the optimal levels of parameter values for best performance of electro-discharge machining, micro-electro-discharge machining, and plasma arc machining processes with sustainability.

To solve the Large-scale Urban Traffic Light Scheduling Problem (LUTLSP), Gao et al. [44] proposed a centralized Mixed Integer Linear Programming (MILP) formulation. For all vehicles, the aim was to minimize the total delay time of the network-wise in a time window.A hybrid of three algorithms, including Water Cycle Algorithm (WCA), Harmony Search (HS), and JA was successfully utilized to solve the LUTLSP in real time. In experiments with real data in Singapore, the reported optimizers were used to solve 11 cases with different numbers of intersections. In order to optimally solve the LUTLSPNew discretization, strategies and improvements in terms of the search mechanism were proposed and applied to the WCA and the native JA.

In the field of electrical engineering, Singh et al. [45] used a two-area interconnected power system to build a Jaya-based controller for Automatic Generation Control (AGC). The controller was designed based on three different design objectives; tie-line power, the settling times of deviations in frequencies, and Area Control Errors (ACEs); Integral Time Multiplied Absolute Error (ITAE), and peak overshoots of the deviations of the frequencies. To form a single objective problem, these different objectives were combined using a decision-making tool known as Analytic Hierarchy Process (AHP). To test the superiority of the proposed controller, it was compared with five other controllers in consideration of different algorithms. The robustness of the proposed controller was established based on five diverse cases of different sets of disturbances ranging from light load to heavy load changes in the areas. Based on a comparative statistical analysis, the numerical results showed the effectiveness and superiority of the proposed controller compared to others. For an AGC problem, the superior nature of the proposed controller was affirmed under different cases based on the deviations in the frequencies of the areas illustrated by the time-domain simulations.

In [46], a novel Elite Opposition-based (EO) Jaya algorithm was presented by Wang et al. for estimating model parameters of photovoltaic (PV) cells. The EO-Jaya algorithm, similar to the generic JA, was free of algorithm-specific parameters. For different PV models, to steadily and accurately estimate their parameters, another

Improved Jaya (I-Jaya) algorithm was proposed by Yu et al. [47]. In I-Jaya, a self-adaptive weight is utilized by the search process to adjust the tendency of avoiding the worst solution and approaching the best solution. The algorithm uses this weight to early approach the potential search region and later implement the local search. On the other hand, to efficiently solve the Maximum Power Point Tracking (MPPT) problem of PV systems, another novel model-free solution algorithm named as the natural cubic spline guided Jaya (S-Jaya) algorithm, was proposed by Huang et al. [48] under partial shading conditions. Comprehensive simulation experiments and studies were conducted to validate the feasibility and effectiveness of the proposed MPPT method. MPPT using the generic JA, MPPT using the S-Jaya algorithm, and well-tuned PSO algorithm performances were compared. Based on the results of the overall tracking efficiency, the convergence speed, and the oscillations in the convergence, the S-Jaya algorithm outperformed benchmarking algorithms. Thus, this finding supports that the proposed S-Jaya algorithm can be used more effectively and efficiently to track the global MPP (GMPP).

Rao et al. [49] proposed a Self-Adaptive Multi-Population Jaya (SAMP-Jaya) algorithm. To test and validate the performance of the proposed algorithm, the computational complex problems of the Congress on Evolutionary Computation (CEC) 2015 in addition to constrained and unconstrained benchmark functions were used. To find the average rank of the algorithm, the Friedman rank test was used, and it was observed that the other algorithms were located after the proposed approach. Moreover, the proposed method was used to optimize the design problem of a Plate-Fin Heat Exchanger (PFHE). The search mechanism of the JA was upgraded using the proposed method with the help of multi-population scheme which depends on the strength of change in solution to adaptively divide the population into number of sub-populations. An easy integration can be made between this multi-population-based scheme and advanced optimization algorithms that is based on single population. As compared to the recently reported optimization methods of the benchmark functions, the experimental results achieved by the SAMP-Jaya for the same problems ensured good agreement. Using the proposed approach, the number of entropy generation units of the PFHE ha been reduced by by 39.17, 27.49, and 8.11% compared to GA, PSO, and CS. Similar to the native Jaya, the SAMP-Jaya algorithm is simple and easy to implement, and it does not require any algorithmic-specific parameters to be tuned. Therefore, the complicated engineering optimization problems with a number of design parameters can be solved by SAMP-Jaya.

To solve diverse Multi-Objective Optimal Power Flow (MOOPF) problems, Warid et al. [50] proposed two novel Jaya-based methods; Modified Jaya (MJaya) and Quasi-Oppositional Modified Jaya (QOM-Jaya) algorithms. The IEEE 30-bus system was used to investigate the effectiveness of the suggested approaches. The results revealed the potential and applicability of the proposed algorithms to solve different MOOPF problems. To measure the reliability of the proposed algorithms in solving the MOOPF problem, a comparison has been held between the solutions obtained by the considered multi-objective optimization cases and those from other heuristic optimization algorithms in the literature. The proposed MJaya and QOM-Jaya algorithms has proved distinction and outperformance over diverse aforesaid algorithms in producing better well-distributed Pareto optimum fronts and optimal solutions. Furthermore, the proposed QOM-Jaya algorithm ensured faster convergence to the global Pareto set than did the MJaya and the considered well-known Neighborhood Knowledge-based Evolutionary Algorithm (NKEA). Hence, the proposed MJaya and QOM-Jaya algorithms represent fast, trustworthy, and efficient optimization tools to solve many MOOPF problems. In practical power systems, the MOOPF problems can be solved reliably using the proposed QOM-Jaya algorithm in terms of fast convergence and best solution.

Over the few last years, variants to the JA have been proposed, contributing in reaching the scientific application areas. In [51], Jaya and Quasi-Oppositional Jaya (QO-Jaya) algorithms were used to solve optimization problems of Gas Tungsten Arc Welding (GTAW), Friction Stir Welding (FSW), Electron Beam Welding (EBW), and Submerged Arc Welding (SAW) processes. Different case studies of optimization were discussed separately and the results of generic Jaya and QO-Jaya algorithms were compared to those obtained by other popular optimization algorithms such as TLBO, SA, and GA. As compared to JA, the use of quasi oppositional-based concept contributed in a higher convergence speed of QO-Jaya algorithm. However, the QO-Jaya algorithm required slightly higher computational time than the JA. The Jaya and QO-Jaya algorithms may be implemented to solve the optimization problems of other joining processes such as tungsten inert gas welding, laser welding, shielded metal arc welding, and flux-cored arc welding.

As a matter of urgency, heat flux from electronic devices need to be dissipated, so Rao et al. [52] tried to optimize the dimensions of Micro-Channel Heat Sink (MCHS) with using JA. MCHS was presented in two case studies in order to validate the performance of JA with TLBO and hybrid Multi-Objective Evolutionary (MOE) algorithms. Both the case studies considered pumping power and thermal resistance as the objective functions. Case study 1 expressed design variables in micro-channel depth, width, and fin width, while case study 2 expressed design variables in micro-channel depth, width at bottom and top, and fin width. In case study 1, It was observed that the JA competes the TLBO algorithm in terms of equal or better optimal results, and competes the hybrid MOE algorithm in terms of better results. In case study 2, JA obtained better optimal results than those obtained by TLBO and hybrid MOE algorithms. In case study 2, the JA has outperformed the TLBO algorithm. Also, JA obtained better Pareto optimal solutions than those obtained by TLBO and hybrid MOE algorithms.

In [53], the elitist version of JA (EJA) was proposed by Rao et al. for the purpose of optimizing the design of the shell-and-tube heat exchangers. Three constrained multi variable heat exchanger design problems were used to investigate the performance of EJA aiming to minimize the total annual cost. Different combinations of commonly used control parameters (e.g., elite size, population size, and number of iterations) were considered to study their influence on JA.

In [54], Rao et al. presented four case studies for optimization of particular thermal devices viz; honeycomb heat sink, cooling tower, thermo-acoustic prime mover, and heat pipe. The case studies demonstrated that the selected thermal devices can be optimally designed by the application of the Jaya and self-adaptive Jaya algorithms. The

Jaya and self-adaptive Jaya algorithms has achieved results that were compared to those obtained by Grenade Explosion Method (GEM), Niched Pareto Genetic Algorithm (NPGA), Multi-Objective Genetic Algorithm (MOGA) technique for honeycomb heat sink, Leap-Frog Optimization Program with Constraints (LFOPC) technique for cooling tower, Response Surface Method (RSM) techniques for thermoacoustic prime mover, and TLBO techniques for heat pipe. The convergence behavior of the self-adaptive JA was found more efficient and closer to the global solution as compared to the generic Jaya and aforesaid algorithms. The function evaluations and the computational time proved the superiority of the self-adaptive Jaya algorithm over other optimization algorithms. Also, the design of other thermal devices may be conveniently optimized using the Jaya and self-adaptive Jaya algorithms. Furthermore, various optimization applications like power optimization of micro-scale power devices, structural optimization, data clustering, etc. may exploit the proposed algorithms.

Wang et al. [55] developed an improved system to recognize facial emotion over human images. The chosen algorithm is a hybrid of Stationary Wavelet Transform (SWT) entropy used to extract features, single hidden layer Feed-forward Neural Network (FNN) used as a classifier, and JA that has little requirement over hyperparameters and has higher ability of recognition than other popular methods in the literature which can ensure the classifier training to converge, and not to fall into local optimum points.

Based on the fact that reducing maintenance costs and prolonging the service life can be achieved using early damage detection strategy, Du et al. [56] extended the JA to solve the damage identification optimization problem in order to ensure the safety and serviceability conditions of existing structures. To define the damage assessment, the Multiple Damage Location Assurance Criterion (MDLAC) was combined with criteria based on the difference between the flexibility matrices of both test models and intact model to formulate the objective function, then, consecutively, the optimization problem was solved. Single and multiple damage scenarios of three various structures are numerically considered to evaluate the performance of the proposed damage identification method. Compared to two other objective functions under the accuracy of detecting the degree and location of damage, the higher effectiveness of the proposed hybrid objective function was clear. In spite of the robustness and efficiency of the numerical simulations of the proposed method in assessing structural damage, it should be pointed out that the study could not cover some aspects of the method due to both the improper noise simulation of real conditions and the simplicity of the structural damage model. Therefore, using crack model as an example of complex damage models needs further investigation to examine if the proposed JA is generally applicable.

Since Carbon Fiber-Reinforced Polymer (CFRP) composites are widespread in application, mostly in defense, aerospace, and automotive industries, Abhishek et al. [57] focused on the optimization of the characteristics of machining performance (i.e. surface roughness, resultant cutting force, and material removal rate) on trying to turn the CFRP (epoxy) composites. To determine the combinations of the process parameters, an L_9 orthogonal array design was adapted for experimentation to get performance of multiple characteristics with no need of higher number of experimental runs as compared to the experiment in full factorial design. Multi-responses were adaptively summarized as an equivalent single response called Multi-Performance Characteristic Index (MPCI) using Fuzzy Inference System (FIS). MPCI was implemented in a nonlinear regression model that has been considered as fitness function for optimization using GA, TLBO, Jaya as well as Imperialist Competitive Algorithm (ICA) algorithms. It was observed that the results provided by JA are reliable and computed in less time and efforts. The effectiveness of Jaya was also validated against the TLBO algorithm, and proved similar results. This aims at the possible application of the proposed optimization method (in combination with JA, nonlinear regression, and FIS) which could well be exploited to control the quality of any process/product offline.

For the first time ever, Degertekin et al. [58] applied the JA to topology layout, and sizing design of truss structures. In structural optimization literature, new optimization algorithms are commonly evaluated based on the benchmark examples of truss design problems. JA was validated on three sizing-layout problems having 81 design variables, as well as three sizing problems including an average-scale structure having 29 design variables and 200 elements, a fairly large-scale structure having a 942-bar tower optimized with 59 design variables, and a large-scale structure having a 1938-bar tower optimized with 204 design variables. Also, Using discrete sizing/layout variables to optimize the topology was developed. For almost all design examples, JA got the best design or was one of the best algorithms indicated in literature with a tiny weight penalty with respect to the global optimum.

In a new creative contribution, Zhang et al. [59] developed a system for Tea-Category Identification (TCI) to process images captured by a 3 Charge-Coupled Device (CCD) digital camera for the purpose of tea category determination. A novel image feature called fractional Fourier entropy was utilized, and proved its ability to effectively extract features from tea images. JA was combined with FNN to produce a novel classifier called Jaya-FNN that was used in TCI. TCI system used the least features, and outperformed other seven well-known algorithms in terms of overall meanwhile, Average Sensitivity Rate (ASR), and TCI system.

In another contribution, an integrated hybridized model comprising Extreme Learning Machines (ELM) with Jaya, TLBO, DE, and PSO was proposed by Das et al. to predict in advance exchange rate of USD to EURO and USD to INR data for 1, 3, 5, 7 and 15 days and 1 month [60]. By comparing the model with Functional Link Artificial Neural Network (FLANN) and Neural Network (NN) optimized with Jaya, TLBO, DE, and PSO, it was clearly observed that the model is able to predict the open price in addition to guiding the investor to invest in Forex market.

Migallón et al. [61] presented efficient parallel proposals of the JA. A hybrid Message Passing Interface (MPI)/Open Multi-Processing (OpenMP) algorithm was developed, exploiting inherent parallelism at two levels; upper level based on parallel shared memory platforms, and upper level based on distributed shared memory platforms. Good scalability results were obtained from algorithms, so a huge number of processes was ideally used by the hybrid algorithm. Parallel JAs were tested for memory platforms as heterogeneous, shared, and distributed, reaching good parallel performance without changing JA behavior. By using 60 processors, 30 unconstrained functions analyzed parallel performance, obtaining an up to 57.6x speed-up. JA has proven to be applicable in many science and engineering problems. Recently, to efficiently exploit cluster computing platforms, Migallón et al. [62] presented another parallel Jaya-based multi-level algorithm, where, the external layer of the JA is parallelized to exploit distributed-memory architectures (or multiprocessors). Moreover, in internal layers, two more levels of parallelization are added to exploit shared-memory architectures (or multicores). By using up to 120 and 135 processes, the proposed algorithm obtained 84% average efficiency, and the convergence was slightly accelerated with respect to the original JA. Considering that engineering problems are usually of high computational cost, the proposed algorithms could improve optimization performance by rapidly resolving these problems using low-power computing platforms, or supercomputing platforms.

4 Experimental Results and Discussion

A MATLAB code has been developed for the meta-heuristic optimization technique of JA to solve the Himmelblau function [63]. By executing the predefined number of iterations, this piece of code should output a text file containing three tab-delimited columns representing the values of design variables x_1 and x_2 and the objective function $f(x)$ values respectively. Then, the last line of code in the main function called 'Jaya' uses the output text file as a parameter to call a function that plots the relation between x_1, x_2, and $f(x)$ as exhibited in Fig. 2.

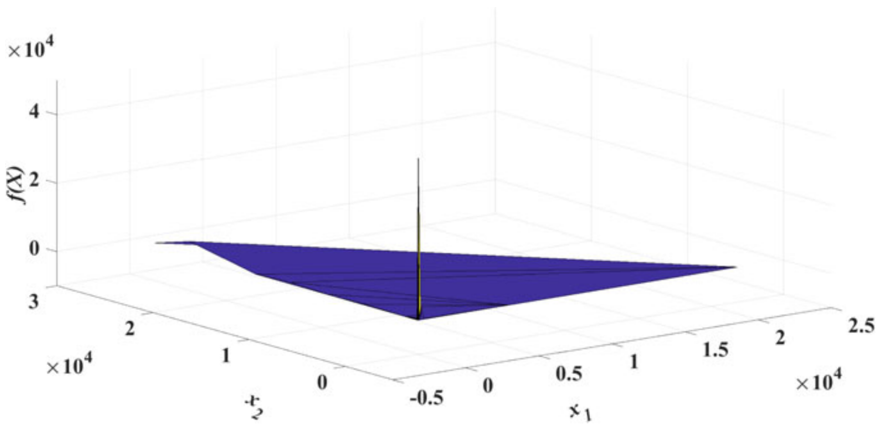


Fig. 2 3-D plot of the objective function $f(x)$ with respect to variables x_1 and x_2

4.1 Parameter Settings

BA, DE, PSO, GA, CS and Jaya are the algorithms used in our comparative study. Table 1 illustrates the parameter settings of these algorithms as used on conducting the experiment. PS and PM with respect to the GA algorithm refer to Parent Selection and Population Model, respectively. In fact, the GA used the tournament PS of a size two and an SS population model for tuning the parameters of the simplified Proportional Integral Derivative (PID) controller, where the two worst solutions are replaced by the two best offspring solutions.

Because the discussed algorithms are nature-inspired, the same population size of $n = 10$ is used and the same generation numbers of $MAX\_GEN = 10$ is employed in order to test their efficiency. Consecutively, all algorithms terminate when the maximum number of fitness function evaluations $MAX\_FE = 100$ is reached. To determine their stochastic nature, each one of the algorithms considered for the comparison was run 10 times. For each run, the values obtained from fitness function for both design variable, as well as the mean values, were recorded.

Table 1 Parameter settings of the competitive algorithms

Algorithm	Parameters	Value
Bat algorithm	Q	[0.5, 1.5]
	β	[0, 1]
	r_i	0.1
	A_i	0.9
Differential evolution	F	0.9
	CR	0.5
Particle swarm optimization	C_1	1.0
	C_2	1.0
	W	0.729
	v	[0, 2]
Genetic algorithm	P_c	0.8
	P_m	0.01
	PS	T = 2
	PM	SS
Cuckoo search	α	1.0
	γ	1.5
	p_a	0.1
	s	1.0
Jaya algorithm	r_1	[0, 1]
	r_2	[0, 1]
	i	10
	n	10

The performance of the proposed JA in solving the Himmelblau function has been found out by comparing the results of the popularly known swarm-based algorithms according to their efficient optimization task. Amongst them, the most popular ones are BA, DE, PSO, GA, CS.

4.2 Himmelblau Function and Benchmark Functions

In mathematical optimization, a multi-modal function, called Himmelblau function is accustomed to testing the effectiveness of optimization techniques [64]. The function can be defined by:

$$f(x, y) = (x^2 + y - 11)^2 + (x + y^2 - 7)^2 \quad (2)$$

The function can be defined on any input domain, but it is usually evaluated on $x_i \in [-6, 6]$ for $i = 1, 2$. Its single local maximum is located at $x = -0.270845$ and $y = -0.923039$ where $f(x, y) = 181.617$, and four identical minima are located locally at:

- $f(3.0, 2.0) = 0.0$,
- $f(-2.805118, 3.131312) = 0.0$,
- $f(-3.779310, -3.283168) = 0.0$,
- $f(3.584428, -1.848126) = 0.0$.

Analytically, all the minima can be located. However, writing the expressions in terms of roots makes them somewhat complicated because they are cubic polynomial radicals. David Mautner Himmelblau (1924–2011) introduced the function that is named after him [64]. Figure 3 illustrates the complete search space for the Himmelblau function.

By observing Fig. 4, we can conclude that the Himmelblau's function is continuous and not convex. Also, it is multimodal and defined on the 2-D space.

The performance of the proposed JA is tested on 7 well documented benchmark functions in the optimization literature including the Himmelblau function. These functions can be differentiated in terms of characteristics like regularity/non-regularity, separability/non-separability, unimodality/multimodality, etc. For each problem, there is difference in the number and ranges of design variables. Table 2 shows the 7 benchmark functions. The results obtained by JA are compared with those obtained by other optimization algorithms such as BA, DE, PSO, GA, and CS to validate the performance of the proposed JA. For diverse algorithms considered in the comparison, The identical function evaluations are maintained to provide a common platform. Thus, while comparing the performances of JA and other optimization algorithms, the consistency in the comparison is maintained. However, in general, the algorithm is said to be better as compared to the other algorithms if it requires a smaller number of function evaluations to converge to the same best solution. Accordingly, in this study, the maximum number of function evaluations was set

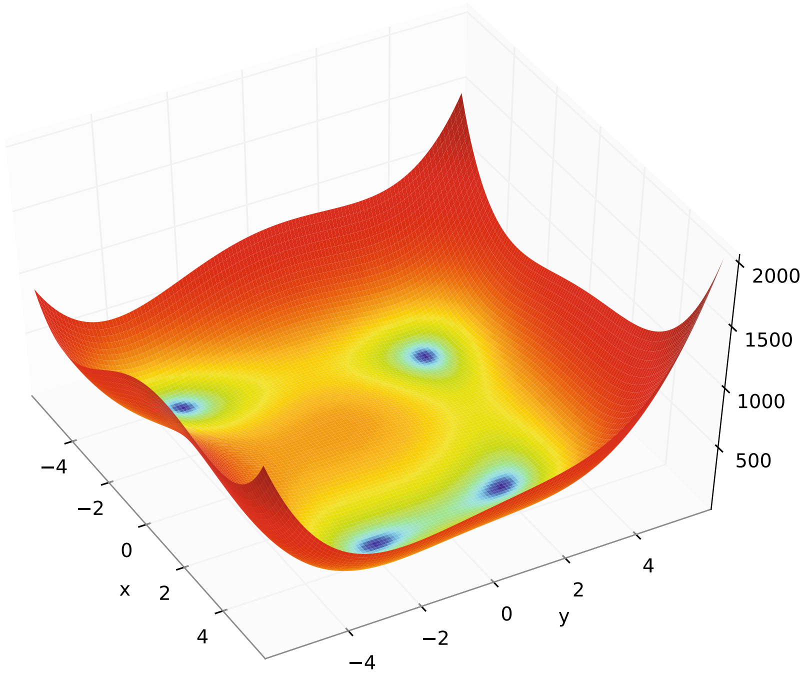


Fig. 3 Complete search space for the Himmelblau function

Fig. 4 Contour plot of the Himmelblau's function

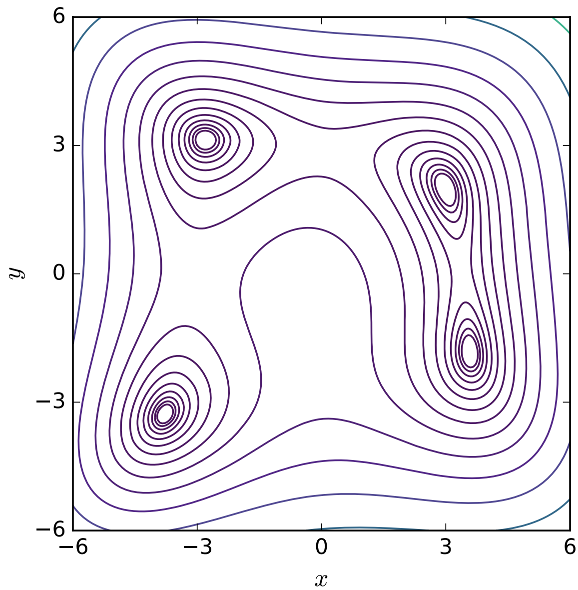


Table 2 Comparative performance of different benchmark functions

Functions	Statistical values	BA	DE	PSO	GA	CS	JA
Ackley	Best	0.00E+0	0.00E+0	0.00E+0	1.00E+0	0.00E+0	0.00E+0
	Worst	0.00E+0	0.00E+0	0.00E+0	0.00E+0	0.00E+0	0.00E+0
	Mean	0.00E+0	0.00E+0	0.00E+0	1.45E+01	0.00E+0	0.00E+0
	SD	0.00E+0	0.00E+0	0.00E+0	1.41E−06	0.00E+0	0.00E+0
Bohachevsky 1	Best	8.58E−06	9.72E−06	8.75E−06	8.63E−06	9.51E−06	**8.44**E**−06**
	Worst	8.79E−06	9.53E−06	8.90E−06	8.93E−06	9.35E−06	**8.57**E**−06**
	Mean	8.58E−06	9.27E−06	8.77E−06	8.60E−06	9.11E−06	**8.44**E**−06**
	SD	3.80E−08	3.95E−09	3.82E−09	3.83E−08	3.97E−08	**3.79**E**−08**
GoldStein–Price	Best	7.66E−05	8.38E−05	8.39E−05	7.39E−05	9.16E−05	**6.54**E**−05**
	Worst	7.17E−05	7.57E−05	7.58E−05	6.58E−05	10.7E−05	**6.82**E**−05**
	Mean	7.66E−05	8.39E−05	8.40E−05	5.87E+00	9.16E−05	**6.54**E**−05**
	SD	8.84E−07	0.00E+0	9.20E−07	1.00E+00	9.4E−07	**8.75**E**−07**
Langerman 5	Best	3.82E−07	0.00E+0	4.86E−07	3.96E−07	3.86E−07	**3.82**E**−07**
	Worst	3.82E−07	0.00E+0	4.86E−07	3.96E−07	3.86E−07	**3.82**E**−07**
	Mean	3.82E−07	0.00E+0	4.86E−07	2.88E−01	3.86E−07	**3.82**E**−07**
	SD	0.53E−08	0.00E+0	**0.51E−08**	5.25E−02	0.57E−08	**0.51**E**−08**
Michalewicz 2	Best	0.00E+0	0.00E+0	0.00E+0	−1.00E+0	7.22E−07	0.00E+0
	Worst	0.00E+0	0.00E+0	0.00E+0	0.00E+0	8.64E−07	0.00E+0
	Mean	0.00E+0	0.00E+0	0.00E+0	−1.80E+00	7.22E−07	0.00E+0
	SD	0.00E+0	0.00E+0	0.00E+0	0.00E+0	4.59E−08	0.00E+0
Perm	Best	6.48E−06	6.62E−06	6.63E−06	6.86E−06	7.22E−06	**6.41**E**−06**
	Worst	6.53E−06	6.67E−06	6.68E−06	6.88E−06	7.87E−06	**6.46**E**−06**
	Mean	6.48E−06	6.62E−06	6.63E−06	6.73E−01	7.71E−06	**6.41**E**−06**
	SD	0.38E−07	0.79E−07	0.39E−07	1.93E−01	0.49E−07	**0.34**E**−07**
Himmelblau	Best	1.05E−05	6.28E−05	7.43E−06	1.26E−14	0.00E+0	0.00E+0
	Worst	3.61E−01	5.37E−05	6.33E−06	1.38E−14	0.00E+0	0.00E+0
	Mean	7.45E−02	2.39E−05	4.65E−07	1.17E−13	2.60E−19	0.00E+0
	SD	7.45E−02	1.20E−05	1.63E−08	1.93E−01	5.35E−19	0.00E+0

as 500 for each benchmark function to provide a common experimental platform for the purpose of maintaining the consistency in the comparison of competitive algorithms. Just like other algorithms, suitable population sizes were chosen to execute the proposed JA 30 times for each benchmark function and for the same number of runs, the mean results obtained are compared with the other algorithms.

By executing the benchmark functions in Table 2, it was observed that the JA gives better performance on solving Himmelblau function than other algorithms using the above-mentioned test functions.

In order for the comparison to be fair, for each function in Table 2, the optimization problems have been solved using the same number of populations and iterations. For each algorithm, Table 1 states the values of algorithm-specific parameters. As given in Table 2, 50 population size and 500 number of iterations are taken to note the average minimum values (J_{min}) in terms of best, worst, mean, and Standard Deviation

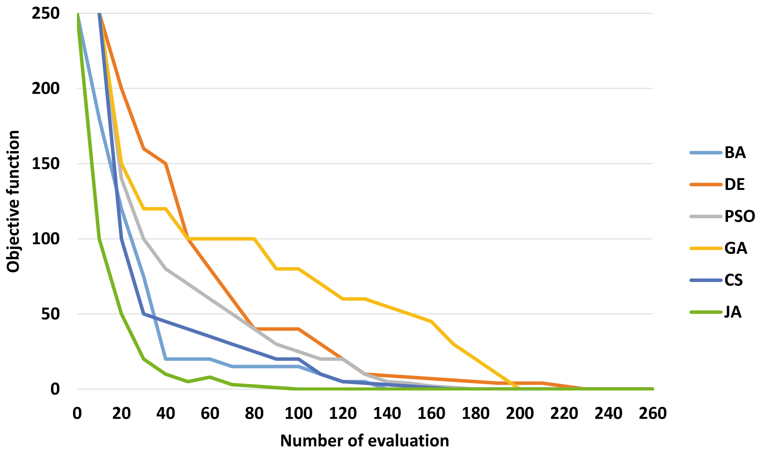


Fig. 5 Convergence characteristics of different algorithms for Bohachevsky-1 function

(SD) values of these functions by performing a total of 30 independent runs for all algorithms. Table 2 indicates that JA gives converge performance in terms of the best, worst, mean and SD as compared to BA, DE, PSO, GA, and CS. The bold mark in Table 2 mentions the minimum values of the aforementioned statistical variables.

Generally, if the optimization algorithm can converge too fast to the optimum value, then it is said to be effective. For 50 population size, the convergence graph of Bohachevsky-1, GoldStein–Price, and Himmelblau functions of their minimum values (J_{min}) against the number of evaluations are plotted in Figs. 5, 6, and 7 respectively. Figures 5 and 6 depict that the JA is relatively converging faster as compared to other above-mentioned algorithms, while Fig. 7 affirms that the JA is the fastest converge algorithm to solve the Himmelblau function. Moreover, other benchmark functions tabulated in Table 2 give similar observations.

4.3 Discussion

Fister et al. [38] have implemented the above-mentioned algorithms in MATLAB and Simulink Student Suite-Release 2014a. The outcomes are compared in terms of function value, processing time, convergence rate, and better results. For this purpose, the objective function had been taken as the Himmelblau function, and the different parameters like number of design variables, population size, max iterations, etc. are remaining constant.

Table 2 describes a comparative performance of the proposed JA with BA, DE, PSO, GA, and CS. In comparison with the BA, DE, PSO, GA, and CS, the JA follows the pliability in information flow by disregarding the biased culture (i.e., considering only the best and the worst).

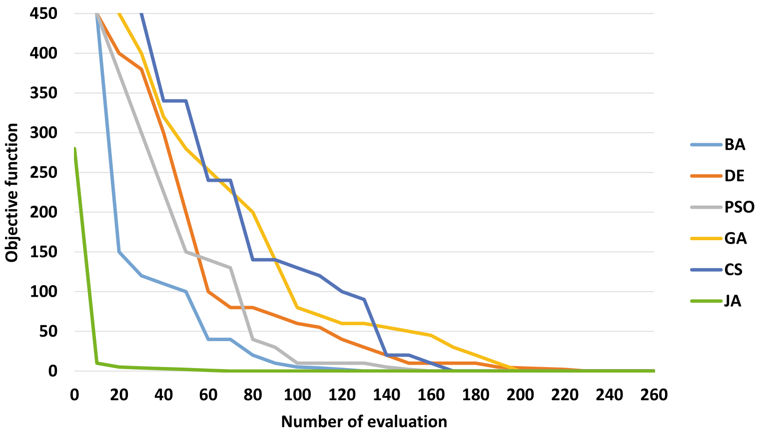


Fig. 6 Convergence characteristics of different algorithms for GoldStein–Price function

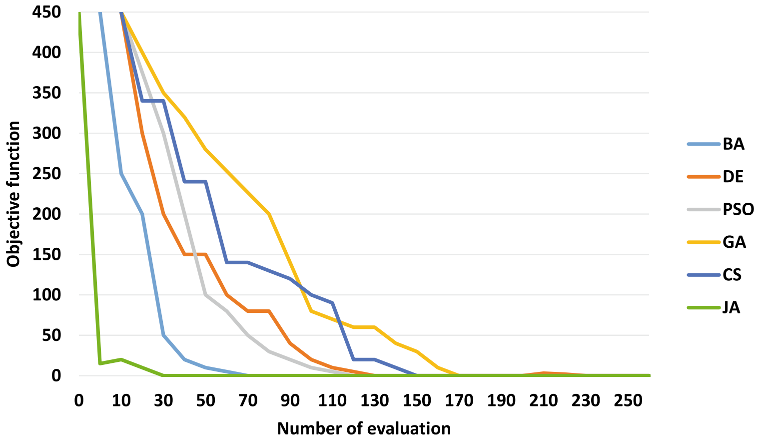


Fig. 7 Convergence characteristics of different algorithms for Himmelblau function

For each run, the best obtained fitness values after controlling for both design variables and corresponding mean values are presented in Table 3. In the ten independent runs, the statistical values (i.e. minimum, maximum, average, and SD) obtained are presented under "Total". bold indicates the best results.

As can be seen from Table 3, there are more or less similarity in the best results for the observed algorithms according to the maximum fitness value achieved because, except JA that obtained the maximum fitness value $f_{max}(y) = 0.9887$, which is above the interval, these curve within the interval $f_{max}(y) \in [0.9747, 0.9788]$. Taking into account the average values, the JA also achieved the best results (i.e., $f(\bar{y}) = 0.9652$), while the CS obtained the worst results (i.e., $f(\bar{y}) = 0.9652$). The fifth run juts the main reason for this laid, where the CS obtained the fitness value $f(y) = 0.9474$.

Table 3 Comparison of algorithms with regard to fitness

Run	Fitness	BA	DE	PSO	GA	CS	JA
1	Variable-1	0.9729	0.9754	0.9816	0.9804	0.9670	0.9310
	Variable-2	0.9726	0.9762	0.9846	0.9600	0.9744	0.9802
	Mean	0.9727	0.9758	**0.9831**	0.9702	0.9707	**0.9887**
2	Variable-1	0.9795	0.9760	0.9445	0.9729	0.9655	0.9765
	Variable-2	0.9497	0.9717	0.9810	0.9556	0.9773	0.9643
	Mean	0.9646	0.9738	0.9627	0.9642	0.9714	0.9711
3	Variable-1	0.9702	0.9777	0.9710	0.9726	0.9690	0.9732
	Variable-2	0.9847	0.9800	0.9893	0.9845	0.9804	0.9643
	Mean	0.9775	**0.9788**	0.9801	**0.9786**	**0.9747**	0.9814
4	Variable-1	0.9757	0.9754	0.9736	0.9455	0.9586	0.9765
	Variable-2	0.9802	0.9681	0.9918	0.9653	0.9807	0.9643
	Mean	**0.9780**	0.9717	0.9827	0.9554	0.9696	0.9732
5	Variable-1	0.9778	0.9722	0.9742	0.9638	0.9615	0.9712
	Variable-2	0.9592	0.9764	0.9750	0.9443	0.9333	0.9462
	Mean	0.9685	0.9743	0.9746	0.9540	0.9474	0.9862
6	Variable-1	0.9724	0.9706	0.9686	0.9782	0.9703	0.9712
	Variable-2	0.9790	0.9526	0.9798	0.9640	0.9559	0.9462
	Mean	0.9757	0.9616	0.9742	0.9711	0.9559	0.9807
7	Variable-1	0.9746	0.9762	0.9805	0.9752	0.9621	0.9832
	Variable-2	0.9810	0.9735	0.9668	0.9727	0.9752	0.9772
	Mean	0.9778	0.9749	0.9737	0.9740	0.9687	0.9707
8	Variable-1	0.9647	0.9797	0.9665	0.9774	0.9485	0.9532
	Variable-2	0.9887	0.9887	0.9865	0.9409	0.9723	0.9772
	Mean	0.9767	0.9709	0.9765	0.9765	0.9604	0.9687
9	Variable-1	0.9787	0.9726	0.9748	0.9721	0.9759	0.9251
	Variable-2	0.9733	0.9745	0.9805	0.9569	0.9624	0.9503
	Mean	0.9760	0.9736	0.9777	0.9645	0.9692	0.9765
10	Variable-1	0.9748	0.9810	0.9706	0.9662	0.9769	0.9632
	Variable-2	0.9430	0.9574	0.9651	0.9568	0.9636	0.9042
	Mean	0.9589	0.9692	0.9678	0.9615	0.9636	0.9723
Total	Min	0.9589	0.9616	0.9627	0.9540	0.9474	0.9687
	Max	0.9780	0.9788	0.9831	0.9786	0.9747	**0.9887**
	Avg	0.9726	0.9725	0.9753	0.9653	0.9653	**0.9770**
	SD	0.0066	0.0066	0.0066	0.0081	0.0079	0.0070

Actually, the results of the tuning process have been negatively affected by eliminating the algorithm local search improvement strategy.

For the ten independent runs, Fig. 8 illustrates graphically the best obtained fitness values.

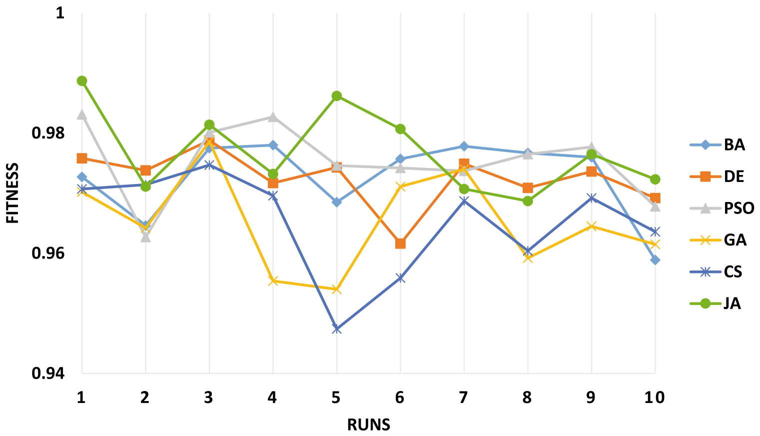


Fig. 8 The best values of various algorithms during the generations

From Fig. 8, it can be concluded that the initial values of the parameters do not generally judge reliability of results obtained by the observed reactive nature-inspired algorithms. However, the CS obtained exceptionally the worst result in the fifth run. Obviously, initial conditions affects performance of this algorithm. Consequently, the algorithm could fall in local optima due to the poor initial parameters. Exclusively and at the first iteration, JA has distinctively achieved better results superior to the other algorithms.

5 Conclusion

This study presented a literature review of the JA. The major aim of this review is to summarize an overview of JA variants and applications. Also, we proved the wide application of JA by reviewing applications in clustering, classification, and other diverse engineering problems. Furthermore, this review investigated the converge performance of JA in solving the Himmelblau function in terms of the accuracy and convergence speed parameters as compared to BA, DE, PSO, GA, and CS algorithms. The choice of JA is based on the fact that, aside being swarm intelligence technique, and not more than a decade ago, it has proved to be extremely successful. From the foregoing discussion, it is obvious that the parameter-less Jaya technique has the advantage of ease of use as juniors can easily solve optimization problems in scientific and engineering fields without having to worry about strict and time-consuming parameter settings. In short, it was observed that JA gives better significant results in minimizing the Himmelblau function than the other hugely popular swarm-based algorithms. In the future, exploitation and improvement of the JA with other optimization algorithms are recommended for further benefits of solving imperative optimization problems.

References

1. Tharwat A, Houssein EH, Ahmed MM, Hassanien AE, Gabel T (2017) Mogoa algorithm for constrained and unconstrained multi-objective optimization problems. App Intell, 1–16
2. Ewees AA, Elaziz MA, Houssein EH (2018) Improved grasshopper optimization algorithm using opposition-based learning. Expert Syst Appl 112:156–172
3. Hassanien AE, Kilany M, Houssein EH, AlQaheri H (2018) Intelligent human emotion recognition based on elephant herding optimization tuned support vector regression. Biomed Sig Process Control 45:182–191
4. Lin M-H, Tsai J-F, Yu C-S (2012) A review of deterministic optimization methods in engineering and management. Math Prob Eng
5. Hussien AG, Hassanien AE, Houssein EH (2017) Swarming behaviour of salps algorithm for predicting chemical compound activities. In: 2017 eighth international conference on intelligent computing and information systems (ICICIS). IEEE, pp 315–320
6. Hussien AG, Hassanien AE, Houssein EH, Bhattacharyya S, Amin M (2019) S-shaped binary whale optimization algorithm for feature selection. In: Recent trends in signal and image processing. Springer, pp 79–87
7. Jimeno-Morenilla A, Sánchez-Romero J, Migallón H, Mora-Mora H (2018) Jaya optimization algorithm with gpu acceleration. J Supercomput, 1–13
8. Hussien AG, Hassanien AE, Houssein EH (2017) A binary whale optimization algorithm with hyperbolic tangent fitness function for feature selection. In: 2017 eighth international conference on intelligent computing and information systems (ICICIS). IEEE, pp 166–172
9. Rao RV, Rai DP, Balic J (2017) A multi-objective algorithm for optimization of modern machining processes. Eng Appl Artif Intell 61:103–125
10. Ahmed MM, Houssein EH, Hassanien AE, Taha A, Hassanien E (2017) Maximizing lifetime of wireless sensor networks based on whale optimization algorithm. In: International conference on advanced intelligent systems and informatics. Springer, pp 724–733
11. Houssein EH, Younan M, Hassanien AE (2019) Nature-inspired algorithms: a comprehensive review. Hybrid Comput Intell Res Appl, p 1
12. Liu W, Liu L, Chung I-Y, Cartes DA (2011) Real-time particle swarm optimization based parameter identification applied to permanent magnet synchronous machine. Appl Soft Comput 11(2):2556–2564
13. Demiroren A, Zeynelgil H (2007) Ga application to optimization of agc in three-area power system after deregulation. Int J Electr Power Energy Syst 29(3):230–240
14. Karaboga D, Gorkemli B, Ozturk C, Karaboga N (2014) A comprehensive survey: artificial bee colony (abc) algorithm and applications. Artif Intell Rev 42(1):21–57
15. Elsisi M, Soliman M, Aboelela M, Mansour W (2017) Optimal design of model predictive control with superconducting magnetic energy storage for load frequency control of nonlinear hydrothermal power system using bat inspired algorithm. J Energy Storage 12:311–318
16. Opara KR, Arabas J (2019) Differential evolution: a survey of theoretical analyses. Swarm Evol Comput 44:546–558
17. Ahmed MM, Houssein EH, Hassanien AE, Taha A, Hassanien E (2019) Maximizing lifetime of large-scale wireless sensor networks using multi-objective whale optimization algorithm. Telecommun Syst, pp 1–17
18. Ismaeel AA, Elshaarawy IA, Houssein EH, Ismail FH, Hassanien AE (2019) Enhanced elephant herding optimization for global optimization. IEEE Access 7:34738–34752
19. Hamad A, Houssein EH, Hassanien AE, Fahmy AA (2018) Hybrid grasshopper optimization algorithm and support vector machines for automatic seizure detection in eeg signals. In: International conference on advanced machine learning technologies and applications. Springer, pp 82–91
20. Houssein EH, Hosney ME, Oliva D, Mohamed WM, Hassaballah M (2020) A novel hybrid harris hawks optimization and support vector machines for drug design and discovery. Comput Chem Eng 133:106656

21. Heidari AA, Mirjalili S, Faris H, Aljarah I, Mafarja M, Chen H (2019) Harris hawks optimization: Algorithm and applications. Fut Gener Comput Syst 97:849–872
22. Houssein EH, Saad MR, Hussain K, Zhu W, Shaban H, Hassaballah M (2020) Optimal sink node placement in large scale wireless sensor networks based on harris' hawk optimization algorithm. IEEE Access 8:19381–19397
23. Hashim FA, Houssein EH, Mabrouk MS, Al-Atabany W, Mirjalili S (2019) Henry gas solubility optimization: a novel physics-based algorithm. Fut Gener Comput Syst 101:646–667
24. Hashim FA, Houssein EH, Hussain K, Mabrouk MS, Al-Atabany W (2019) A modified henry gas solubility optimization for solving motif discovery problem. Neural Comput Appl, pp 1–13
25. Rao R (2016) Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems. Int J Indus Eng Comput 7(1):19–34
26. Warid W, Hizam H, Mariun N, Abdul-Wahab N (2016) Optimal power flow using the jaya algorithm. Energies 9(9):678
27. Pandey HM (2016) Jaya a novel optimization algorithm: What, how and why? In: 6th international conference-cloud system and big data engineering (Confluence). IEEE, 728–730
28. Rao RV, Waghmare G (2017) A new optimization algorithm for solving complex constrained design optimization problems. Eng Optim 49(1):60–83
29. Hamad A, Houssein EH, Hassanien AE, Fahmy AA (2017) A hybrid eeg signals classification approach based on grey wolf optimizer enhanced svms for epileptic detection. In: International conference on advanced intelligent systems and informatics. Springer, pp 108–117
30. Ismail FH, Houssein EH, Hassanien AE (2018) Chaotic bird swarm optimization algorithm. In: International conference on advanced intelligent systems and informatics. Springer, pp 294–303
31. Nanda J, Mishra S, Saikia LC (2009) Maiden application of bacterial foraging-based optimization technique in multiarea automatic generation control. IEEE Trans Power Syst 24(2):602–609
32. Rao RV, Savsani VJ, Vakharia D (2011) Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems. Comput-Aided Des 43(3):303–315
33. Rao RV (2016) Teaching-learning-based optimization algorithm. In: Teaching learning based optimization algorithm. Springer, pp 9–39
34. Rao R (2016) Review of applications of tlbo algorithm and a tutorial for beginners to solve the unconstrained and constrained optimization problems. Dec Sci Lett 5(1):1–30
35. Rao RV, Kalyankar V, Waghmare G (2014) Parameters optimization of selected casting processes using teaching-learning-based optimization algorithm. Appl Math Modell 38(23):5592–5608
36. Rao RV, Rai DP (2017) Optimization of selected casting processes using jaya algorithm. Mater Today Proc 4(10):11056–11067
37. Yao X (1999) Evolutionary computation: theory and applications. World Scientific
38. Fister D, Fister I Jr, Fister I, Šafarič R (2016) Parameter tuning of pid controller with reactive nature-inspired algorithms. Robot Auton Syst 84:64–75
39. Pandey SK, Mohanty SR, Kishor N (2013) A literature survey on load-frequency control for conventional and distribution generation power systems. Renew Sustain Energy Rev 25:318–334
40. Wen G, Hu G, Hu J, Shi X, Chen G (2016) Frequency regulation of source-grid-load systems: a compound control strategy. IEEE Trans Ind Inf 12(1):69–78
41. Shi Y, Eberhart RC (1998) Parameter selection in particle swarm optimization. In: International conference on evolutionary programming. Springer, pp 591–600
42. Aarts E, Korst J (1988) Simulated annealing and boltzmann machines
43. Rao R, Rai D, Ramkumar J, Balic J (2016) A new multi-objective jaya algorithm for optimization of modern machining processes. Adv Prod Eng Manag 11(4):271
44. Gao K, Zhang Y, Sadollah A, Lentzakis A, Su R (2017) Jaya, harmony search and water cycle algorithms for solving large-scale real-life urban traffic light scheduling problem. Swarm Evol Comput 37:58–72

45. Singh SP, Prakash T, Singh V, Babu MG (2017) Analytic hierarchy process based automatic generation control of multi-area interconnected power system using jaya algorithm. Eng Appl Artif Intell 60:35–44
46. Wang L, Huang C (2018) A novel elite opposition-based jaya algorithm for parameter estimation of photovoltaic cell models. Optik 155:351–356
47. Yu K, Liang J, Qu B, Chen X, Wang H (2017) Parameters identification of photovoltaic models using an improved jaya optimization algorithm. Energy Convers Manag 150:742–753
48. Huang C, Wang L, Yeung RS-C, Zhang Z, Chung HS-H, Bensoussan A (2018) A prediction model-guided jaya algorithm for the pv system maximum power point tracking. IEEE Trans Sustain Energy 9(1):45–55
49. Rao RV, Saroj A (2017) A self-adaptive multi-population based jaya algorithm for engineering optimization. Swarm Evol Comput 37:1–26
50. Warid W, Hizam H, Mariun N, Wahab NIA (2018) A novel quasi-oppositional modified jaya algorithm for multi-objective optimal power flow solution. Appl Soft Comput 65:360–373
51. Rao RV, Rai DP (2017) Optimisation of welding processes using quasi-oppositional-based jaya algorithm. J Exp Theoret Artif Intell 29(5):1099–1117
52. Rao R, More K, Taler J, Ocłoń P (2016) Dimensional optimization of a micro-channel heat sink using jaya algorithm. Appl Thermal Eng 103:572–582
53. Rao RV, Saroj A (2017) Constrained economic optimization of shell-and-tube heat exchangers using elitist-jaya algorithm. Energy 128:785–800
54. Rao R, More K (2017) Design optimization and analysis of selected thermal devices using self-adaptive jaya algorithm. Energy Convers Manag 140:24–35
55. Wang S-H, Phillips P, Dong Z-C, Zhang Y-D (2018) Intelligent facial emotion recognition based on stationary wavelet entropy and jaya algorithm. Neurocomputing 272:668–676
56. Du D-C, Vinh H-H, Trung V-D, Hong Quyen N-T, Trung N-T (2018) Efficiency of jaya algorithm for solving the optimization-based structural damage identification problem based on a hybrid objective function. Eng Optim 50(8):1233–1251
57. Abhishek K, Kumar VR, Datta S, Mahapatra SS (2017) Application of jaya algorithm for the optimization of machining performance characteristics during the turning of cfrp (epoxy) composites: comparison with tlbo, ga, and ica. Eng Comput 33(3):457–475
58. Degertekin S, Lamberti L, Ugur I (2018) Sizing, layout and topology design optimization of truss structures using the jaya algorithm. Appl Soft Comput 70:903–928
59. Zhang Y, Yang X, Cattani C, Rao R, Wang S, Phillips P (2016) Tea category identification using a novel fractional fourier entropy and jaya algorithm. Entropy 18(3):77
60. Das SR, Mishra D, Rout M (2017) A hybridized elm-jaya forecasting model for currency exchange prediction. J King Saud Univ-Comput Inf Sci
61. Migallón H, Jimeno-Morenilla A, Sanchez-Romero J-L (2018) Parallel improvements of the jaya optimization algorithm. Appl Sci 8(5):819
62. Migallón H, Jimeno-Morenilla A, Sánchez-Romero J, Rico H, Rao R (2019) Multipopulation-based multi-level parallel enhanced jaya algorithms. J Supercomput, pp 1–20
63. Rao RV (2019) Jaya: an advanced optimization algorithm and its engineering applications. Springer
64. Himmelblau DM (1972) Applied nonlinear programming. McGraw-Hill Companies

World Cup Optimization Algorithm: Application for Optimal Control of Pitch Angle in Hybrid Renewable PV/Wind Energy System

Navid Razmjooy, Vania V. Estrela, Reinaldo Padilha, and Ana Carolina Borges Monteiro

Abstract In the last decades because of the subsequent rise in the prices of fossil fuels, the application of sustainable energies like solar energy and wind energy has been increased. A combined energy generation system containing a wind turbine and photovoltaic system is studied. Such a hybrid system has more efficiency from the single wind turbine or a single PV array. In this study, the Maximum Power Point Tracking (MPPT) system is adopted to get the maximum possible power from the PV array. The control of the pitch angle in the wind turbine is a popular appliance to adjust the aerodynamic torque of the turbine once the wind speed is superior to the considered speed ratio. Formally, pitch control systems usually employ a PI controller which needs a precise mathematical model of the system. Here, a new applicable optimization algorithm is utilized to optimize the pitch angle controller. Test and validation studies with the proposed algorithm are compared with GA and results are presented with a good agreement.

Keywords Power generation · Wind turbine · Photovoltaic · World cup optimization algorithm · Genetic algorithm · Boost converter · Perturb and observe algorithm (p&o) · Maximum power point tracker · Pitch angle · Optimal control

N. Razmjooy (✉)
Department of Electrical and Control Engineering, Tafresh University, 3951879611 Tafresh, Iran
e-mail: navid.razmjooy@hotmail.com

V. V. Estrela
Department of Telecommunications, Fluminense Federal University (UFF), Niterói, Brazil
e-mail: vania.estrela.phd@ieee.org

R. Padilha · A. C. B. Monteiro
School of Electrical and Computer Engineering (FEEC), University of Campinas—UNICAMP, Av. Albert Einstein—400, Barão Geraldo, Campinas, SP, Brazil
e-mail: padilha@decom.fee.unicamp.com

A. C. B. Monteiro
e-mail: monteiro@decom.fee.unicamp.com

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_3

1 Introduction

In the last decades, renewable energy systems have been increasingly turned into a popular solution for supplying energy due to technological progressions and due to lower pollution than fuel-based energy resources [1]. The contribution of renewable energies in power generation is about nineteen percent, with sixteen percent of global final energy consumption; wind and solar energies as new renewable energies accounted for another 3% and are growing so fast [2]. Wind and photovoltaic energies are the most accessible resources among other renewable energies [3].

The total energy production by using photovoltaic (PV) modules is an expensive technical solution which requires a large surface. Furthermore in winter when the solar potential is low, large storage capacity units can supply the consumers required electricity.

Using a hybrid structure for the wind turbine and the PV arrays for supplying the consumption load demand in the distribution system increases the potential of the system in all the seasons. The hybrid PV/wind systems have high potency to supply continuous electricity by lessening the energy storage units and hence, it uses power conversion and control equipment more efficiently than either of the individual sources.

Typically, different methodologies like graphical construction method [4], probabilistic approach [5] and iterative approaches [6] are proposed by the researchers to have an optimal configuration of the hybrid systems in terms of economic and technical analysis.

A fundamental scheme for system dynamics control is to partially slow down the speed change due to involving the large inertia which hardens the speed control of power converter is highly variable applications. Pitch control is rather fast; so it can be used in a better case of tuning the power flow; especially when they are close to the high-speed limit.

The pitch angle control is a popular appliance to tune the aerodynamic torque of the wind turbine when the speed is superior rated speed. Typically, most of the conventional pitch control strategies are based on PI controllers [7–10]. Developing a hard mathematical model is impossible to get high nonlinearity characterizing the PV-Wind hybrid system. Deterministic methods like generalized reduced gradient methods and sequential quadratic programming approaches are not flexible to assimilate the solution for the problem.

For solving and handling a normal modeling problem, a classical algorithm can be adopted. This is often done by considering several presumptions which are not easy to adjust in different conditions.

At the same time, we can achieve a simple and operative controller using evolutionary algorithms. Evolutionary algorithms are turning into useful tools. They can be employed for solving complex and practical problems in different areas and are improving exponentially.

In this study, the PI controller is utilized to adjust the speed of the rotor and the electrical power through the generator torque and the collective pitch angle. The world cup optimization algorithm is used as an evolutionary algorithm for optimal control of the variable speed, pitch-regulated PV-Wind system. World cup optimization algorithm is a new optimization algorithm that is inspired by the FIFA world cup competitions. Despite its recent development, it has illustrated successful results in a number of practical applications like Control of DC motors [11], optimizing the power system stabilizer [12], solving ordinary differential equations [13], optimization of the neural networks [14] and even image processing applications [15], etc.

2 System Configuration

Figure 1 shows the general configuration of the presented stand-alone system. The system includes a PMSG wind generator and a PV array as primary energy resources and a battery bank as the backup energy resource. An AC/DC diode rectifier is utilized in the output voltage of the wind turbine to connect the wind turbine and the PV array through a DC-link voltage. MPPT is also utilized to extract the maximum available power from the wind turbine. The type of battery bank is Lead-Acid. It is employed to store the energy (in this work battery section is not pending), for regulating the DC-Bus voltage through a reversible current DC/DC converter and to supply the load in the hybrid status when low solar radiation conditions and (or) wind speed [16]. The generated energy for both wind source and (or) PV array is transferred from the DC-Bus to the house load through a DC/AC inverter and filtered by an LC filter.

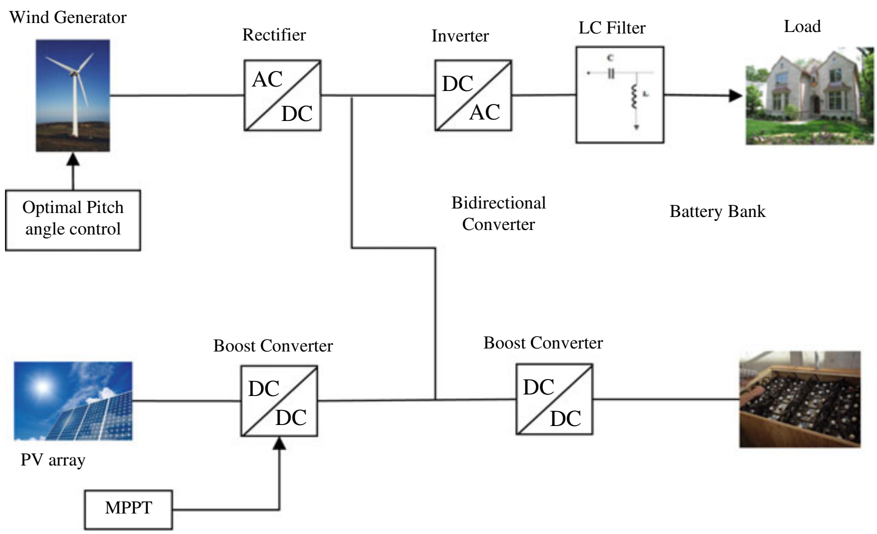


Fig. 1 Structure of a hybrid Turbine-PV system

3 Modeling Approach

In this section, the main configuration of the system and its components have been explained.

3.1 Photovoltaic System

Photovoltaic units (PVs) produce electricity from the sun and are surprisingly simple, effective, and durable. These components, with no moving components, can start up the gadgets, recharge batteries, or generate energy for the power grid [17, 18].

An array of solar cells collects energy through the photovoltaic effect (the same solar generator). It was discovered in 1839 by the French physicist Alexander D'Becquerel. In this way, PV cells produce electricity from solar photons. Figure 2 shows the main arrangement of a photovoltaic system.

When the sun hits a PV cell, that cell absorbs a number of photons, and the energy of these photons in the semiconducting material becomes an electron. By acquiring energy from a photon, an electron can leave its usual position in the semiconductor atom and become part of the current in an electric circuit [19–21].

A PV cell is a nonlinear device which can be illustrated as a current source in parallel with a diode [22]. As can be observed from Fig. 3, the photovoltaic cell is typically modeled by a one-diode electrical equivalent.

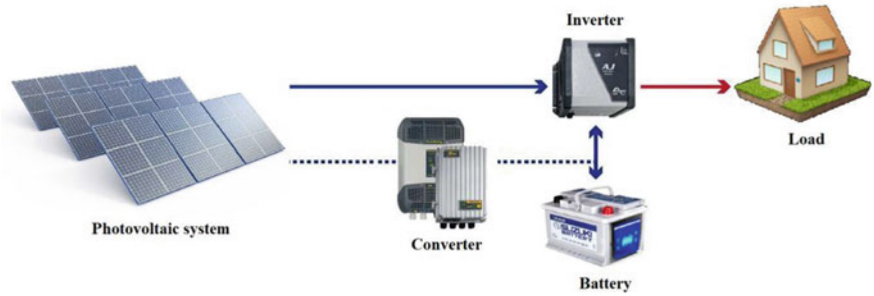
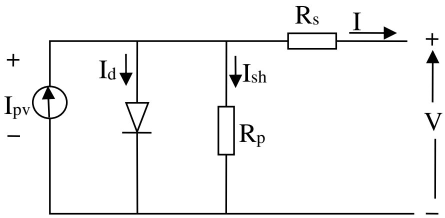


Fig. 2 Structure of a photovoltaic system

Fig. 3 Model of a single PV cell



The output current source is the difference between the normal diode current (I_d) and the photocurrent (I_{PV}) and is directly proportional to the light falling on the cell [23].

Real PV cell includes the connection of parallel and series interior resistance, namely R_p and R_s that is formulated as follows:

$$I = I_{pv} - I_d = I_{pv} - I_0\left[\exp\left(\frac{V + R_s \times I}{m \times K \times T_c}\right) - 1\right] \quad (1)$$

In the above equation *Tc, m*, and *K* are temperature, the Boltzmann's constant (1.3806503 e−23 J K^{-1}), and the diode quality constant, respectively, *I* is the output PV current, V_t is a thermal voltage of array, *V* is the output PV voltage, and I_{pv} is the current that is generated by the incident light (which is directly proportional to the Sun irradiation) [24]. I_{pv} can be described as follows:

$$I_{pv} = (I_{pvn} + K_i \Delta t)\frac{G}{G_n} \quad (2)$$

where, I_{pvn} is the generated current from the light at the nominal condition, G_n is the integral of the spectral irradiance extended to all wavelengths of interest and is equal to 1000 Wm^{-2} [25], and I_0 is the diode saturation current as follows:

$$I_0 = \frac{I_{sc+K_i \times \Delta t}}{\exp(\frac{V_{oc}+K_v \times \Delta t}{a \times V_t})} \quad (3)$$

where, *a* is the diode *ideality* constant.

In this research, the BP340 PV data-sheet is utilized. Table 1 illustrates the parameters of this datasheet.

Table 1 The parameters of BP340 PV module

Parameter	Variable	Value
Voltage at P_{max}	V_{mpp}	17.3 V
Short-circuit current	I_{SC}	2.54 A
Open-circuit voltage	V_{OC}	21.8 V
Temperature coefficient of open-circuit voltage	K_v	−(80 ± 10) mV/°C
Maximum power	P_{mpp}	40 W
Current at P_{max}	I_{mpp}	2.31 A
The temperature coefficient of short-circuit current	K_i	(0.065 ± 0.015)%/°C

3.1.1 Maximum Power Point Tracking (MPPT) Algorithm

The power output of a PV cell changes with revolves in the direction of the sun, with solar insulation level and with temperature variation as shown in the Fig. 4 [26]. As can be seen in the PV (power vs. voltage) curve of the module there is a single maximum of the power; i.e. there is just a peak power corresponding to a particular voltage and current. The efficiency of the solar PV module is low and about 13%; therefore, a maximum power point tracker (MPPT) is used to extract the maximum power from the solar PV module and transferring that power to the load. The maximum power point tracker (MPPT) algorithm is now customary in grid-tied photovoltaic power systems and is growing as a more popular in hybrid systems. Figure 4 shows the nominal P-V and I-V curves of a nominal PV module.

MPPT includes a high-frequency switch-mode DC/DC converter and a controller interconnecting a PV power source and a load that maximizes the power output from a PV module or array with different operating conditions and therefore maximizes the system energy efficiency [27].

Typically, if an MPPT installs in a hybrid system, voltage mismatches get low and losses are kept to a minimum due to conversion. Albeit, the capacity of MPPT is installer dependent and can develop operation even though costs are increased through the MPPT purchase. From Eqs. (4) and (5) and regarding that step-down converter is used; the resistance corresponding to the peak powerpoint can be achieved by changing the duty cycle.

$$V_o = D \times V_i \tag{4}$$

$$R_o = D^2 \times R_i \tag{5}$$

where, V_o describes the output voltage, D is the duty cycle, V_i determines the input voltage, and R_i and R_o are the input and the output impedances, respectively.

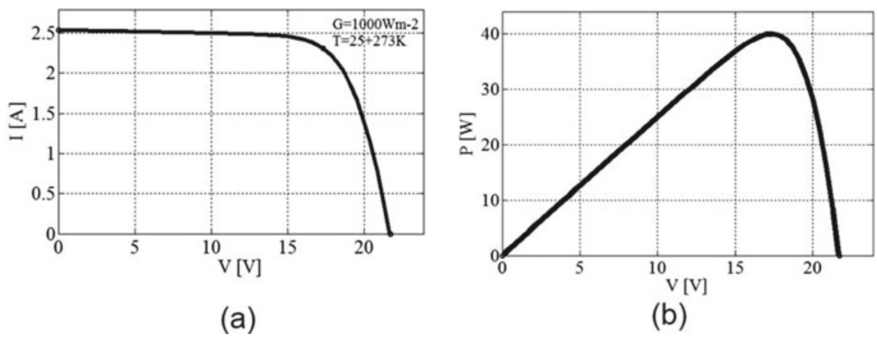


Fig. 4 The nominal PV module characteristic curves **a** I-V curve, and **b** P–V curve

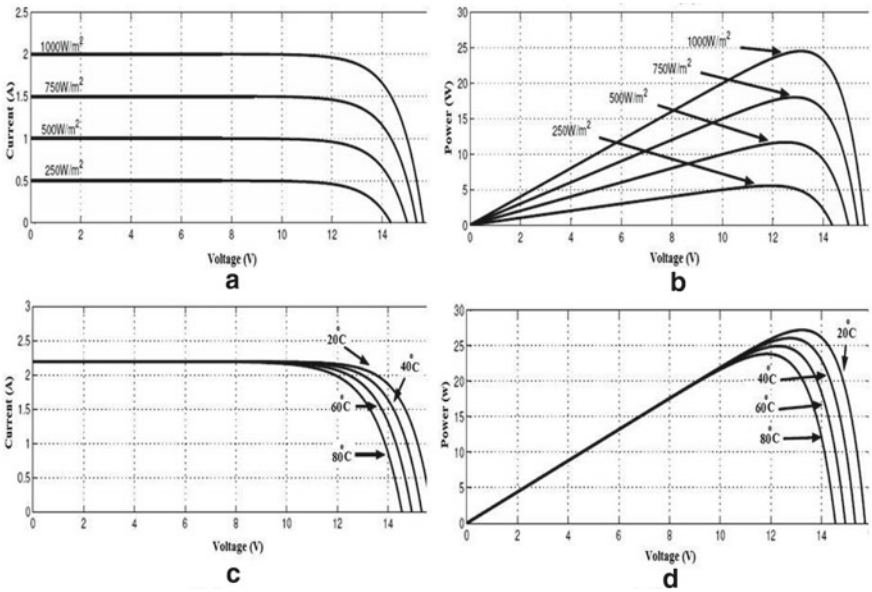


Fig. 5 Change in the module characteristics due to change in insulation (**a**, **b**), and level the change in temperature (**c**, **d**)

There are several algorithms that have been proposed to adjust the duty cycle to achieve peak power. In this research, the sources Perturb and observe (P and O) method based on the MPPT search algorithm is employed. From Fig. 5, the *P* and *O* algorithm modifies the operating voltage or current of the photovoltaic panel until we achieve maximum power from it.

Once the steady-state is achieved, the algorithm oscillates around the peak point. The algorithm is improved such that it adjusts a reference voltage of the module based on its peak voltage. Afterward, a PI controller has been adopted to move the operating point of the module into that special level of voltage. P&O is simple but quick to implement and is still used as a popular algorithm (Fig. 6).

3.1.2 DC/DC Boost Converter

DC/DC boost converters are power converters with an output DC voltage greater than its input DC voltage which is extremely utilized in regulated switch-mode DC power supplies. The input of boost converter is a deregulated voltage achieved from PV array and changes constantly in order to radiation and temperature fluctuant and the average DC output voltage should be adjusted to be balanced by the desired value, during the input voltage variation [28]. The circuit diagram of a DC/DC boost converter with output regulated voltage feedback is shown in Fig. 7 [26].

Figure 8 shows the duty cycle of the boost converter.

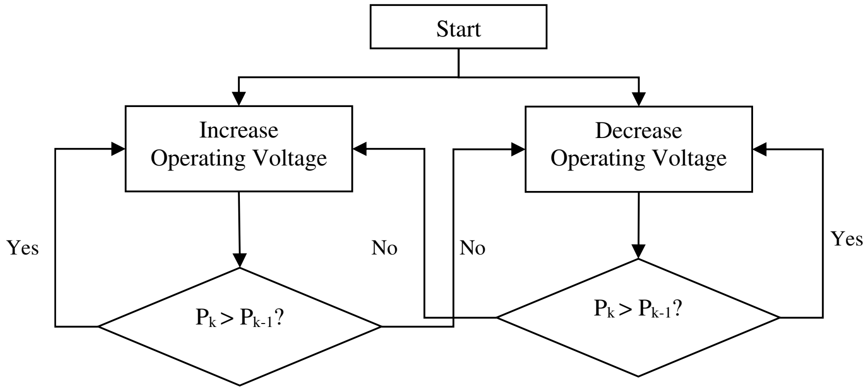


Fig. 6 Flowchart of P&O MPPT algorithm: P_k is the current power and P_{k-1} is the previous power

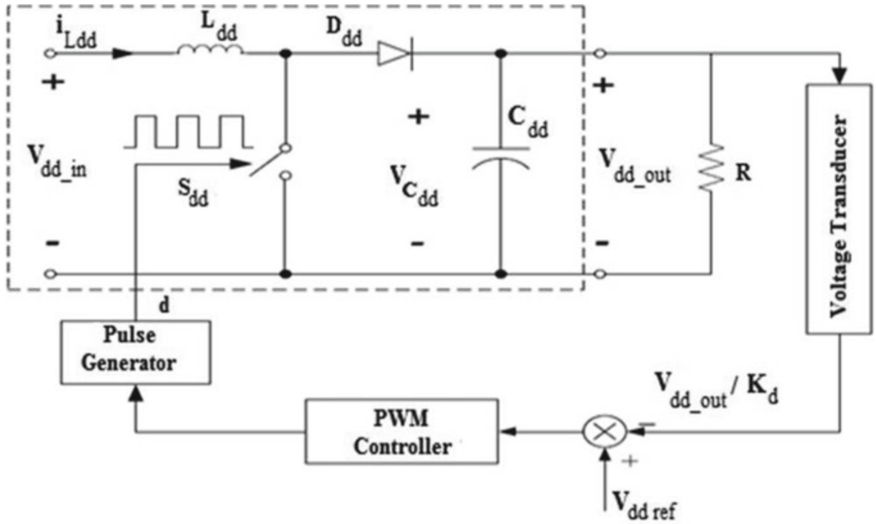


Fig. 7 DC/DC boost converter

The average value of the output voltage in the steady-state is achieved as follows:

$$V_{dd\_out} = \frac{V_{dd\_in}}{1 - D} \quad (6)$$

where, D describes the duty ratio of the switching pulse. For $0 < D < 1$, the output voltage is always higher than the input voltage and this is the reason the circuit is called a DC/DC boost converter (see Fig. 6). The converter controller is adopted for adjusting the DC bus voltage in a desirable range. The output voltage is compared with the reference value; the error signal is processed through a simple PI-based

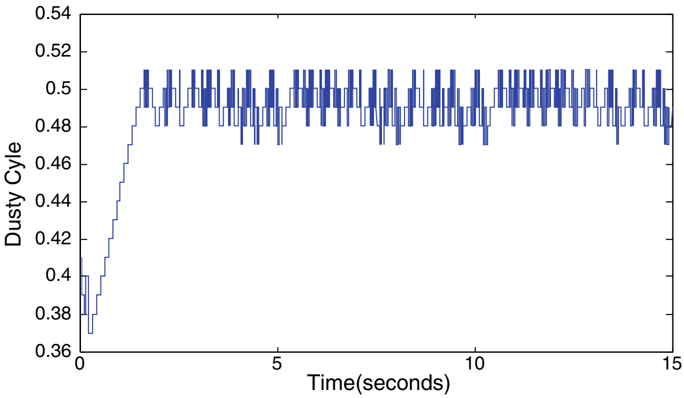


Fig. 8. Boost converter duty cycle

PWM. The control signal is employed to produce a PWM pulse with the right duty ratio following the reference value by the output voltage.

3.1.3 DC/AC Inverter

Inverters are power converters that convert the direct current into the alternating current. Inverters are utilized in different PV system configurations like stand-alone systems, rechargeable batteries, grid-connected systems, and pumping systems with no storage batteries.

Typically, the inverter performs an interface between the PV array and the utility grid. For stand-alone systems, the inverter directs as the grid administrator and feeds the loads.

In this research, a 3-phase, 12-switch inverter PWM voltage source inverter is employed to convert the power from DC to AC.

The performance of the inverter (η) is dependent on power. The duty of an inverter is to keep on the AC side of the voltage constant at the 230 V and to efficiently convert of the input power P_{in} into the output power P_{out}. The performance of the inverter is formulated as follows:

$$\eta = \frac{P_{out}}{P_{in}} = \frac{V_{ac} \times I_{ac} \times \cos\phi}{V_{ac} \times I_{ac}} \Rightarrow I_{dc} = \frac{V_{ac} \times I_{ac} \times \cos\phi}{\eta \times V_{dc}} \tag{7}$$

where, I_{dc} describes the current required by the inverter from the DC side (e.g. from the controller) regard to be able to keep the rated voltage on the AC side (for example on the load), V_{dc} is the input voltage for the inverter transferred by the DC side. The efficiency of converting in most inverters is about 90 percent [28]. The main circuit of a 3-phase voltage source inverter (VSI) is shown in Fig. 9.

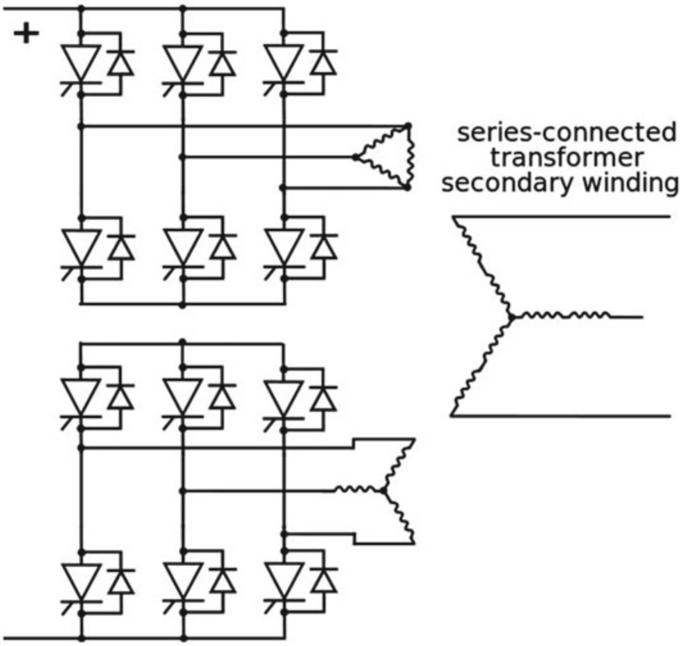


Fig. 9 12-pulse line-commutated inverter circuit

3.2 Wind Turbine

A wind turbine is a mechanical system which is designed to convert the kinetic energy of the input wind into the rotary mechanical energy; in more advanced models, a generator is adopted to convert the rotational energy into the electricity which is called a wind generator. Wind energy is one of the most popular clean energy resources [29–32].

Wind power can be produced by a wind turbine. As the wind turbine rotates, the energy is transmitted by the turbine to the generator, causing it to rotate and then generate AC power.

The electric power produced by the pneumatic turbine is known as AC dirt. Because, given the fluctuation in the wind, the electricity generated in the wind turbine is intentionally in terms of frequency and voltage. To solve this problem, the electric power of the wind turbine is converted to DC. This is done by a converter.

In the last decade, several types of wind turbines have been designed. Most of them include a rotor that turns round propelled by lift or drag forces which result from its interaction with the wind. By considering the position of the rotor axis, wind turbines are categorized into two types of horizontal-axis and vertical-axis.

Modern wind generators preserve more than 90% of available energy from the wind compared with fuel efficiency between 30- 40% for a formal coal-fired

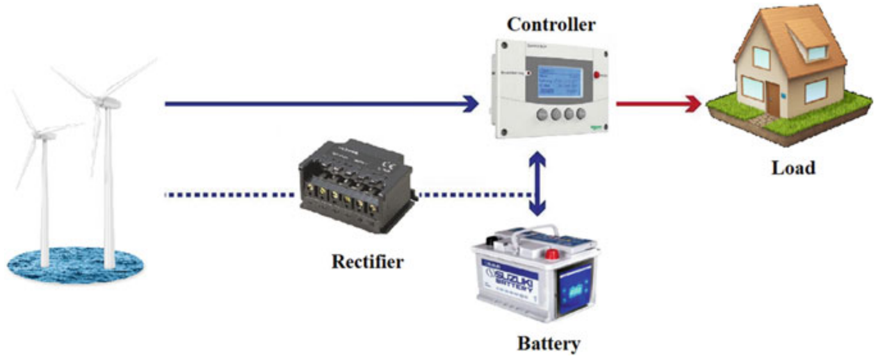


Fig. 10 Structure of a photovoltaic system

station, which basically loses a significant portion of energy through heat loss and pollution. Figure 10 shows the structure of a photovoltaic system.

Several methods have been proposed to control aerodynamic forces on the turbine rotor and thus to manipulate the power in sever wind speeds for protecting the turbine from damages. There are three kinds of techniques for power control in the industry: pitch control, stall control, and active stall regulation.

In this research, a 13.5 m/s constant wind speed with a 10 m/s base speed is assumed and pitch control is selected to control the system. For controlling the pitch angle, the blades can be turned into or out the wind as the power output becomes too high or too low, respectively.

Typically, the superiority of pitch control is its assisted startup, proper power control, emergency stop, and increased energy capture. The mechanical power in the moving air can be given by Chen and Cheng [33]:

$$P_w = \frac{1}{2} \times C_p \times \rho \times A \times V_w^3 \tag{8}$$

$$A = \frac{\pi d^2}{4} \tag{9}$$

$$C_p = \frac{1}{2} \times \left[\frac{116}{\lambda_i} - 0.4 \times \theta_p - 5\right] \times \exp\left(\frac{-21}{\lambda_i}\right) + 0.0068\lambda \tag{10}$$

$$\lambda = \frac{\omega w R}{v\omega} \tag{11}$$

$$\frac{1}{\lambda_i} = \frac{1}{\lambda + 0.08\theta_p} - \frac{0.035}{\theta_p^3 + 1} \tag{12}$$

where, ωw is angular velocity of rotor, $v\omega$ is the wind speed upstream of the rotor, R is the rotor radius, C_p is the power-efficient, P_w is mechanical power in moving air, ρ is the air density and is equal to 2.113 at sea level (T = 25°), A describes the area swept by the rotor blades exposed to the wind, v_w is the wind speed, and θ_p is pitch angle [°], that is stand between blade cross-section chord and plane of rotation.

4 World Cup Optimization Algorithm

World Cup Optimization (WCO) algorithm is introduced by Razmjooy et al. [8]. A new optimization algorithm that is a mimic from the FIFA world cup competition to reach the Gold Cup.

Like other optimization algorithms, WCO starts with a random population. The population here is called the team. Each team is generally taking place in the seeds according to the Ranking table [34]. Rank is one of the most significant cases in the FIFA competition which is arranged based on the teams wins and fails [35].

Since rank defines what team should be placed in what seed, Tit has a direct effect on the position of the teams.

After initializing the random teams and finding the cost of each team (cost function quantity), n first stronger teams are placed in the first seed, and this seeding continues till the weakest team.

At the first level, strong teams have been upgraded to the next level with no competition.

After seeding and initial classification, the main competition starts. In this step, teams compete with each other to achieve more points based on their wins.

After step 1 competitions, from each seed, two teams by the highest point are upgraded to the next level and the rest are eliminated.

This algorithm has also another special parameter which is called Play-Off. This parameter gives a second chance to the third place of the seeds to have one more opportunity for upgrading to the next level. In this case, the team with the highest point among the third-place teams has been upgraded.

After competitions, the final challenge is done between two highest point teams and the winner achieves the gold medal. In the following, the WCO algorithm is summarized:

Step 1 Initialize the teams.

The teams belong to the *countries* and the countries are randomly placed in the M number of continents:

$$Continent = [country_1, country_2, \ldots, country_{N_{\text{var}}}] \tag{13}$$

$$country_i = [x_1, x_2, \ldots, x_{N_{\text{var}}}] \tag{14}$$

where, N_{var} is the optimization problem dimension and x_i is the ith team of the country.

Rank for each continent can be evaluated by considering the achieved points f_r in a continent $(x_1, x_2, \ldots, x_{N_{\text{var}}})$. i.e.

$$Rank = f_r(continent) = f_r(x_1, x_2, \ldots, x_{O_{\text{var}}}) \tag{15}$$

$$O = N \times M \tag{16}$$

where, N and M describe the dimension of the variable and the number of continents, respectively.

The convergence time can be improved by considering an interval of random values and dividing it into the continents. This feature cause the WCO algorithm to decrease the convergence time than the other algorithms.

Step 2 Cost function Evaluation.

After the teams compete with each other for the first time, they should be ranked based on their wins and fails. This ranking in the FIFA rule is not so fair; because there is maybe a continent that includes the strongest teams which are competed with each other.

WCO uses a solution to cover this drawback; to do this solution, the mean value and the standard deviation of the continents should be achieved:

$$\overline{X} = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{17}$$

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n} (X_i - \overline{X})^2} \tag{18}$$

where, n is the members quantity in X and $\overline{X}$ and σ describe the mean value and the standard deviation of the continent X.

Step 3 Ranking:

$$\begin{aligned} X_1 &= [X_{11}, \ldots X_{1n}]^T, \\ X_2 &= [X_{21}, \ldots X_{2n}]^T, \\ &\ldots, \\ X_5 &= [X_{51}, \ldots X_{5n}]^T \end{aligned} \tag{19}$$

$$X_{Total} = [X_{11}, \ldots X_{1n}, X_{21}, \ldots X_{2n}, \ldots, X_{51}, \ldots X_{5n}]^T \tag{20}$$

where, T describes the transpose operator and n is the number of teams.

In this step, two best teams of each continent have been chosen and placed into the other vector (X_{Rank}) to utilize in the next competitions and the optimal values of X_{Total} have been considered as the first cup's winner as follows:

$$X_{Rank} = [X_{11}, X_{12}, X_{21}, X_{22}, \ldots, X_{51}, X_{52}]^T \tag{21}$$

$$\begin{aligned} &X_{champion} = \min(X_{Total}) = \\ &\min([X_{11}, \ldots X_{1n}, X_{21}, \ldots X_{2n}, ..., X_{51}, \ldots X_{5n}]^T) \end{aligned} \tag{22}$$

where, $X_{champion}$ is the minimum value of the solutions.

Step 4 Next stage competition.

After the initial competition, based on the previous ranking from the competitions before, new teams and continents have been regenerated.

Here, the WCO perform differently from the real FIFA by considering a vector with two parts:

$$Pop = X_{total} = [X_{Best}, X_{Rand}] \tag{23}$$

where,X_{Rand} describes a random quantity in a definite interval, $Pop(X_{total})$ is the regenerated teams of the size $(N \times M)$ and X_{Best} illustrates a vector as follows:

$$L < X_{Best} < U \tag{24}$$

$$U = \frac{1}{2} \times ac \times (Ub + Lb) \tag{25}$$

$$L = \frac{1}{2} \times ac \times (Ub - Lb) \tag{26}$$

where, *ac* is a coefficient in the interval $[L_b, U_b]$.

Step 5 Improving the exploration and exploitation in the algorithm.

X_{Rand} and X_{Best} describe the exploration and the exploitation, respectively,X_{Best} is the best position of the previous search space and X_{Rand} describes the new random genesis numbers in the search space.

Step 6) If the termination criteria are satisfied, end the algorithm; otherwise iterate the algorithm. Figure 11 shows the flowchat diagram of the World Cup Optimization Algorithm.

5 PID Controllers

In over 90% of dynamic controllers, different types of PIDs have been used. PID controllers for the first time were used in the shipping industry in the early twentieth century [9]. Today, PID controllers are employed in different applications.

The PID controller is a classical controller that is divided into three basic parts: proportional, integral, and derivative. This controller is one of the most widely used controllers in the industry which indicates the importance of this type of controller.

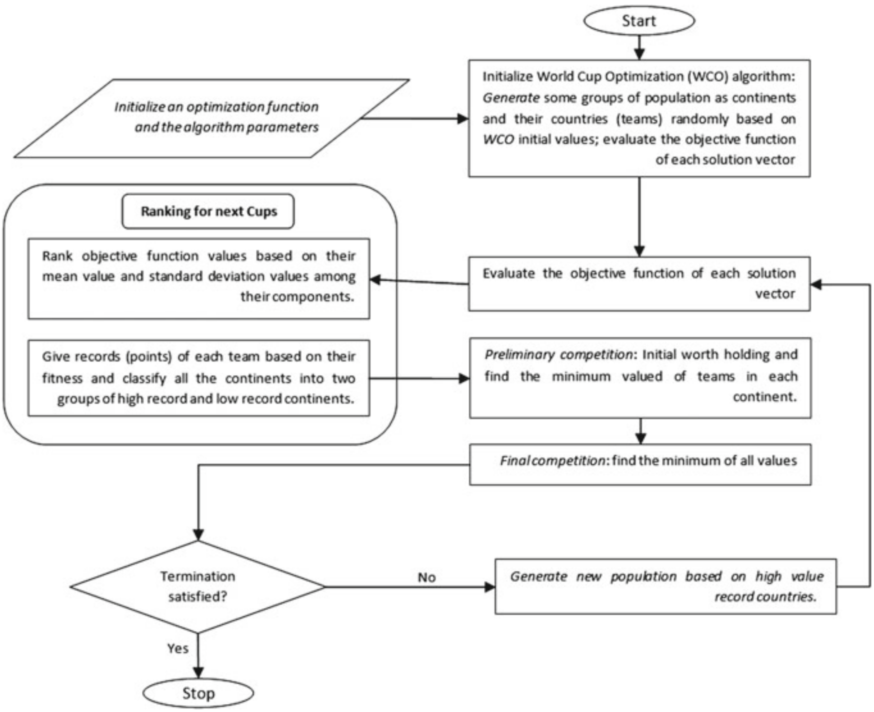


Fig. 11 Flowchart of world cup optimization algorithm

It is also popular closed-loop control mechanisms which are adopted in several control applications like pressure control, DC motor speed control, and temperature control.

The purpose of using the PID algorithm in closed-loop control is to accurately control the system output under different conditions without knowing the exact behavior of the system in response to the input [10, 36].

The input signal in PID is a considered error and the output is a designed signal which performs on the feedback system to minimize that error.

Regardless of the simple concept of these controllers, the design of such controllers in practice is beyond the control of its three main parameters. Different factors affect the performance of this controller, including controller structure, process degree, time constant ratio, process dynamics, actuator element dynamics, derivative filter type, parameter settling time, system overshoot, nonlinear behavior in System, etc. Each of these factors can play a role in designing and adjusting the PID controller.

Each of the derivatives, integral and proportional parameters have specific properties in the control algorithms and each of them is utilized for certain targets. In some cases, some of these parameters can be neglected due to inefficient performance.

Using the above three parameters, we can design three types of controllers including P, PI, PD, or PID. In the following, a sample PID structure is shown.

As seen from the figure above, at first, the feedback has been taken from the output of the system. This value is entered into the comparator and compared with the SETPOINT (input). The error results from this comparison are entered as $e\ (t)$ into the PID control function, and by applying the P, I, and D coefficients in the controller output, there is a control signal to improve the system [37].

This signal changes the system behavior and closes the output into the desired value to equal it. In the following, the characteristics of the PID parameters are presented.

5.1 Proportional Controller (P)

In the Proportional controller, there is a direct relationship between the output and the input, with a certain coefficient that is called *Gain* (Error Signal = Gain × Output).

Of course, the proportional controller is not enough on its own. Because when the system output goes to the optimal value, the error decreases and the control output signal decreases as well. Therefore there is always a lasting error between the optimal amount and the actual output.

This error can be reduced by increasing the controller gain, but it causes system instability and output oscillation. To solve these problems, a proportional controller is usually used with derivative and integral controllers.

5.2 Integral Controller (I)

As its name implies, in this configuration, there is an integral relation between inputs and outputs. This controller is used to compensate for a persistent error, since, as long as there is an error in the output, the integral statement changes, and as a result, the output error decreases gradually.

5.3 Partial-Integral Controller (PI)

The PI controller is a combination of integral and proportional configurations that are connected in parallel. This controller, if properly designed, will have the benefits of both integrative and proportional control. Sustainability, speed, and no permanent state error are features of this controller.

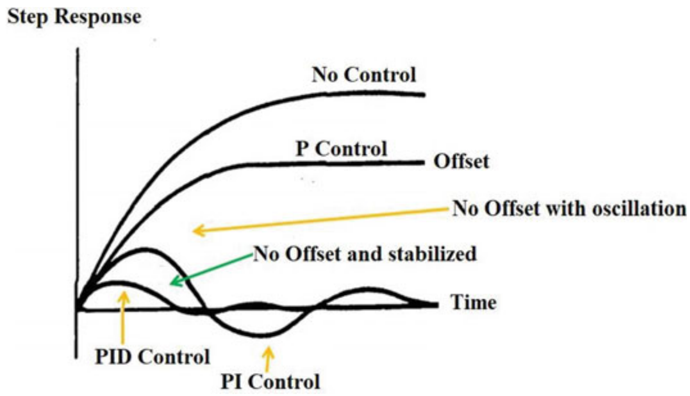


Fig. 12 The step response of the PID controller

5.4 *Partial-Distributed Controller (PD)*

The PD controller is a parallel combination of two types of derivative and integral configurations. The derivative controller has this attribute that coordinates itself with the input changes at once. Therefore, in cases where a fast response is required, such controllers can be used, but the refinement that the derivative operation enhances the noise in the process environment, plus the derivatives are only sensitive to the input changes, so the derivatives are not used alone, but when the derivative property is required in a process, the controller is constructed as a derivative-proportional or derivative-integral or derivative-proportional-integral.

5.5 *PID Controller*

This kind of controller is a parallel combination of proportional, integral and derivative controllers, and is the most common type of controller in the industry. Figure 12 shows the step response of a PID controller.

6 Control Strategy

As presented in Sect. 3.2, pitch angle adjustment for the blades provides a useful appliance. This system regulates or limits turbine efficiency in severe winds.

After wind speed confining, the aerodynamic torque sensitivity is very small to pitch angle. Thus an extortionary controller gain is needed here at higher wind speeds, where a small variation in pitch angle can have a large effect on the torque.

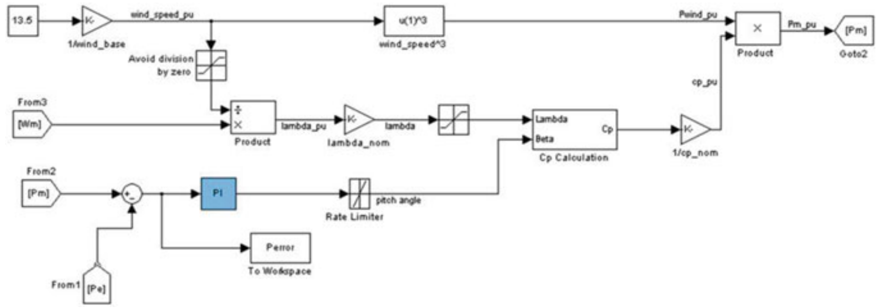


Fig. 13 Wind turbine and PI controller of pitch angle with Simulink/Matlab

The torque sensitivity varies almost linearly with pitch angle frequently, and hence, it can be compensated by changing the overall gain of the controller linearly in inverse proportion to the pitch angle. This improvement gain with an operating point is termed as gain scheduling.

In pitch angle control (see Fig. 13), there is a PI controller which has a proportional gain and an integral time constant. The objective of the WCO is to find the optimal parameters of the PI controller called proportional gain (K_p) and integral gain constant (K_i), for optimizing some performance measure function (cost function). Figure 13 shows the simulink model of the wind turbine and PI controller of pitch angle.

Typically, the PI controller is selected in order to a set of parameters like overshoot (*Mp*), settling time (*ts*), rising time (*tr*) etc. In this article, the objective is to reduce time setting and overshoot regards to system characteristics (turbine mechanical power, electrical power, and pitch angle); hence, a new cost function is described which is considered below:

$$ITASE = \int_{0}^{t_{sim}=10} t \times \left|\Delta p_m^2\right| \tag{27}$$

$$\Delta P_m = P_m - P_e \tag{28}$$

where, t is simulation time, P_m and P_e are turbines mechanical and electrical power respectively.

7 Simulation Results

A 3.5 MW three-phase turbine is adopted in the presented system; the nominal speed of the wind turbine and wind speed are 11 m/s and 13.5 m/s respectively. The

Table 2 Parameters of PV array, wind turbine, boost converter and GA parameter used in the proposed system

PV array		Wind turbine and boost converter		GA parameters		WCO parameters	
Rs	0.46	Wind base	11	Population size	35	Play off	2%
Rp	239.41	Lambda Nom	8.11	Crossover probability	0.9	Alpha (α)	0.5
Iscn	2.54	Cp Nom	0.48	Mutation probability	0.02	Maximum iteration	100
Ipvn	2.5448	Switching frequency	20KHz	Maximum iteration	100	Var_{min}	1
Vocn	21.8	Inductor	20mH	Var_{min}	1	Var_{max}	50
T	25 °C	Capacitor	6mF	Var_{max}	50		

generation power of PV and turbine are 800 KW and 1.5 MW. Transition power is about 1.5 MW and the other parameters are illustrated in Table 2 [38]:

The PI controller of the system is optimized by using WCO and the results are compared by GA.

The optimal parameters for both algorithms are illustrated in Tables 3 and 4 illustrates two different operating points that are employed for studying the methods' performance.

Final achieved values for the controller, overshoot, and settling time for pitch angle, mechanical power, and electrical power are illustrated below:

From Tables 5 and 6, it is clear that WCO has less overshoot in both operating points that the GA. It is also seen that the settling time in WCO is faster than the GA.

The results of pitch angle, electrical power, and mechanical power are shown in Figs. 14, 15 and 16.

Table 3 Optimal parameters of the pitch angle controller

	K_p	K_I
WCO	6.67	37.73
GA	8.42	33.25

Table 4 The operating points for the analysis system

Case No	Cp	W(m/s)
1	0.45	9
2	0.40	12

Table 5 Mechanical power analysis for operating points

Case No	WCO		GA	
	Over shoot	Setting time (s)	Over shoot	Setting time (s)
1	0.3092	1.8436	0.5917	2.948
2	0.039	1.1482	0.3447	2.2604

Table 6 Electrical power analysis for operating points in the Results wind turbine

Case No	WCO		GA	
	Over shoot	Setting time (s)	Over shoot	Setting time (s)
1	0.6953	2.9746	0.9435	3.6482
2	0.7127	1.9276	0.9907	3.4218

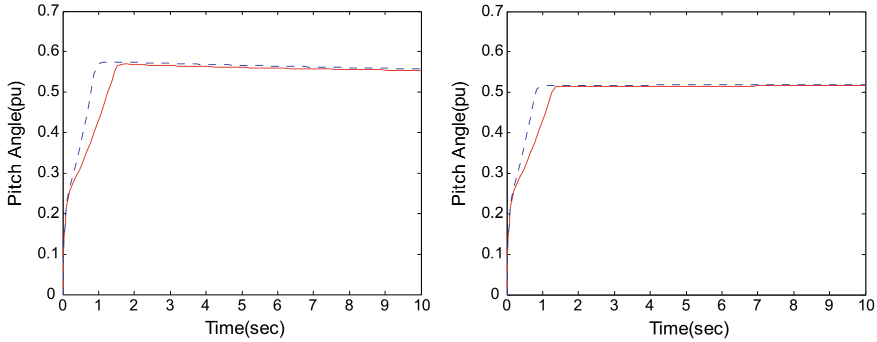


Fig. 14 Pitch angle for 2 different operating points; solid (WCO), dashed (GA)

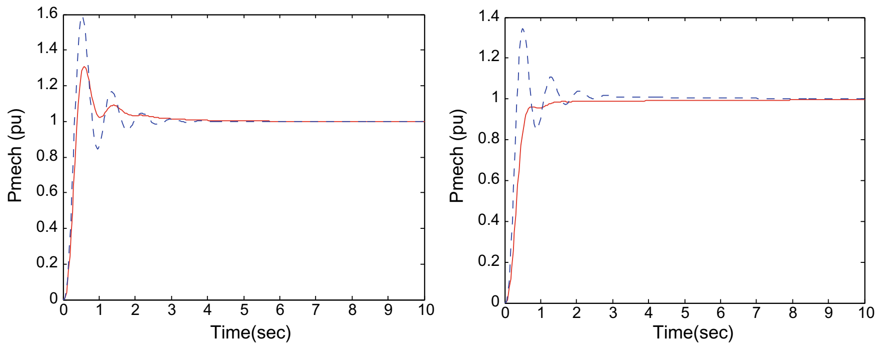


Fig. 15 Mechanical power for 2 different operating points; solid (WCO), dashed (GA)

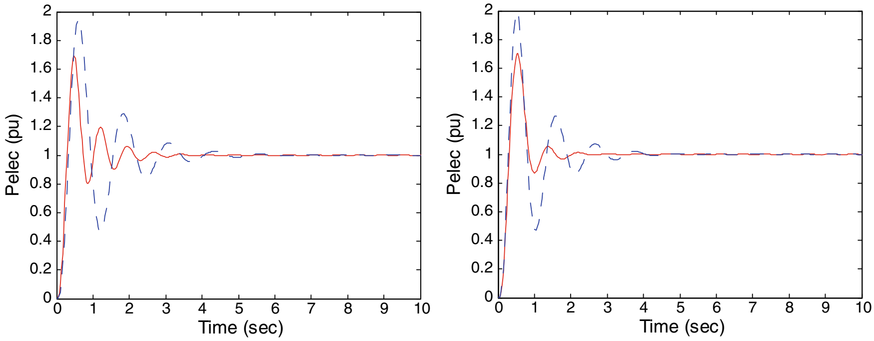


Fig. 16 Electrical power for 2 different operating points; solid (WCO), dashed (GA)

8 Conclusion

A hybrid system containing renewable photovoltaic and wind energy is modeled. The control system is implemented to the wind turbine and consists of the control of pitch angle and mechanical and electrical power systems. Linear PI controller is employed to control both the turbine and its parameters initially designed at a specific operating point. The world cup optimization (WCO) algorithm is used to find the optimal parameters of the PI controller for the turbine for minimizing the maximum overshoot and the settling time.

ITASE is selected as the fitness function as a new time-domain function to measure the performance of the controller. Simulation studies are implemented in two different operating points where the optimal controller is designed; optimized values are compared by GA and showed the prominence of WCO. Results show that the presented algorithm is efficient to achieve the optimal parameters of the PI controllers and develops the transient efficiency of the wind turbine generator system over a wide range of operating conditions.

References

1. Hosseini H, Farsadi M, Lak A, Ghahramani H, Razmjooy N (2012) a novel method using imperialist competitive algorithm (ICA) for controlling pitch angle in hybrid wind and PV array energy production system. Int J Tech Phys Probl Eng (IJTPE), 145–152
2. Mollahosseini A, Hosseini SA, Jabbari M, Figoli A, Rahimpour A (2017) Renewable energy management and market in Iran: a holistic review on current state and future demands. Renew Sustain Energy Rev 80:774–788
3. Khojasteh D, Khojasteh D, Kamali R, Beyene A, Iglesias G (2017) Assessment of renewable energy resources in Iran; with a focus on wave and tidal energy. Renew Sustain Energy Rev
4. Borowy BS, Salameh ZM (1996) Methodology for optimally sizing the combination of a battery bank and PV array in a wind/PV hybrid system. IEEE Trans Energy Convers 11:367–375
5. Tina G, Gagliano S, Raiti S (2006) Hybrid solar/wind power system probabilistic modelling for long-term performance assessment. Sol Energy 80:578–588

6. Yang H, Lu L, Zhou W (2007) A novel optimization sizing model for hybrid solar-wind power generation system. Sol Energy 81:76–84
7. Xia A, Hu G, Li Z, Huang D, Wang F (2018) Self-optimizing pitch control for large scale wind turbine based on ADRC. In: IOP conference series: materials science and engineering. IOP Publishing, p 012155
8. Razmjooy N, Khalilpour M, Ramezani M (2016) A new meta-heuristic optimization algorithm inspired by FIFA world cup competitions: theory and its application in PID designing for AVR system. J Control Autom Elect Syst 27:419–440
9. Khalilpour M, Valipour K, Shayeghi H, Razmjooy N (2013) Designing a robust and adaptive PID controller for gas turbine connected to the generator. Res J Appl Sci Eng Technol 5:1544–1551
10. Razmjooy N, Khalilpour M (2015) A new design for PID controller by considering the operating points changes in hydro-turbine connected to the equivalent network by using invasive weed optimization (IWO) algorithm. Int J Inf Sec Sys Manage 4:468–475
11. Bandaghiri PS, Moradi N, Tehrani SS (2016) Optimal tuning of PID controller parameters for speed control of DC motor based on world cup optimization algorithm, parameters, 1:2
12. Razmjooy N, Madadi A, Ramezani M, Robust control of power system stabilizer using world cup optimization algorithm
13. Razmjooy N, Shahrezaee M, Solving ordinary differential equations using world cup optimization algorithm
14. Razmjooy N, Sheykhahmad FR, Ghadimi N (2018) A hybrid neural network–world cup optimization algorithm for melanoma detection. Open Med 13:9–16
15. Shahrezaee M (2017) Image segmentation based on world cup optimization algorithm. Majlesi J Electr Eng 11
16. Zaidi Z, Boudjema F (2010) Hybrid control and optimization of a plus-energy-house with DHWS. international renewable energy congress IREC, Sousse, Tunisia
17. Mellit A, Tina GM, Kalogirou SA (2018) Fault detection and diagnosis methods for photovoltaic systems: a review. Renew Sustain Energy Rev 91:1–17
18. Mohammadi K, Naderi M, Saghafifar M (2018) Economic feasibility of developing grid-connected photovoltaic plants in the southern coast of Iran. Energy 156:17–31
19. Mahmoudimehr J, Shabani M (2018) Optimal design of hybrid photovoltaic-hydroelectric standalone energy system for north and south of Iran. Renewable Energy 115:238–251
20. Firouzjah KG (2018) Assessment of small-scale solar PV systems in Iran: regions priority, potentials and financial feasibility. Renew Sustain Energy Rev 94:267–274
21. Hosseini SA, Kermani AM, Arabhosseini A (2019) Experimental study of the dew formation effect on the performance of photovoltaic modules. Renew Energy 130:352–359
22. Jamri MS (2009) Modeling and control of a photovoltaic energy system using the state-space averaging technique. Am J Appl Sci 1
23. Ishengoma FM, Norum LE (2002) Design and implementation of a digitally controlled stand-alone photovoltaic power supply. Nordic workshop on power and industrial electronics, pp 12–14
24. Brofferio S, Cristaldi L, Della Torre F, Rossi M (2010) An in-hand model of photovoltaic modules and, or strings for numerical simulation of renewable-energy electric power systems. 2010 IEEE workshop on environmental energy and structural monitoring systems IEEE, pp 14–18
25. Muljadi E, Butterfield CP (2001) Pitch-controlled variable-speed wind turbine generation. IEEE Trans Ind Appl 37:240–246
26. Chaudhari VA (2005) Automatic peak power tracker for solar pv modules using dspacer software. Maulana Azad National Institute of Technology. Master thesis of technology in energy. Bhopal: Deemed University
27. Hosseini H, Tusi B, Razmjooy N, Khalilpoor M (2011) Optimum design of PSS and SVC controller for damping low frequency oscillation (LFO). In: 2011 2nd International conference on control, instrumentation and automation (ICCIA). IEEE, pp 62–67

28. Woywode O, Güldner H (2001) Application of statistical analysis to DC-DC converters. IEEJ Tran Ind Appl 121:557–562
29. Hansen MO (2015) Aerodynamics of wind turbines. Routledge
30. Hirth L, Müller S (2016) System-friendly wind power: How advanced wind turbine design can increase the economic value of electricity generated through wind power. Energy Econ 56:51–63
31. Parada L, Herrera C, Flores P, Parada V (2018) Assessing the energy benefit of using a wind turbine micro-siting model. Renew Energy 118:591–601
32. Barr SM (2018) Optimization of tow-steered composite wind turbine blades for static aeroelastic performance
33. Chen T-Y, Cheng YL (2008) Global optimization using hybrid approach. WSEAS Trans Math 7:254–262
34. Razmjooy N, Khalilpour M, Ramezani M (2016) A new meta-heuristic optimization algorithm inspired by FIFA world cup competitions: theory and its application in PID designing for AVR system. J Control Autom Electr Syst. 1–22
35. Razmjooy MRN (2016) Model order reduction based on meta-heuristic optimization methods. In: 2016 1st international conference on new research achievements in electrical and computer engineering. 978–1–5090–2702–6/16/$31.00 ©2016 IEEE, Iran
36. Zhong J (2006) PID controller tuning: a short tutorial, class lesson), Purdue University
37. Kurokawa R, Sato T, Vilanova R, Konishi Y (2018) Closed-loop data-driven trade-off PID control design. IFAC-PapersOnLine 51:244–249
38. Hosseini H, Farsadi M, Khalilpour M, Razmjooy N (2011) Hybrid energy production system with PV array and wind turbine and pitch angle optimal control by genetic algorithm (GA)

Optimization Techniques in Intelligent Transportation Systems

Mehdi Ghatee

Abstract Intelligent Transportation Systems (ITS) refer to a range of transportation applications based on communication and information technology. These systems by the aid of modern ideas, provide comfortable, efficient and safe services for transportation users. They are located in the linkage of information technology, computer science, electrical engineering, system analysis, civil engineering, and optimization. They form a main branch of the smart cities and are fundamental for the development of countries. ITS applications, usually, use the capabilities of sensor networks, electrical devices, and computer processing units to deliver a service, however, they are not limited to the hardware devices. Instead, the modeling of transportation problems and solving them efficiently are more challenging problems. Especially, when they support good ideas for controlling the transportation systems or guiding some users. In this chapter, the application of optimization models for the transportation context are discussed. To this end, the models for data collection by a sensor network are needed. Then, for mining these data and extracting the necessary knowledge for transportation context awareness, some fundamental models such as regression analysis, frequent pattern mining, clustering or classification can be applied. These backgrounds are used to extend the appropriate models of ITS in an integrated architecture. To solve these models, the classical network and combinatorial optimization methods, simulation-optimization techniques, and metaheuristic algorithms will be explained. Then, we categorize the applications of these optimization models in the different subsystems of ITS architecture. The output of this investigation can be used to develop different ITS services for the urban and inter-cities networks.

Keywords Intelligent transportation systems · Traffic assignment · Network optimization · Mathematical programming · Metaheuristic algorithms · Sensor networks · Simulation optimization · Meta-model

---

M. Ghatee (✉)
Department of Mathematics and Computer Science, Amirkabir University of Technology, Tehran, Iran
e-mail: ghatee@aut.ac.ir
URL: https://www.aut.ac.ir/ghatee

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_4

1 Introduction

Intelligent transportation systems (ITS) have a tremendous impact on optimizing the benefits of transport infrastructure, travel, passenger and cargo safety, improving security, enhancing the quality of life and reducing living and environmental costs. These systems come together in a comprehensive architecture to provide interoperability between different departments and provide convenient services to traffic users. ITS architecture creates a soft infrastructure to facilitate and to communicate between subsystems. They also change the legalization and support the standards for infrastructures and services [1, 2].

On the other hand, the services of ITS are very extensive. Just as in some instances, they manage the traffic flows by modern sensors and controllers. They share some information for travelers. They collect some data from different detectors, counters, real-time images, smartphones, GSM antennas, etc. [3, 4]. Because of the big data that are collected by different transport services, the computational data mining approaches are exhaustively needed [5].

The data mining approaches can be divided into three parts. In the first part, we need to define an appropriate data pre-processing plan to clean data, to remove the outliers and noises, to complete the miss data, to remove irrelevant features, to reduce the dimensions and to visualize them [6]. Then, we can select a good subset of features to extract the knowledge from the samples. Sometimes, we also need to extract some new features for the same purpose [7]. In the second part, we apply different regressions, pattern recognizers, classifiers or cluster analyzers to find the relationships of the different components and elements [8]. The third part is making decisions. In the case of ensemble models, we also need decision fusion based on the outputs of the mentioned systems to integrate the decisions [9]. Because of the curse of dimensionality, the big-data analysis has been also focused by many researchers [10–13].

In addition to the data mining perspective, we need to optimize the usage of the existing facilities and to improve the quality of service levels. It is usually advisable to provide a robust mathematical model. Some of the obtained models can be solved by exact algorithms [14–17]. More models are not solvable in the limited times. In these cases, metaheuristic algorithms are very useful [18, 19]. In this chapter, we review some metaheuristic algorithms to solve these problems.

Metaheuristic algorithms can be classified into six parts. The first category is based on evolutionary computing. The most important example in this part is genetic algorithm [20]. The second category is tabu-based algorithms [21, 22]. In these algorithms, a tabu list is added to limit the searching space. The memetic algorithm [23] also combines an evolutionary algorithm with tabu list. The third category is based on statistical physics. For example, simulated annealing [24] is a famous metaheuristic that uses a Boltzmann machine to adapt with metal annealing. The fourth category is based on swarm intelligence. The ant colony optimization [25] and particle swarm optimization [26] can be cited as two important examples. The fifth category includes nature-inspired metaheuristic algorithms [27]. Hopfield neural network [28] and artificial bee colony (ABC) algorithm [29] are two famous methods in this category. The sixth category is based on human relationships. For example, the harmony search [30] mimics the improvisation of music players. Imperialist competitive algorithm [31] also uses the imperialistic competition among empires to take possession of their countries.

To study the effects of exact methods and metaheuristic algorithms in transportation problems, in this chapter, we first review some combinatorial algorithms which can be directly used for such problems [32]. Then, we present the application of metaheuristics such as particle swarm optimization that has been used to extract the features from ITS data [33]. They can also be combined with machine learning algorithms such as neural networks, support vector machines, deep networks to improve the quality of modern techniques in ITS implementations [34]. Finally, we give some mathematical programming models that are famous in ITS to show how one can use them to implement transport services. Then, the simulation-based methods for ITS will be discussed [34]. Furthermore, the application of optimization models in the user services of ITS architecture will be presented.

Since, in ITS services, both of the passengers and goods should be managed, different models for supporting the passengers and managing the commercial trucks should be taken into account. For example, the routing service is very different for a user who wishes to travel by own vehicle or by public transit. It is also different for the transportation of common goods or for hazardous materials [35]. Also, the results of such systems can be delivered to a specific user when the information is private or when the solution is specialized. It can be also broadcasted to public users when there is not a privacy challenge. Just as an example, the navigation by GPS data is a private service based on the iterative shortest path problem [36], but broadcasting the Level of Service (LOS) is a public service that can be provided based on the image processing.

Besides, the type of services are related to the wide range of technologies such as electrical boards, video surveillance, satellite-based remote data processing, drone monitoring, Internet of Things (IoT), connected vehicles and smartphones, see e.g., [6, 9, 37]. Because of the variety of these technologies, an integration plan can be used to make an appropriate decision. Thus, in this chapter, we give some information regarding the mathematical models for sensor fusion, data fusion and decision fusion for transport cases [38].

To categories these issues, in this chapter, we discuss on the nine subsystems of ITS [1] including Advanced Traffic Management System (ATMS), Advanced Travel Information System (ATIS), Commercial Vehicle Operations (CVO), Advanced Public Transport System (APTS), Vehicle Safety System (VSS), Emergency Management System (EMS), Construction, Maintenance and Repair of Infrastructure (CMRI), Data Archiving System (DAS) and Advanced Human-based Transportation System (AHTS). For each of these subsystems, we derive the different modeling approaches and present how these models can be solved by exact or metaheuristic algorithms. We discuss some models to optimize information system provider (ISP) to support all of the services [39]. We refer to some models that process the data and support the drivers by proposing several options. We also present some details about the development of a Decision Support System (DSS) for transportation problems, by the aid of machine learning approaches [35]. At a higher level, maybe an expert system can be extended to execute a necessary automatic process instead of humans [40]. Nowadays, such systems have been installed in autonomous vehicles widely. Since the different restrictions should be considered to evaluate the feasibility of the different solutions and we need to compare the solutions by at least one objective function [34], we review some models where some of their parameters or restrictions are predicted by appropriate regressions. Then, different linear, nonlinear and discrete programming problems can be followed to solve them. The most important objective functions in the ITS problems can be summarized as the following:

1. Improving safety [9]
2. Minimizing travel time [18]
3. Harmonizing the network flow [41]
4. Minimizing the transportation risk [42]
5. Minimizing the shipment cost [43]
6. Minimizing the installation cost [44].

To improve these objective functions, the different approaches have been used. For example, in [45], a nonlinear programming was used for parking pricing problem. In [19] a genetic algorithm was used to minimize the shipment cost. In [46] a simulated annealing algorithm was used to optimize the toll pricing. In [18] an ant colony was used to solve the routing problem. In [47], for signal setting in the intersections, the different traffic assignment models were used to predict the drivers' behavior with respect to the changes in the signal setting times and then a genetic algorithm was applied to find the best time setting for the signals of the intersections.

Finally, when the results are extracted from one optimization model, the outputs can be sent to the specific users or public users through various user interfaces such as smartphone applications, onboard devices in the vehicles, variable message signs, etc. In the meantime, the integrity of the vehicles, infrastructures, control centers and humans is important to present the results. In an architecture plan, usually, we can define the relationship and the data flow between these components [1].

Following this chapter, the reader can understand the good view about multi-disciplinary problems in the transportation field, understand how can define a sensor network to collect the necessary data from the network, mine the data to extract the necessary knowledge for transportation context awareness, learn how can model the different concepts and entities of ITS in an integrated form and practice to solve the corresponding models with powerful and effective optimization models including the exact and metaheuristic algorithms. The output of this investigation can be used to develop different ITS services for the urban and inter-cities networks.

2 Data Collection and Data Mining for ITS Developments

Intelligent transportation systems are based on a large network of sensors, identifiers, detectors, counters and other components of information gathering [48, 49]. A huge amount of video information is added to this data from video surveillance cameras and license plate recognition (LPR) cameras. Social networking data and mobile operators also play important roles here [6, 38]. These data streams require an appropriate process of data integration and data mining. In data mining textbooks such as [8], there are different techniques for data pre-processing and data processing. In the former, data cleaning techniques such as outlier detection, data retrieval, data visualization, and data fusion can be followed. For the latter, regression analysis, frequent pattern recognition, clustering, and classification can be implemented. Also, some optimization for feature extraction, feature selection, and feature clustering have been developed [12, 13]. In ITS contexts, because of the big data problems, some special attempts have been done to solve the problem with modern techniques [50, 51].

3 Network Optimization Models for ITS Developments

Network models have a great effect on transportation problems [52]. In Fig. 1, a comparison between three main paradigms for solving the transportation optimization problems is given. As one can see, transportation researchers continuously note to the "network optimization" models, and although in recent years the number

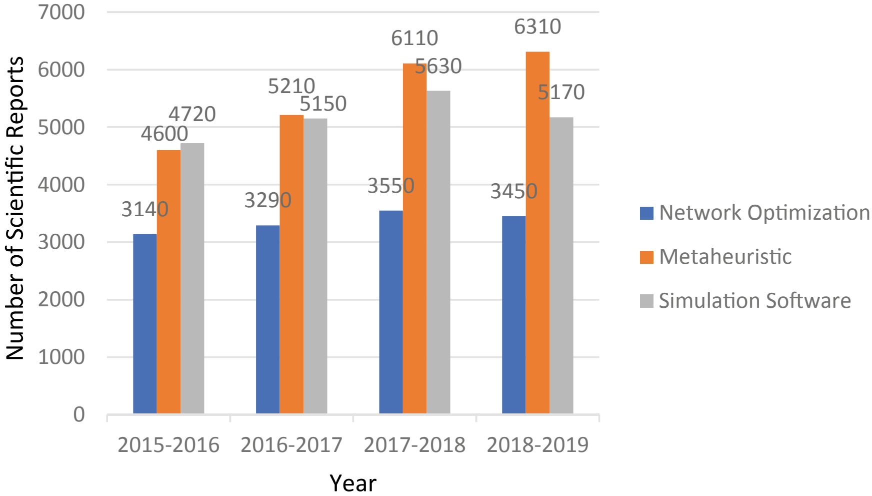


Fig. 1 Comparison between the different paradigms for transportation optimization problems based on Google scholar statistics

of scientific reports in this field has been less than "metaheuristics" and "simulation software" approaches, it has never diminished in importance. The most famous problems of network models are the shortest path problem, maximum flow problem, the minimum cost flow problem, and the multi-commodity flow problem [32]. Continuous and discrete network models with linear and nonlinear constraints are also very important [53]. Their algorithms have also polynomial, pseudo-polynomial and exponential complexities. This chapter explains some of the most important aspects of this issue. A wide range of these models has been proposed for the transportation of travelers and goods, project management and investment.

In what follows, we consider a network $G = (N, A)$ where N and A denote the set of nodes and links. n and m are considered as the number of nodes and links, respectively. We associate some labels for these networks and present some applicable models. For more study, one can refer to [54] or [55].

3.1 Shortest Path Problem

This problem is defined to determine the shortest path on a network. Each link (i, j) of the network, is associated with a label $c_{i,j}$ indicating the length, the cost, the travel time, etc. First, assume that there is a fix node in the network called the source (s), and the goal is to find the shortest path to all other nodes in the network. To model this state, the integer variable $x_{i,j}$ is considered for each link (i, j). Assume that $n - 1$ agents are located in the source node s. Each agent must find the shortest path from s to a particular node and stay there. The others continue their paths. Now consider the variable $x_{i,j}$ equal to the number of agents that pass through link (i, j). $\sum_{\{j:(i,j)\in A\}} x_{i,j}$ shows the number of agents that exit from i and $\sum_{\{j:(j,i)\in A\}} x_{j,i}$ displays the number of agents who enter in i. Then, for each of the nodes, except of the source node, one can state:

$$\sum_{\{j:(i,j)\in A\}} x_{i,j} - \sum_{\{j:(j,i)\in A\}} x_{j,i} = -1 \tag{1}$$

This means that every agents entering the node i goes out except of one who stays there. The agents try to minimize the traveling costs. Thus, when an agent stays in a node i, this means that, it finds the shortest path to i.

In addition, for the source node, the input is equal to $n - 1$ and the output is zero. On the other hand, the total lengths of the paths is equal to $\sum_{(i.j)\in A} c_{i.j} x_{i.j}$ and this value must be minimized. So the problem can be summarized as follows:

$$\min \sum_{(i,j)\in A} c_{i,j} x_{i,j} \tag{2}$$

$$\text{s.t.} \sum_{\{j:(i,j)\in A\}} x_{i,j} - \sum_{\{j:(j,i)\in A\}} x_{j,i} = \begin{cases} (n-1) & i = s, \\ -1 & i \neq s, \end{cases} \tag{3}$$

$$x_{i,j} \in Z^{\geq 0} = \{0, 1, \ldots\} \tag{4}$$

This problem is the base for all of the advanced traveler information systems. However, when a great number of travelers select the same path, the congestion on the path increases and this is not useful for traffic management systems. This problem can be solved by using the K-shortest path problem, iterative paths by penalty approach, defining some gate points and forcing the routes to cross the gate, defining the similarity or dissimilarity measures between the paths and minimizing the similarity and maximizing the dissimilarity between the candidate paths. These approaches have been used for providing reasonable paths for traffic assignment [56], multi-commodity flow problem [57], network design [44], and network controlling problems [41]. Some of the recent works on this model in intelligent transportation systems are as the following:

- Dynamic decision making for connected vehicles [58]
- Commercial intelligent navigation for transportation management [59]
- Path selection for emergency relief after disasters [60]
- Path planning by using Internet of Things (IoT) [61].

3.2 Maximum Flow Problem

This problem has a structure similar to the shortest path problem except associating an upper bound for each link (i, j) as the link capacity instead of link cost. In this problem, we consider two nodes as the source node (s) and the target node (t). The goal is to find the maximum flow that can be transmitted from the source node to the target node considering the link capacities. This model can be stated as the following:

$$\max v \tag{5}$$

$$\text{s.t.} \sum_{\{j:(i,j)\in A\}} x_{i,j} - \sum_{\{j:(j,i)\in A\}} x_{j,i} = \begin{cases} v, & i = s \\ 0, & i \neq s, t \\ -v, & i = t \end{cases} \tag{6}$$

$$x_{i,j} \in Z^{\geq 0} = \{0, 1, \ldots\}, \quad v \geq 0. \tag{7}$$

This problem is very useful for network planning and estimating the capacity of traffic networks. For example, in [62], a maximum-flow problem has been used to manage multiple incidents in traffic evacuation management. The following present some recent works based on maximum flow in ITS contexts:

- Dynamic route guidance system by parallel implementation [63]
- Finding the capacity of ride-sharing [64].

3.3 Minimum Cost Flow Problem

This problem is inherited from both of the shortest path and maximum flow problems and provides a complete structure for many flow problems in network optimization theory [65]. Many other network problems are also generalized from this problem. Here, we associate a cost $c_{i,j}$ and a capacity $u_{i,j}$ for each link (i, j). The least capacity $l_{i,j}$ can be also considered for this ink. A flow matrix $x = [x_{i,j}]$ with minimum cost $\sum_{(i,j)\in A} c_{i,j}x_{i,j}$ should satisfy the following:

$$\min \sum_{(i,j)\in A} c_{i,j}x_{i,j} \tag{8}$$

$$\text{s.t.} \sum_{\{j:(i,j)\in A\}} x_{i,j} - \sum_{\{j:(j,i)\in A\}} x_{j,i} = b_i, \quad i = 1, \ldots, n \tag{9}$$

$$l_{i,j} \le x_{i,j} \le u_{i,j} \;\; \forall(i, j) \in A \tag{10}$$

In this model, $x_{i,j}$ is the number of flows on the link (i, j). b_i is positive when node i is the supplier and is negative when i is the demander. For a transitive node, $b_i = 0$. This model can be extended for location problem [66] and dispatching [67] for example. This problem is also extended under uncertainty for public transportation network design [65]. Also, nonlinear objective functions and piecewise objective functions for this problem are very applicable. To see more, one can see [14–17, 32].

3.4 Assignment Problem

In this case, n trucks are assigned on m routes. The suitability and fitness of the trip j by truck i is represented by $c_{i,j}$. The purpose of this problem is to assign all the trucks to trips in the best settings. To model this problem, the network $G = (N_1 \cup N_2, A)$ is defined where N_1 is the set of the available trucks and N_2 is the set of trips. A includes the links between N_1 and N_2. The problem can be presented as the following network model:

$$\max \sum_{i=1,\ldots,n, j=1,\ldots,m} c_{i,j}x_{i,j} \tag{11}$$

$$\text{s.t.} \sum_{i=1,\ldots,n} x_{i,j} = 1, \quad j = 1, \ldots, m \tag{12}$$

$$\sum_{j=1,\ldots,m} x_{i,j} \ge 1, \quad i = 1, \ldots, n \tag{13}$$

$$x_{i,j} \in \{0, 1\} \; \forall i \in \{1, \ldots, n\}, \forall j \in \{1, \ldots, m\} \tag{14}$$

The first constraint ensures that each trip is done by a single truck. The second limitation imposes that each trip can be assigned to at most a truck. This problem is very useful for the assignment of the devices of ITS to the urban networks [66]. Also, in [35], the assignment of drivers, trucks, and roads have been discussed. Furthermore, in [68], a new airline stochastic fleet assignment problem with random passenger demands under risk aversion has been studied.

3.5 Multi-commodify Flow Problem

This model is also one of the generalizations of the minimum cost flow problem and can model multiple commodities transportation. This model appears in most of the issues involved in the transportation of materials, service networks, and telecommunications applications. If the transportation of a commodity does not any effect on the other commodities, then the problem can be decomposed into some minimum cost flow problems and each problem can be solved separately. But since the links are shared between the commodities, the problem cannot be easily solved [32]. The general form of this problem can be considered as the following:

$$\text{Min} \sum_{k=1,\dots,K} c^k x^k = \sum_{k=1,\dots,K} \sum_{(i,j)} c^k_{i,j} x^k_{i,j} \tag{15}$$

$$\text{s.t.} \sum_{\{j:(i,j)\in A\}} x^k_{i,j} - \sum_{\{j:(j,i)\in A\}} x^k_{j,i} = b^k_i, \quad \forall i \in \{1,\dots,n\}, \forall k \in \{1,\dots,K\} \tag{16}$$

$$\sum_{k=1,\dots,K} x^k_{i,j} \le u_{i,j}, \quad \forall (i,j) \in A \tag{17}$$

$$0 \le x^k_{i,j} \le u^k_{i,j}, \quad \forall k \in \{1,\dots,K\} \forall (i,j) \in A \tag{18}$$

In this model, $x^k_{i,j}$ is the flow of commodity k on the link (i,j). Also, $c^k_{i,j}$ represents the transportation cost of commodity k. The first series of constraints satisfies the supply-demand equivalency for commodity k. The second meets the link capacity. It can be simplified as the following:

$$\sum_{1\le k\le K} x^k_{i,j} + s_{i,j} = u_{i,j} \ \forall (i,j) \in A \tag{19}$$

where $s_{i,j}$ is the remaining capacity of the link (i,j). This model is very useful to predict the flow over the links in the urban networks [32]. This problem is also used to estimate the flow under congestion effects [15]. Solving this problem by metaheuristic algorithm and two applicable extensions of this model with fuzzy parameters and fractional objective functions have been stated in [15, 19, 57]. Some of the recent advantages of this model in ITS topics can be seen in the following:

- Multi-commodity location-routing problem [69]
- Routing based on congestion monitoring [70].

3.6 Network Problems with Side Constraints

In this type of problem, a basic problem such as the shortest path problem is considered and the functional constraints are added. For example, the shortest path problem can be extended by imposing a new constraint to restrict the travel time or travel cost with an upper bound. To solve these problems, we can use the Lagrangian relaxation to transfer the bothersome constraints to the objective function. In this case, the structure of the remained constraints is similar to the constraints of a problem that can be solved efficiently. Of course, it is necessary to determine the Lagrangian multipliers (coefficients) by using optimization methods such as gradient-based algorithms [32]. To present some details, return to the multi-commodity minimum cost flow problem (16)–(19). As one can note, the constraints (18) is bothersome constraints. By defining the non-negative Lagrangian multiplier $\lambda_{i,j}$ with respect to each constraint of (18), the objective function (16) can be represented as the following:

$$\text{Min} \sum_{k=1,\ldots,K} c^k x^k = \sum_{k=1,\ldots,K} \sum_{(i,j)\in A} c^k_{i,j} x^k_{i,j} + \sum_{(i,j)\in A} \lambda_{i,j} \left(\sum_{k=1,\ldots,K} x^k_{i,j} - u_{i,j} \right) \quad (20)$$

Now, we define the following Lagrangian sub-problem:

$$L\big(\{\lambda_{i,j} : (i,j) \in A\}\big) = \text{Min} \sum_{k=1,\ldots,K} \sum_{(i,j)\in A} c^k_{i,j} x^k_{i,j} + \sum_{(i,j)\in A} \lambda_{i,j} \left(\sum_{k} x^k_{i,j} - u_{i,j} \right) \quad (21)$$

s.t. (17) and (19).

For each settings of $\{\lambda_{i,j} : (i,j) \in A\}$, this problem includes K minimum cost flow problem which can be solved efficiently. Just, we need to find nonnegative multipliers $\{\lambda_{i,j} : (i,j) \in A\}$ to maximize $L\big(\{\lambda_{i,j} : (i,j) \in A\}\big)$. Because $L\big(\{\lambda_{i,j} : (i,j) \in A\}\big)$ is the lower bound of the original objective function (16) and by maximizing this function, we expect to find the optimal solution for (16). If $\lambda^*_{i,j}$ denotes such optimal solution, the problem (22) can be solved to obtain a feasible solution of the multi-commodity flow problem. When the objective function (16) with respect to this feasible solution is equal to $L\big(\{\lambda^*_{i,j} : (i,j) \in A\}\big)$, the obtained solution is optimal solution of problem (16)–(19) and the problem can be terminated. In this case, we have $\lambda^*_{i,j}\left(\sum_k x^k_{i,j} - u_{i,j}\right) = 0$ for all of the links, which is a representation of the complementary slackness condition. See e.g., [32], page 606 for technical points. To obtain $\lambda^*_{i,j}$, the following iterative descent gradient method can be used:

$$\lambda_{i,j}^{t+1} = \lambda_{i,j}^{t} + \theta^t \nabla_\lambda L\big(\{\lambda_{i,j} : (i, j) \in A\}\big), \quad \forall t \in \{0, 1, ...\}, \forall (i, j) \in A \tag{22}$$

where θ^t is the step-size and $\lambda_{i,j}^{0}$ is a random or an appropriate initial solution. This can be simplified as:

$$\lambda_{i,j}^{t+1} = \lambda_{i,j}^{t} + \theta^t \left(\sum_{k=1,...,K} x_{i,j}^{k} - u_{i,j} \right), \quad \forall t \in \{0, 1, ...\}, \forall (i, j) \in A \tag{23}$$

The step-size is very important. $\theta^t = \frac{1}{t}$ is a good option for some real cases. Also, it is necessary to check non-negativity of $\lambda_{i,j}^{t+1}$. When this variable is not non-negative, usually it is better to substitute this variable with zero. To see modern approaches to define θ^t one can refer to [32]. Some applications of Lagrangian relaxation in transport studies have been also presented in the literature. Just as an instance, in [71] a new traffic signal optimization has been proposed to minimize fuel consumption and emission by the aid of the Lagrangian relaxation approach.

3.7 Traffic Assignment Problem

The traffic assignment model is one of the main issues in transportation studies [56, 72]. This is influenced by factors such as traffic zones, demands and traveler's utilities. Network managers are always seeking to find flows to do some control to raise the level of service. Also to compare some applicable plans, they can simulate the plans by traffic assignment models to predict the new flows for each plan. Then the rank of each plan can be evaluated and the best one can be selected [56, 73]. To implement this issue, it is necessary to analyze the geographical areas. The most important areas with a sufficiently great population and the same socio-cultural context, are referred to as traffic zones. For each zone, an important node can be selected to assign the number of zone populations to the corresponding node. Apart from these nodes, the network contains some nodes and links corresponding to the intersections and passages. In most cases, in addition to these points, some virtual nodes are added to better represent the shape of the transportation network.

Selecting the number and the size of zones is also dependent on a balance between the simplicity and the accuracy of the traffic assignment model. That is if we consider a small number of zones with great areas, the analysis is simpler and vice versa. After balancing, some steps should be followed. In the first step, by collecting annual data or aggregating daily telecommunication data, the number of travels can be determined between traffic zones. For example, in [74], the demands have been estimated by using cellular probe data. Then, based on the sample size, the actual travel times between these points should be predicted. Then, the number of users who wish to travel on a path is estimated. In this step, the combination of the following criteria can be minimized:

- Total travel time in the entire network
- Total mileage by total vehicles
- The amount of fuel consumption
- The amount of environmental pollutants.

But it is difficult to estimate the behavior of users for path selection. Meanwhile, there has been a lot of effort on the route selection models. In this regard, equilibrium models are the most important analytical tools to estimate the user paths, see e.g. [75] for a review on network equilibrium models. These models assume that the users select the paths based on their perception of the network. To present the perception, the precise models [76], uncertain models [77] and imprecise models [78] can be followed.

In the first type, which can be referred to as the user equilibrium, the model assumes that each user has an accurate understanding of the shortest path. This requires understanding the length of the paths, how the users select the paths and where the incidents of the network are. These models also assume that all users have the same behavior and selection model. Although these assumptions do not satisfy usually, they have been used in a large number of assignment models because of their simplicity in implementation [75]. Some of the recent researches on this model can be followed in:

- System optimal and user equilibrium time-dependent traffic assignment model [79]
- A risk-averse user equilibrium traffic assignment model [80]
- A bi-objectives dynamic user equilibrium traffic assignment [81]
- Achieving proportionality in user equilibrium traffic assignment [82]
- A ridesharing user equilibrium [83]
- Traffic signal control under dynamic user equilibrium model [84].

In the second model, which is entitled to the stochastic user equilibrium, the uncertainty in users' perception is considered. In these cases, the error in the users' perception can be implemented by some random variables with the normal or other probability distributions [85]. The term "random error" is interpreted as a misunderstanding of travel time in the network based on inaccurate knowledge of passengers from network conditions [86]. In the literature, the following are mentioned to interpret the random error term:

- Implementing properties of the route that are not considered in the survey
- Computational errors in the variables used in mathematical models
- Personality differences in path selection factors
- Internal difference between travels
- Level of information about network and service level.

By taking these concepts into account, one can obtain a path-selection function for network users. In such functions, instead of a single path, multiple paths are extracted for each user and based on a probability distribution, the users are assigned to the different paths. To present some details, assume the set of origin-destination zones

$\Gamma = \{(o, d)\}$. Let $P_{o,d}$ includes all of the paths from the origin zone o to destination zone d. Let the traveling time of a path $p \in P_{o,d}$ can be represented as the following:

$$c_p(x) = FT_p(x) + RT_p(x) \tag{24}$$

where $FT_p(x)$ and $RT_p(x)$ are the free travel time (traveling time when there is not any congestion) and the random travel time of path p. In both of these parameters, the flow over the links (x) are considered fixed. Now, to distribute the users who wish to travel between the pair of origin-destination (o, d), one can define the following probability distribution for the path p:

$$\text{Prob}_p(x) = \text{Prob}\{c_p(x) \le c_q(x),\ \forall q \in P_{o,d}\} \tag{25}$$

If $\text{dem}_{o,d}$ denotes the number of traveling demand between (o, d), the number of the users who wish to select path p for their travel can be estimated as the following:

$$\begin{aligned} f_p(x) &= \text{dem}_{o,d}.\text{Prob}_p(x), \quad \forall p \in P_{o,d} \\ x &= (x_{i,j})\ \mathit{showes\ vector\ of\ link\ flows}. \end{aligned} \tag{26}$$

When $RT_p(x)$ is represented by a Gumbel distribution with parameter θ, Eq. (26) can be simplified as the following [87]:

$$f_p(x) = \text{dem}_{o,d}.\frac{\exp\big(-\theta.FT_p(x)\big)}{\sum_{q\in P_{o,d}} \exp\big(-\theta.FT_p(x)\big)}, \quad \forall p \in P_{o,d} \tag{27}$$

This model is referred as logit-based traffic assignment model. The path-choice model (28) is the solution of the following nonlinear programming problem:

$$\min z = \frac{1}{\theta} \sum_{(o,d)\in\Gamma} \sum_{p\in P_{o,d}} f_p(x) Ln\big(f_p(x)\big) + \sum_{(i,j)\in A} \int_0^{x_{i,j}} c_a(v)dv \tag{28}$$

$$\text{s.t.} \sum_{p\in P_{o,d}} f_p(x) = \text{dem}_{o,d}\ \forall (o, d) \in \Gamma \tag{29}$$

$$\begin{aligned} \sum_{(o,d)\in\Gamma} \sum_{p\in P_{o,d}} f_p(x)\delta_p(i, j) = x_{i,j}, \quad &\forall (i, j) \in A, \\ &\delta_p(i, j) \in \{0, 1\}, \\ \delta_p(i, j) = 1 \iff &(i, j)\ \mathit{belongs}\ p \end{aligned} \tag{30}$$

$$f_p(x) \ge 0, \quad \forall p \in P_{o,d},\ (o, d) \in \Gamma. \tag{31}$$

To see the proof, one can see [87]. Some algorithms for finding the logit-based stochastic user equilibrium assignment can be pursued in [88]. In [89], a multi-criteria logit-based traffic equilibrium assignment model has been proposed that uses the advanced traveler information system (ATIS) to guide the users. Toll pricing problem under logit-based stochastic user equilibrium has been also proposed by [90]. Some extensions of logit model can be addressed as the generalized nested logit model [91], the paired combinatorial logit model [92], the generalized nested logit model [91] and logit-based stochastic user equilibrium with elastic demand [93]. On the other hand, by using different probability distributions for modeling the error in the driver's perception, more choice-model can be defined. Some of these extensions do not support closed forms. For example, in [94], by using the normal distribution, a Probit-based time-dependent stochastic user equilibrium has been studied. As another example, by using Weibull distribution, the weibit stochastic user equilibrium model has been proposed by [95] that addresses the identically distributed assumption.

Another direction of research is the way to consider all of the paths between (o, d). Instead of all path traversing, we can find a set of reasonable paths by using K-shortest paths [96], dissimilar paths [97], multi-objective shortest path [98], dynamic shortest paths [99], stochastic shortest paths [100] and fuzzy shortest path [101]. Also under uncertainty on the traveling demands, one can note to [56, 102].

3.8 Sensor Network Problem

Sensor networks have great effects on intelligent transportation systems [103]. There are two basic types of models to consider these problems. The first is to design sensor networks. In fact, the designer is looking for a suitable location for sensors. When the number of sensors is high, the sensors can be randomly distributed over the network and their results will be combined to detect some incidents or to manage the network [104]. In the case of necessity to find accurate locations for the sensors, several appropriate criteria can be defined to evaluate their locations. For example, in [49] multi-objective programming has been used to optimize a sensor network. In [48] a review on the location of sensors and controllers has been stated.

On the other hand, due to the increased volume of traffic in metropolitan areas, the estimation of supply and demand between traffic zones has received much attention. It is very costly and time consuming to determine these values between different areas of the urban network by traffic counters and sensors on links. To solve this problem, different location models have been proposed to determine the minimum number and location of counters. One of the tools used in this field is car license plate recognition cameras. Because of the good performance of these cameras, there are several models to find their locations in urban networks. In one of these models, by adding budget constraints to the location problem, the maximization of the path coverage is considered. In another model, the goals are maximizing the number of scanned paths and minimizing the number of cameras. For more details, one can see [105].

3.9 Network Control Problem

The route selection and traffic control problems are dependent together. The control tools such as traffic lights should be designed related to traffic predictions on different routes. On the other hand, the traffic flows are related to the controlling parameters such as signal timings. To model this relationship, two close-up ideas have been presented by researchers including bi-level programming [106] and mutual consistency [107].

This first method is analytically related to the nonlinear programming and differential equations. Let $y \in S$ be the vector of independent variables and $x(y)$ be the dependent variables. In the controlling applications, y includes controlling variables and $x(y)$ is the flow with respect to y. Usually, we need to minimize the following controlling problem:

$$\min F(x(y), y) \tag{32}$$

$$\text{s.t. } y \in S. \tag{33}$$

where the following conditions are met:

$$f(x, y)(z - x) \geq 0, \quad \forall x, z \in K(y) \tag{34}$$

In these constraints, $K(y)$ shows the feasible solutions for the dependent variables. Sometimes, this set is modeled as the following:

$$K(y) = \{x : g(x, y) \geq 0,\ h(x, y) = 0\} \tag{35}$$

Really, Eq. (35) present the relationship between independent and dependent variables, while the constraints given in Eq. (36) limit the values of dependent variables with respect to any fixed value of y. As indicated by [108], for a fixed y^0, the solution x^0 satisfies Eqs. (35)–(36) if there are multipliers π^0 and μ^0 that

$$f\big(x^0, y^0\big) - \nabla g\big(x^0, y^0\big)\pi^0 - \nabla h\big(x^0, y^0\big)\mu^0 = 0 \tag{36}$$

$$\pi^{0\,t} g\big(x^0, y^0\big) = 0 \tag{37}$$

$$\pi^0 \geq 0 \tag{38}$$

Under some assumptions on $\left(x^0, y^0\right)$, Eqs. (37)–(39) can be approximated by the following first-order approximation:

$$x(y) \approx x^0 + \nabla x\left(y^0\right)\left(y - y^0\right) \tag{39}$$

where $\nabla x\left(y^0\right) = [J_x]^{-1}\left[-J_y\right]$ and J_x and J_y are Jacobian of the following nonlinear system with respect to x and y:

$$\begin{pmatrix} f(x, y) - \nabla g(x, y)\pi^t - \nabla h(x, y)\mu \\ \pi^t g(x, y) \\ h(x, y) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{40}$$

By substituting Eq. (40) in problem (33)–(34), we obtain a model just with respect to y. When we solve this new model, we can obtain $x(y)$. For more details, see [108].

In the second method, the control variables and equilibrium flows are independently defined and they are solved iteratively by assuming fixed values for the other. For example, optimal controlling has been defined for the constant link flows and the traffic assignment has been solved based on the constant control variables. These two problems are solved iteratively to achieve an expected convergence. Meanwhile, by simulation software, one can imitate the real traffic and so it plays a significant role in implementing traffic controlling variables. The simulation software can be also used to predict the flows by solving one of the traffic assignment models. Since the result of a control variable with the help of a simulator can be easily evaluated, such methods are very useful. On the other hand, the simulators determine link flows and so the variations in the problem can be simulated before any change in the problem. On the other hand, when simulation software is used to optimize a problem, due to the lack of derivativeness conditions, gradient techniques are not very useful. Instead, the gradient can be estimated numerically or metaheuristics can be used to optimize the problem without the necessity to the gradient [47]. In the first approach, that does not guarantee to find the global minima, the following iterative equation is used:

$$\psi^{k+1} = \psi^k + \alpha \nabla_\psi z \tag{41}$$

where $z = PI$ is the performance index of the network with respect to the controlling variable ψ and k counts the time step and α is a step-size. For bth component of ψ, the partial derivation can be estimated as the following:

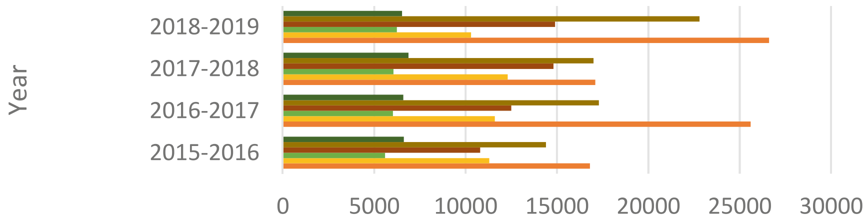
$$\begin{aligned} (\nabla_\psi z)_b &= \frac{\partial z(\psi)}{\partial \psi_b} = \frac{\Delta_z}{2\Delta_b} \\ \Delta_z &= z(\psi_1, \ldots, \psi_b + \Delta_b, \ldots, \psi_n) + z(\psi_1, \ldots, \psi_b - \Delta_b, \ldots, \psi_n) - 2z(\psi_1, \ldots, \psi_b, \ldots, \psi_n) \end{aligned} \tag{42}$$

where Δ_b denotes a small perturbation on the bth component of ψ.

In addition, the performance index of the network with respect to this perturbation are denoted with $z(\psi_1, \ldots, \psi_b + \Delta_b, \ldots, \psi_n)$ and $z(\psi_1, \ldots, \psi_b - \Delta_b, \ldots, \psi_n)$.

In the second idea, which has been followed more, direct metaheuristic algorithms such as simulated annealing [109], genetic algorithm [110], etc. are used on the simulation model. To see a comparison between the different metaheuristic algorithms in transportation contexts, one can see Fig. 2. This figure shows that genetic algorithm, neural network and partical swarm optimization are very usable in transportation problems.

Sometimes, the results of the simulation software have been also considered to develop a meta-model and the optimization has been applied on the meta-model. In these relevant researches, after collecting the simulation data, one can define a meta-model such as a spline interpolator, regression, or neural network, and the process of optimization is performed on the meta-model. For example, in [34] different machine learning algorithms are used to produce the meta-model and nonlinear optimization is used to optimize the performance index based on the results of the meta-model. Again metaheuristic methods can be customized with meta-models to find optimal global solutions for network controlling problems.



	2015-2016	2016-2017	2017-2018	2018-2019
Tabu Search	6620	6590	6870	6520
Neural Network	14400	17300	17000	22800
Particle Swarm Optimization	10800	12500	14800	14900
Ant Colony Optimization	5590	6020	6050	6240
Simulated Annealing	11300	11600	12300	10300
Genetic Algorithm	16800	25600	17100	26600

Number of Scientific Report

Tabu Search · Neural Network · Particle Swarm Optimization · Ant Colony Optimization · Simulated Annealing · Genetic Algorithm

Fig. 2 Comparison between the different metaheuristic approaches for transportation optimization problems based on Google scholar statistics

4 Simulation Optimization for ITS Developments

To investigate the performance of complex systems, analytical models become very complicated. Because of the complex and stochastic characteristics of these systems, the simulation is used to analyze their measures [111]. Recently, the optimization of system parameters with a simulation model was proposed. Such solution-processes are referred to as simulation optimization. Heuristic optimization methods showed better performance in this area because they don't need to differential equations [112, 113]. Ceylan and Bell [110] used a genetic algorithm on the simulation model directly to optimize the signal settings in urban networks with stochastic user equilibrium jointly. But simulation of large-scale networks is very time-consuming. To overcome the limitation of simulation, meta-modeling was first proposed by Blanning in 1975 [114]. The meta-model is a shape-function that relates the input vectors to the output vectors. They can be used to predict without the necessity of experimented events. Among, the different meta-models, we present some details about the fuzzy regression [115] and artificial neural networks [40]. When the meta-model is created, a metaheuristic should be selected to optimize the parameters. A genetic algorithm is a good option, while it can drop in local optimal points in the case of optimizing a non-convex objective function. Exceeding the size of the initial population is a good idea but isn't sufficient. Tabu search proposed by Glover [21, 22], is a better method that tries to perform a hill-claiming mechanism to get a global extreme point. Thus, we use a Tabu search to describe the process of simulation-optimization in what follows.

4.1 Meta-Models

Meta-models play important roles in implementing the effects of independent variables on dependent variables. These tools are used to evaluate dependent variables when independent variables change. For example, the cost, price or risk can be implemented by meta-models in terms of demand, supply, maintenance costs, failure rate, and so on. There are two important kinds of meta-models. In the first one, the meta-model can model the uncertainty in the parameters. The second kind recognizes the reality with crisp thresholds or by mathematical functional forms. In the next, we introduce a fuzzy regression for the first type and a neural network for the second type.

- Fuzzy Regression

Regression analysis is one of the most widely used statistical techniques for determining the relationship between variables to describe or to predict some stochastic phenomena. Fuzzy regression analysis is an extension of the classical regression analysis, in which some elements of the model are represented by fuzzy numbers. This category consists of linear and non-linear versions. Generally, by composing neural network and fuzzy quantities, we can obtain nonlinear fuzzy regression to learn any relation between the parameters with more accuracy and by removing the effects of outliers [115]. For simplicity and without loss of generality, we use fuzzy linear regression in this part to predict the effect of ITS parameters on the performance index. The fuzzy linear regression (FLR) model can be stated as follows:

$$\tilde{Y}_i\left(\tilde{X}_i\right) = \tilde{A}_0 + \tilde{A}_1 \otimes \tilde{X}_{i,1} + \cdots + \tilde{A}_n \otimes \tilde{X}_{i,1} = \sum_{j=0,\ldots,n} \tilde{A}_j \otimes \tilde{X}_{i,j},\ i = 1, \ldots, m \tag{43}$$

where $\tilde{X}_{i,j}$ is the fuzzy value of the jth independent variable in the ith observation and $\tilde{X}_{i,0} = 1$. The multiplication of $\otimes$ can be defined exactly or approximately. The problem of fuzzy regression analysis is to find the fuzzy parameters $\tilde{A}_0, \ldots, \tilde{A}_n$ such that $\tilde{Y}_i\left(\tilde{X}_i\right)$ is close to $\tilde{Y}_i$. Thus, we need to minimize the following:

$$\min \sum_{i=1,\ldots,m} \left|\tilde{Y}_i\left(\tilde{X}_i\right) - \tilde{Y}_i\right|^p = \sum_{i=1,\ldots,m} \left|\sum_{j=0,\ldots,n} \tilde{A}_j \otimes \tilde{X}_{i,j} - \tilde{Y}_i\right|^p \tag{44}$$

where p is a positive integer number. Since the parameters are considered as fuzzy triangular numbers, we can estimate $\tilde{A}_j\tilde{X}_{i,j}$ as a triangular number [116]. By considering just triangular fuzzy numbers $\tilde{A}_j = \left(A_j^c, A_j^L, A_j^R\right)$ and $\tilde{X}_{i,j} = \left(X_{i,j}^c, X_{i,j}^L, X_{i,j}^R\right)$ with the centers A_j^c and $X_{i,j}^c$, the left spreads A_j^L and $X_{i,j}^L$ and the right spreads A_j^R and $X_{i,j}^R$. Let these fuzzy numbers are positive. The multiplication can be estimated as the following:

$$\tilde{A}_j \otimes \tilde{X}_{i,j} \cong \left(A_j^c X_{i,j}^c, A_j^c X_{i,j}^L + X_{i,j}^c A_j^L, A_j^c X_{i,j}^R + X_{i,j}^c A_j^R\right) \tag{45}$$

Thus, the objective function (32) with respect to the fuzzy observations $\tilde{Y}_i = \left(Y_i^c, Y_i^L, Y_i^R\right)$, can be represented as the following:

$$\begin{aligned} &\min \sum_{\{i=1,\ldots,m\}} \left|\left(OBJ_i^c, OBJ_i^L, OBJ_i^R\right)\right|^p \\ &OBJ_i^c = \sum_{j=0,\ldots,n} A_j^c X_{i,j}^c - Y_i^c, \\ &OBJ_i^L = \sum_{j=0,\ldots,n} A_j^c X_{i,j}^L + X_{i,j}^c A_j^L + Y_i^R, \\ &OBJ_i^R = \sum_{j=0,\ldots,n} A_j^c X_{i,j}^R + X_{i,j}^c A_j^R + Y_i^L \end{aligned} \tag{46}$$

that is the sum of the norms of fuzzy numbers. To solve this problem, [115] proposed a multi-objective programming approach. We can also do similar to [117] by defining some fuzzy goals for each of the objective functions and trying to receive fuzzy goals. Such models can be solved by metaheuristic algorithms simply. After solving this model and finding $\tilde{A}_j$, we can optimize the performance index of the corresponding intelligent transportation system. In what follows, we consider $p = 1$, and try to solve the following 3-objective function model instead of (34) by using Rommelfanger's approach [118] to minimize the most possible case, to maximize the spread of the optimistic area and to minimize the spread of the pessimistic area:

$$\begin{cases} \min z_1 = \sum_{\{i=1,\dots,m\}} \sum_{j=0,\dots,n} |A_j^c X_{i,j}^c - Y_i^c| \\ \max z_2 = \sum_{\{i=1,\dots,m\}} \sum_{j=0,\dots,n} |A_j^c X_{i,j}^L + X_{i,j}^c A_j^L + Y_i^R| \\ \min z_3 = \sum_{\{i=1,\dots,m\}} \sum_{j=0,\dots,n} |A_j^c X_{i,j}^R + X_{i,j}^c A_j^R + Y_i^L| \end{cases} \tag{47}$$

Also, the variables $\tilde{X}_{i,j} = \left(X_{i,j}^c, X_{i,j}^L, X_{i,j}^R\right)$ should meet some constraints. In what follows, we present a Tabu search algorithm to solve this model.

- Neural Network

Artificial neural networks are famous tools to learn the relationship in real phenomena. Among them, a multilayer perceptron has been used as a meta-model in many studies [40]. A multilayer perceptron with enough number of neurons in a hidden layer can estimate any functional form in the shallow and deep constructions [37, 40]. Although there are different optimization methods to train a perceptron network, the backpropagation algorithm has been proved to be powerful for many experimental studies. In what follows, we show how a multilayer perceptron can be used to define a meta-model for a real problem.

4.2 Metaheuristic Optimizer

After getting the data from a simulation software, and developing a meta-model such as fuzzy regression, neural network, etc., we need to develop an optimizer. As Fig. 2. shows, the different metaheuristic algorithms are used in transportation studies. Now, we present some bases of Tabu search. Tabu search is a general heuristic search procedure devised for finding a global maximum of a function. The modern version of the algorithm was developed by Glover [21, 22] for large-scale combinatorial optimization problems. It has a flexible memory to retain the information about the previous steps of the search, using it to create and to exploit new solutions in the search space. Initially, Tabu Search (TS) choose a random feasible solution x_c with corresponding objective function θ_c. The optimal solution x_{best} and its value θ_{best} are supposed to equal to x_c and θ_c, respectively. Also, it selects values for the following parameters: P (probability threshold), N (number of neighbors), $N_{\max}$ (the maximal number of iterations allowed without improvement), NNI (number of non-improved iterations) = 0.

A step of TS starts with a current solution y_c and by applying a simple modification to x_c, it produces N feasible solutions $x_1, \ldots, x_N$ which are named constructed neighbors. We use the following procedure to choose the neighbors at each iteration:

- For $t = 1, \ldots, N$ and $i = 1, \ldots, n$, select a random number $r \sim Uniform(0, 1)$.
- If $r < P$, then, ith component of x_t is equal to $x_c(i)$, otherwise, select randomly an integer $l \neq x_c(i)$ and let $x_t(i) = l$.

Let $x_1, \ldots, x_N$ be sorted with respect to their objective functions $\theta_1, \ldots, \theta_N$ in descending order. This list is used to move the solution. To avoid a local maximum in search space, the move to x_1 is applied even if x_1 is worse than y_c, but if $\theta_{best} < \theta_1$, set NNI = 0, else set NNI = NNI + 1. However, this can cause the cycling of the search. To avoid cycling as much as possible, a Tabu list is introduced. The Tabu list stores all Tabu moves that are not permitted to be applied to the present solution. The moves stored in the Tabu list are those carried out most frequently and recently. Therefore, a move is classified as Tabu or not, when some criteria called Tabu restrictions, are employed. The use of a Tabu list decreases the possibility of cycling because it prevents the return within a certain number of iterations to a solution visited recently. After a specified duration (Tabu list size or Tabu period), a Tabu move is free to be visited again and removed from the Tabu list. This loop is repeated until a NNI $< N_{\max}$. Now, this is applied to any optimization model.

4.3 Details of Implementation of Simulation Optimization

In this part, we consider a problem with n independent variables denoted with $X = (x_1, \ldots, x_n)$. Based on the restrictions, these variables should satisfy $X \in F(X)$. For example, the feasible set $F(X)$ can be represented by a linear system such as $AX = b$ or $X \in \{0, 1\}^n$. Now, the objective function should be implemented. As we discussed deeply, to implement an objective function, the fuzzy or crisp variables can be considered and then the appropriate meta-model can be defined.

Let the variables be crisp. For example, when the question is about the variable speed limits in the urban highways, we can define x_i ($i = 1, \ldots, n$) as the speed in street i. Then, the performance index of the network can be defined in terms of safety, total travel time, etc. Now, we can create m scenarios $X_j = \left(x_{j,1}, \ldots, x_{j,n}\right)$ ($j = 1, \ldots, m$) with different speeds $x_{j,i}$ for highway i in the feasible set $F(X)$. These speeds can be defined for the highways in a simulation software such as AIMSUN, see e.g. [3, 34]. Then the performance index of the network in terms of the important criteria can be computed by the simulation software.

Let $z_1(X_j), \ldots, z_k(X_j)$ are obtained with respect to any speed variables $X_j = (x_{j,1}, \ldots, x_{j,n})$. Really, we need a meta-model to recognize the relation between X_j and the performance indices $[z_1(X_j), \ldots, z_k(X_j)]$. As presented earlier, a multilayer perceptron can be defined in which the vectors $X_j (j = 1, \ldots, m)$ are inputs and the vectors $[z_1(X_j), \ldots, z_k(X_j)]$ are the corresponding outputs. Denote $N(X_j)$ as a neural network that returns the vector of performance indices for any speed variables X_j.

Now, we need to define an optimizer on $N(X_j)$. By defining $\theta(X_j) = g(N(X_j))$, we can obtain a real-value function $g(N(X_j))$ to be optimized. For example, by using weighting parameters $w = (w_1, \ldots, w_k)$, we can present the following function:

$$\theta(X_j) = \sum_{l=1,\ldots,k} w_l O_l(N(X_j)) \tag{48}$$

where $O_l(N(X_j))$ is the lth output of $N(X_j)$. We are ready to apply Tabu search on $\theta(X_j)$ to find the best speed variables $X_j = (x_{j,1}, \ldots, x_{j,n})$.

Although, we present these details for variable speed limit problem in the urban networks, all of these steps are generally applicable. Just, we need a simulation software to evaluate the goodness of the scenarios in some meaningful criteria. Some examples of simulation-optimization techniques for transportation studies are presented in the following:

- Optimization of controlling variables [36]
- Optimization of toll pricing problem [34]
- Green intermodal transportation problem [119]
- Sustainable transportation planning [120]
- Collaborative urban freight transportation [121]
- Parking space management [122]
- Dynamic disaster relief distribution [123]
- Road surface maintenance scheme [124]
- Rescheduling the train traffic in uncertain conditions [125]
- Dynamic multimodal freight routing [126]
- Scheduling in stochastic freight transportation [127]
- Managing mass casualty incidents [128]
- Urban transportation demand management [129].

5 Advanced Optimization Solutions for ITS Services

Intelligent transportation systems play a key role in improving the performance of smart cities and intercity roads. But these systems work together and serve as a puzzle. Really, the development of unbalanced intelligent transportation systems often leads to failed projects and only consumes high costs. It is proved that an integrated plan for ITS should be followed in each country for ITS development. In Fig. 3, a pyramid

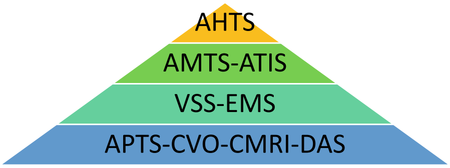


Fig. 3 Pyramid of ITS architecture in Iranian national plan. Adapted from its.aut.ac.ir/nits

of ITS services in Iran national plan is presented where these services are commonly used in the different ITS architecture around the world, see e.g., [1].

The top of this pyramid is an advanced human-based transportation system (ATMS). This service is supported by two branches of advanced traffic management system (ATMS) and advanced travel information system (ATIS). The vehicle safety system (VSS) and emergency management system (EMS) are in the third level of importance. Finally, the infrastructure services are considered including advanced public transportation system (APTS), commercial vehicle operations (CVO), construction, maintenance and repair of infrastructure (CMRI) and data archiving system (DAS). In the following, nine-dimensional subsystems of this architecture are presented and we give some information about the optimization processes in these subsystems.

5.1 Models of Advanced Human-Based Transportation System (AHTS)

This service serves different people. These people can be healthy or damaged. For example, appropriate services for the elderly, children, the blind, the deaf and the disabled are defined here [130]. For example, users' safety services, especially for vulnerable users are supported by this system [131]. Also, we need to prioritize humans at intersections to cross safely. Really, we need to consider human factors for extending ITS through the urban and intercity networks [132]. Some of the most important services for vulnerable users are as the following:

- Users detection using v2x communications [133]
- Pedestrian detection by machine learning approaches [134]
- Collision avoidance systems [135]
- Improving safety and mobility [136]
- Tracking vulnerable road users [137].

In Table 1, some applications of optimization models in this system have been presented.

Table 1 The application of optimization models in AHTS

User service	Sample reference	Description and process
Alert the user when crossing the street	[138]	Alerting drivers approaching a pedestrian crossing by utilizing sensors to detect a movement. The system may be calibrated to optimize the detection rate, minimizing false alarm and no detection
Increased rear view of the biker	[139]	A vision-based system for rear-end collision detection has been proposed to increase the safety of motorcyclists and to minimize their road fatalities
Safety warning for blind people	[140]	An indoor navigation support for blind people has been proposed to identify the objects within modeled indoor environments. The user has been supported by a text-to-speech engine. The hardware ergonomics of the module should be optimized
Guidance system for vulnerable people	[39]	For an unknown environment, a navigation system tailored to the special needs of blind people has been developed. The system uses fast routing algorithms generating lists of maneuvers, suitable positioning tools and reliable map-matching algorithms for route guidance instructions
A priority system for vulnerable users	Although vulnerable users, such as the elderly or children or the blind vulnerable users usually need more time to cross the street compared with the ordinary people, there is still no industrial development priority system for the vulnerable users	

5.2 *Models of Advanced Traffic Management System (ATMS)*

This system controls the intersections, highways and arterial streets, along with the entrance and exit of the passageways, and creates a comprehensive monitoring on the traffic flow and violations. The main modules of this system are shown in Fig. 4. The services of this system depend on the type of infrastructure. Of these, all of these services are developed based on data mining. Thus, network monitoring is the main core of all of the components of ATMS. For example, an electronic toll collection module is one of the most important modules of ATMS that uses various automotive identification technologies to determine when the vehicle enters a zone and how long it stays there. The toll will be defined based on these data. In Table 2, some applications of optimization models in ATMS are presented.

Fig. 4 Main components of ATMS

5.3 Models of Advanced Travel Information System (ATIS)

In this system, travel information is provided to the passengers before trip or during the trip. This can be shared either inside or outside the vehicle. This system is completely dependent on ATMS decisions. It also provides a service for traffic management by guiding the users to some routes. In fact, all transportation data is being used in a unit called the Information Services Provider (ISP), and it is used by both of ATMS and ATIS. These two systems collaborate to provide the best services. In Table 3, the optimization models in ATIS has been mentioned.

5.4 Models of Vehicle Safety System (VSS)

In this system, a series of technologies are designed to improve vehicle control and safety of motion. Modules of this system cover a range of active and passive methods. The driving evaluation systems in this category provide appropriate services for increasing safety and so the different insurance companies and police offices require

Table 2 The application of optimization models in ATMS

User service	Sample reference	Description and process
Network monitoring	[141]	A comprehensive review of wireless sensor networks for ITS solutions for traffic optimization and real-time traffic light control
Controlling the traffic lights	[41]	Using a combination of network optimization techniques and nonlinear programming to optimize signal settings
Traffic prediction	[142]	Traffic flow prediction by minimizing the sensors in the links to adjust the origin-to-destination matrices
Lane management	[143]	A new bi-level formulation for time-varying lane-based capacity reversibility problem for traffic management by using the genetic algorithm (GA) with the simulation-based systems
Traffic information broadcasting	[144]	Broadcasting methods for sending safety messages and routing information in a wide and high mobility vehicular ad hoc networks by determining the best communication strategies for each node according to its neighborhood density
Traffic incident management	[145]	Dispatching response units by an optimization process for multiple incident response management by integrating Geographic Information System (GIS) with traffic simulation and optimization engines
Electronic toll collection	[146]	Deployment of electronic toll collection (ETC) by using a model to maximize social welfare associated with a toll plazas
Reducing greenhouse gas emissions	[147]	To explore the environmental impact of pooling of supply chains at the strategic level, an optimization model has been developed to reduce emissions for two transport modes, road, and rail

(continued)

Table 2 (continued)

User service	Sample reference	Description and process
Integration of the rails with the roads	[148]	This reference presented the problem of optimally locating rail/road terminals for freight transport by using a linear 0–1 program that was solved by a heuristic approach
Smart parking management	[149]	A recharge scheduling system was presented for parking lots using a realistic vehicular mobility/parking pattern focusing on individual parking lots
Variable speed limit	[150]	To maximize recurrent bottleneck flow, a control strategy for combining Variable Speed Limits (VSL) and Coordinated Ramp Metering (CRM) design, has been proposed
Dynamic road management	[151]	Here, time-dependent tolls have been studied for optimizing the network performance by a bi-level optimization problem
Variable message signs	[66]	Finding an optimal set of locations was pursued to install a given number of variable message signs by a bi-level stochastic integer programming model

Table 3 The application of optimization models in ATIS

User service	Sample reference	Description and process
Dynamic path guidance	[152]	Dynamic route guidance by A^* algorithm and genetic algorithm were presented in this paper
Travel planning	[153]	It proposed a collective travel planning to find the least cost route connecting multiple sources, via at most k meeting points
Dynamic trip-sharing	[154]	It surveyed bike or car sharing algorithms to pick-up a vehicle at any location and return it to any other station

them essentially [7, 9, 36, 38, 155]. The models of optimization for the services of this system are stated in Table 4.

Table 4 The application of optimization models in VSS

User service	Sample reference	Description and process
Safety alert at intersection	[156]	This paper considers the cooperative intersection optimization in which road users, infrastructure, and traffic control centers can communicate and coordinate the traffic safely and efficiently
Advanced vehicle control	[157]	In this paper, plug-in hybrid electric vehicles have been used to optimize the energy flow by generating the most efficient operating conditions for a parallel pre-transmission hybrid and a specific driving cycle. The engine, electric machine, and transmission operating modes were then used to generate a rule-based control strategy
Avoiding collisions at intersections	[158]	This paper proposed an intersection system by using cooperative coordination between vehicles to adapt the speed for finalizing the maneuvers both safely and efficiently. This was solved by a hierarchical fuzzy rule-based system optimized by a genetic algorithm (GA)

5.5 *Models of Emergency Management System (EMS)*

The purpose of this system is to improve the response time to accidents, to increase the likelihood of survival and to reduce injuries. To reduce the response time, the communication time between the agents and emergency management center should be reduced. This system has important implications for reducing casualties and driving peace. Table 5 shows some optimization models in this system.

5.6 *Models of Advanced Public Transport System (APTS)*

APTS includes passenger data to improve the operations of public transportation agents. It supports network design, fleet management, bus and crew scheduling, fare collection, route guidance, transferring between different modes of transit, etc. These services can be provided inside or outside of the vehicle. This system also controls the transit vehicles and can call an advanced service to get a priority in the intersections. Some of the most related services in this system can be categorized as the following:

- Sampling method to the cluster the patterns of users based on smart card data [163]

Table 5 The application of optimization models in EMS

User service	Sample reference	Description and process
Emergency calls and dispatch	[67]	This paper used a mathematical formulation for the emergency medical service system by providing a quick response to emergencies. It optimizes the location of the ambulances and their allocation to the customers. This model was solved by the genetic algorithm
Emergency routing	[159]	This paper involved warehouse selection, fleet routing, and scheduling to meet demand in the strict time window to obtain a scalable solution based on the route capacity and location selection constraints by the aid of a two-level optimization problem.
Road patrol services	[160]	This paper optimized the police patrol services by determining the important locations and routes based on the topology of road networks and cross-entropy approach
Protection of the civil infrastructure	[161]	This paper proposed a mathematical decision problem for quantifying the infrastructure measures congruously and maximizing their values
Supporting the natural disasters and reconstruction	[162]	This paper developed a plan for optimizing the resource allocation to compete for the recovery projects, quantifying the overall functional loss of damaged transportation networks during the recovery efforts, evaluating the impact of limited availability of resources on the reconstruction costs; It also minimized the performance loss of transportation networks and reconstruction costs

- Big data in public transportation [164]
- Time-based transit fares [165]
- Fuzzy dynamic scheduling for public transit [166]
- Bus monitoring system via ZigBee radio network [167]
- Providing service reliability for different public transit services [168]
- Mitigation of disruptions in public transit by bee colony optimization [169].

Table 6 includes some optimization applications in APTS.

Table 6 The application of optimization models in APTS

User service	Sample reference	Description and process
Public transportation on fixed routes	[170]	A reliable bus route schedule has been designed by taking into account the bus travel time uncertainty and the bus drivers' schedule recovery efforts. A Monte Carlo simulation-based solution method was used to solve the presented robust optimization model
Dial-a-ride public transportation systems	[171]	Each customer requirement has been specified in terms of an origin, a destination and a time window. This paper proposed a set of routes, each assigned to a public vehicle, to satisfy the requests by maximizing the total ride time and the total waiting time
Fare collection management	[172]	The data of the fare collection system has been used for transit planning in this paper by estimating the destination location for each individual boarding a bus with a smart card
Fleet management	[173]	A mathematical model has been studied to consider the relevant activities at tactical and operational levels for private companies and public agencies on passengers and freight transportation services. In this paper, the combinatorial optimization problems, such as vehicle routing and scheduling have been applied in the static and dynamic statues
Bus priority at roads	[174]	This paper formulated transit road space priority at the network level and utilized an efficient heuristic method. A bi-level programming approach is adapted for this purpose. The upper level involves an objective function from the system managers' perspective, whereas, at the lower level, a users' perspective is modeled. Then a genetic algorithm is used to solve this problem
Bus priority at intersections	[175]	To implement the transit priority in the signalized intersections, the major factors affecting transit priority have been identified, and the formulation of both passive and active transit priority strategies for arterials with coordinated traffic signals were described

(continued)

Table 6 (continued)

User service	Sample reference	Description and process
Public transportation data mining	[51]	The automated fare collection system has been used to collect data from transactions for extracting passenger's origin and destination. Such transit origin data are highly valuable for transit system planning and route optimization

5.7 Models of Commercial Vehicle Operations (CVO)

This system includes technologies that are most useful for commercial trucks. Fleet management, electronic clearance, weighting-in-motion, international border crossing and freight monitoring are the most important services of this system. Of course, some of these services are shared by public carriers. This system also contains some special services for hazardous material transportation [35].

Since the CVO system plays an essential role in the management of commercial vehicles, it has a great effect on the economic development of a country and increases the efficiency of the transportation network and the national boundaries of the country. Besides, due to reduced downtime, trucks can reduce shipping costs. On the other hand, it also plays an important role in creating and improving the security of the country and preventing theft and smuggling, due to the possibility of individual tracking of loads and trucks. Table 7 presents some optimization models for implementing CVO services:

Table 7 The application of optimization models in CVO

User service	Sample reference	Description and process
Fleet management	[176]	It defined some heuristics to manage a fleet of trucks to provide truckload pickup-and-delivery services under time windows
Cargo management	[177]	For cargo port scheduling, a mathematical model has been established to minimize the total service time and make-span. It solved by a multi-phase particle swarm optimization (MPPSO) algorithm
Hazardous materials management	[35]	This paper proposed a decision support system for routing, scheduling, and assignment for hazardous material transportation

Table 8 The application of optimization models in CMRI

User service	Sample reference	Description and process
Winter road maintenance	[178]	This paper survived on the winter road maintenance operations by considering the decision-making problems at the strategic, tactical, operational, and real-time levels
Working zone management	[179]	This paper proposed an optimization technique to determine the appropriate work-zone plans to minimize the total costs including agency costs, road-user delay costs and accident costs, subject to working time constraints. This model has been solved by a modified simulated annealing algorithm

5.8 Models of Construction, Maintenance and Repair of Infrastructure (CMRI)

This system works to improve manufacturing, maintenance and repair processes, optimize the efficiency of existing equipment, protect manpower and accelerate the process of repairing road bottlenecks. Protecting the work-zone and clearing the paths in the winter and snow conditions is another function of this system. The optimization models that are used in this system are reviewed in Table 8.

5.9 Models of Data Archiving System (DAS)

This system includes the mechanisms to create an integrated data center plan to share the data between intelligent transportation departments and sectors. In fact, it supports all departments to make decisions based on the provided data. As one can see in Fig. 5, the different data from the infrastructure that can be collected by ATMS or CMRI are integrated in the ITS data warehouse. Also, the users' data that can be gathered by ATIS, APTS, and ATMS are augmented to the data warehouse. Finally, the vehicles' data that are provided by ATMS, APTS, EMS, CVO and CMRI, are sent to the data warehouse.

In the data warehouse, after integrating the data formats and using cleaning techniques, the data are stored in some data cubes to provide some necessary analysis to execute queries, searches, processing on the patterns in the streams, visualizing data and making decisions. The resulted data can be organized in some data marts with respect to the different business purposes. To see more details, see e.g., [8]. Some of the references that focus on optimization models in DAS are presented in Table 9.

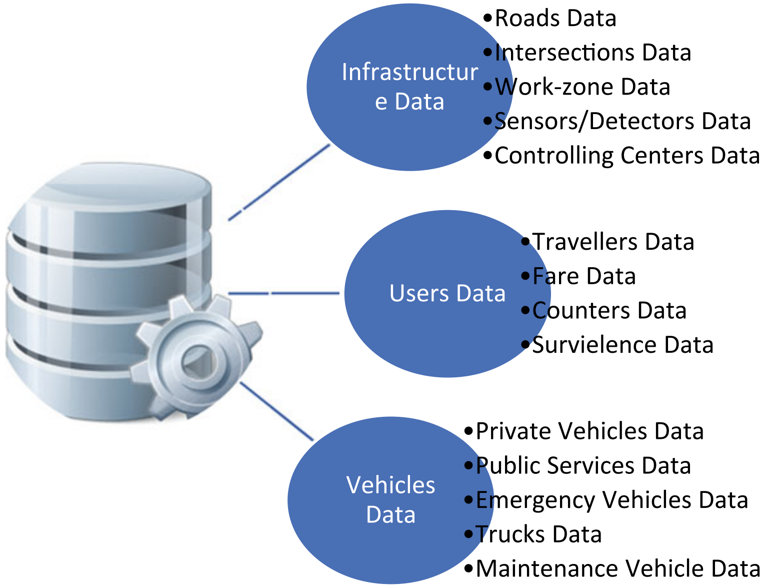


Fig. 5 ITS data warehouse that integrates the different data of transportation systems

Table 9 The application of optimization models in DAS

User service	Sample reference	Description and process
Management based on data warehouse	[50]	This paper presented a data warehouse approach with big data analyses for decision making on intelligent urban mobility and road transport
Big data analysis	[180]	This paper optimized the traffic flow by identifying the vehicles and providing alternate routes. It concluded that the big data analysis for real-time traffic information reduces casualties, minimizes congestion, and increases safety across street networks

6 Conclusion and Future Directions

The purpose of the development of Intelligent Transportation Systems (ITS) is to provide information technology-based solutions and systems analysis to enhance the quality of transportation, to improve safety, to utilize clean and integrated transportation. Of course, these systems contribute to the promotion of users' driving culture, to increase legality, to consider the standards and to control traffic automatically. Therefore, these systems are highly developed in the urban and intercity networks. These systems can be categorized in nine branches:

1. Advanced Human-based Transportation System (AHTS)
2. Advanced Traffic Management System (ATMS)
3. Advanced Travel Information System (ATIS)
4. Vehicle Safety System (VSS)
5. Emergency Management System (EMS)
6. Advanced Public Transport System (APTS)
7. Commercial Vehicle Operations (CVO)
8. Construction, Maintenance and Repair of Infrastructure (CMRI)
9. Data Archiving System (DAS).

This chapter focused on these nine systems and presented the applications of the optimization techniques in these systems. We firstly, considered the data collection and data cleaning in ITS developments. Some data mining approaches such as frequent pattern mining, regression analysis, data clustering and data classification can be used to extract the necessary knowledge from transportation data [10–13]. Then we presented the optimization models. These models can be implemented by graph and network analysis. Thus, we gave some network optimization models that can be used for ITS developments. The different models such as shortest path problem, maximum flow problem, minimum cost flow problem, assignment problem, multi-commodify flow problem, traffic assignment problem, sensor network problem and network control problem have been discussed.

Besides, simulation optimization can be followed for ITS developments. Thus, we presented some details about how one can define meta-models to define a functional forms for simulation data. Among the different methods, we presented the implementation points about fuzzy regression and neural network. Then, we showed how a metaheuristic optimizer can be used on a meta-model to find the near optimal solutions.

As the final part, we presented the advanced optimization solutions for the considered 9 systems of ITS.

The following scientific gaps have been obtained that can be covered by the next researches:

1. Although the human-based transportation systems have been neglected in car-oriented ITS architecture, this is very important to improve a safe life for people. Thus, Much more effort is needed to identify the real needs of the vulnerable people, such as the elderly, children, the blind, the deaf and the disabled persons, and to develop appropriate transportation systems for them by considering their special needs. The role of optimization models in this area can be crucial in order to avoid the cost overruns and to develop the comprehensive transport systems for these people.
2. In transportation management systems (ATMS), imbalanced development presents many risks. Developing intelligent transportation systems with the ability to generate high volumes of data without planning how to process them is useless. In many cities, high volumes of data are collected by detectors, installed at intersections and they are left unused. Mobile operators are always collecting huge amounts of users' data and eventually discarding them. The need to process

big data collected by traffic management systems including surveillance cameras, sensors, detectors, and counters is strongly felt. In this regard, optimization models are very effective for designing low-cost models of data collection, data analysis, and data compression. Project management techniques that meet the prerequisites for different systems are another field of optimization that can be followed by system-developing organizations.

3. In the development of ATIS, the effect of the fusion of data obtained from different information sources is very high. This effect is revealed in trip planning and traveling guidance modules. Today, various data are collected by traffic management systems. Mobile data and telecommunications antenna also collect different kinds of users' data. Social networks and Internet-of-Things (IoT) are also very useful to collect transportation data. Providing new and effective fusion methods that can integrate data collected with different structures is a very challenging subject. The structures are graphs, time-series, streams of data and so on. Thus the structured data and non-structured data can be combined in a unified platform. Providing suitable multi-criteria optimization models that can provide an effective way of integrating data concerning the importance and accuracy of each data segment is a research expectation. Knowledge extraction methods from this data also require optimization of machine learning models. Besides, how to optimally present this data to the user as well as how to disseminate the extracted knowledge is a major challenge. Obviously, if the same data is given to all users, congestion in the traffic network will increase. Optimization of the transfer of this knowledge to users according to probabilistic models can be considered in future optimization researches.
4. In VSS and EMS implementations, there are many issues that require robust optimization. According to the results of the current research, these models have been extended for these parts very scarcely. While robust optimization models can play a very important role in achieving appropriate safety responses. Also, due to the need to simulate very complex scenarios in order to develop programs and standards for safety and incident management, simulation-optimization methods are highly recommended to solve the problems in these categories. The possibilities of tracking cars with the help of IoT and smartphones are not to be overlooked. Providing online optimization models to solve the problems of VSS and EMS is also a major challenge.
5. In the APTS, CVO, CMRI, we need gradually to consider the relationship between vehicles (V2V), between vehicles and infrastructure (V2I) and between vehicles and people (V2P). Optimization models can help to customize these technologies to suit the needs of these systems. Since this technology is so expensive, creating a model for optimal use of these systems, optimizing their equipment location and installation, inclusive models for collecting and processing their data are crucial. Also, unlike autonomous vehicles, the costs of this sector are mostly dependent on the infrastructure and must be funded by governments. Thus, it is a need to provide an optimal economic model for the development of this technology for each country.

6. The last but not the least, at DAS, we have enormous challenges in integrating data and identifying the data cube needed to store data and to analyze them according to the different needs. In the field of transportation, there are many sectors, including police departments, municipalities, road agencies, telecommunications operators, car manufacturing companies, travel and tourism companies, rescue organizations, national security agencies, borders, ports, terminals, airports and train stations. Also, people are the most important sector in transportation. All of these sectors produce transportation data, affect transportation data collection or use transportation data. The maturity level of the data in this section is not the same. People's privacy must also be protected. So it's very difficult to build and update a data warehouse in this area. Optimization models can help to make an appropriate decision about how to manage and to store data by implementing different stakeholders' views within the model constraints and model goals. These models should be able to prevent the privacy of individuals while meeting the real needs of the sectors involved in this segment. This challenge will remain one of the major data science challenges in the field of transportation.

Acknowledgments It is my pleasure to thank my dear spouse and all my hard-working students for their support.

References

1. Intelligent transportation systems—national ITS architecture [Online]. Available: https://www.its.dot.gov/research_archives/arch/index.htm. Accessed: 01 Oct 2019
2. National Iranian ITS plan—ITSRI-Amirkabir University of Technology [Online]. Available: http://its.aut.ac.ir/nits/. Accessed: 10 Oct 2019
3. Abpeykar S, Ghatee M (2014) Supervised and unsupervised learning DSS for incident management in intelligent tunnel: a case study in Tehran Niayesh tunnel. Tunn Undergr Space Technol 42:293–306
4. Dabove P (2019) Smartphones: recent innovations and applications. Nova Science Publishers, Incorporated
5. Bessis N, Dobre C (2014) Big data and internet of things: a roadmap for smart environments, vol 546. Springer
6. Eftekhari HR, Ghatee M (2016) An inference engine for smartphones to preprocess data and detect stationary and transportation modes. Transp Res Part C Emerg Technol 69:313–327
7. Eftekhari HR, Ghatee M (2018) Hybrid of discrete wavelet transform and adaptive neuro fuzzy inference system for overall driving behavior recognition. Transp Res Part F Traffic Psychol Behav 58:782–796
8. Han J, Pei J, Kamber M (2011) Data mining: concepts and techniques. Elsevier
9. Bejani MM, Ghatee M (2018) A context aware system for driving style evaluation by an ensemble learning on smartphone sensors data. Transp Res Part C Emerg Technol 89:303–320
10. Vlahogianni EI (2015) Computational intelligence and optimization for transportation big data: challenges and opportunities. In: Engineering and applied sciences optimization. Springer, pp 107–128
11. Mehmood R, Meriton R, Graham G, Hennelly P, Kumar M (2017) Exploring the influence of big data on city transport operations: a Markovian approach. Int J Oper Prod Manag 37(1):75–104

12. Abpeykar S, Ghatee M, Zare H (2019) Ensemble decision forest of RBF networks via hybrid feature clustering approach for high-dimensional data classification. Comput Stat Data Anal 131:12–36
13. Abpeykar S, Ghatee M (2019) Neural trees with peer-to-peer and server-to-client knowledge transferring models for high-dimensional data classification. Expert Syst Appl 137:281–291
14. Fakhri A, Ghatee M (2016) Application of benders decomposition method in solution of a fixed-charge multicommodity network design problem avoiding congestion. Appl Math Model 40(13–14):6468–6476
15. Fakhri A, Ghatee M (2014) Fractional multi-commodity flow problem: duality and optimality conditions. Appl Math Model 38(7–8):2151–2162
16. Fakhri A, Ghatee M (2016) Minimizing the sum of a linear and a linear fractional function applying conic quadratic representation: continuous and discrete problems. Optimization 65(5):1023–1038
17. Fakhri A, Ghatee M (2013) Solution of preemptive multi-objective network design problems applying benders decomposition method. Ann Oper Res 210(1):295–307
18. Niksirat M, Ghatee M, Hashemi SM (2012) Multimodal *K*-shortest viable path problem in Tehran public transportation network and its solution applying ant colony and simulated annealing algorithms. Appl Math Model 36(11):5709–5726
19. Ghatee M (2011) QoS-based cooperative algorithm for integral multi-commodity flow problem. Comput Commun 34(7):835–846
20. Davis L (1991) Handbook of genetic algorithms. CumInCAD
21. Glover F (1989) Tabu search—part I. ORSA J Comput 1(3):190–206
22. Glover F (1990) Tabu search—part II. ORSA J Comput 2(1):4–32
23. Moscato P (1999) Memetic algorithms: a short introduction. In: New ideas in optimization. McGraw-Hill Ltd., UK, pp 219–234
24. Van Laarhoven PJ, Aarts EH (1987) Simulated annealing. In: Simulated annealing: theory and applications. Springer, pp 7–15
25. Dorigo M, Birattari M (2010) Ant colony optimization. Springer
26. Kennedy J (2010) Particle swarm optimization. In: Sammut C, Webb GI (eds) Encyclopedia of machine learning, Springer, Boston, MA, pp 760–766. https://doi.org/10.1007/978-0-387-30164-8_630
27. Yang X-S (2010) Nature-inspired metaheuristic algorithms. Luniver Press
28. Hopfield JJ, Tank DW (1985) 'Neural' computation of decisions in optimization problems. Biol Cybern 52(3):141–152
29. Karaboga D, Basturk B (2007) A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. J Glob Optim 39(3):459–471
30. Geem ZW, Kim JH, Loganathan GV (2001) A new heuristic optimization algorithm: harmony search. Simulation 76(2):60–68
31. Atashpaz-Gargari E, Lucas C (2007) Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. In: 2007 IEEE congress on evolutionary computation, IEEE. pp 4661–4667
32. Ahuja RK, Magnanti TL, Orlin JB, Weihe K (1995) Network flows: theory, algorithms and applications. ZOR-Methods Models Oper Res 41(3):252–254
33. Tu C-J, Chuang L-Y, Chang J-Y, Yang C-H (2007) Feature selection using PSO-SVM. Int J Comput Sci 33(1):1–18
34. Abpeykar S, Ghatee M (2018) Decent direction methods on the feasible region recognized by supervised learning metamodels to solve unstructured problems. J Inf Optim Sci 39(6):1245–1262
35. Asadi R, Ghatee M (2015) A rule-based decision support system in intelligent hazmat transportation system. IEEE Trans Intell Transp Syst 16(5):2756–2764
36. Eftekhari HR, Ghatee M (2019) A similarity-based neuro-fuzzy modeling for driving behavior recognition applying fusion of smartphone sensors. J Intell Transp Syst 23(1):72–83
37. Pashaei A, Ghatee M, Sajedi H (2019) Convolution neural network joint with mixture of extreme learning machines for feature extraction and classification of accident images. J Real-Time Image Process 17:1–16

38. Ghatee M (2019) Smartphone-based systems for driving evaluation. In: Dabove P (ed) Smartphones: recent innovations and applications. Nova Science Publishers, Incorporated, 2019, pp 143–222
39. Pressl B, Wieser M (2006) A computer-based navigation system tailored to the needs of blind people. In: International conference on computers for handicapped persons. Springer, Berlin, Heidelberg, pp 1280–1286
40. Ghatee M, Hashemi M (2009) An expert system for network control problems and its applications in large scale network design under uncertainty. In: International network optimization conference (INOC 2009), Pisa, Italy
41. Ghatee M, Hashemi SM (2007) Descent direction algorithm with multicommodity flow problem for signal optimization and traffic assignment jointly. Appl Math Comput 188(1):555–566
42. Ghatee M, Hashemi SM, Zarepisheh M, Khorram E (2009) Preemptive priority-based algorithms for fuzzy minimal cost flow problem: an application in hazardous materials transportation. Comput Ind Eng 57(1):341–354
43. Ghatee M, Hashemi SM (2007) Ranking function-based solutions of fully fuzzified minimal cost flow problem. Inf Sci 177(20):4271–4294
44. Ghatee M, Hashemi SM (2009) Application of fuzzy minimum cost flow problems to network design under uncertainty. Fuzzy Sets Syst 160(22):3263–3289
45. Eftekhari HR, Ghatee M (2017) The lower bound for dynamic parking prices to decrease congestion through CBD. Oper Res 17(3):761–787
46. Soudmand S, Ghatee M, Hashemi SM (2013) SA-IP method for congestion pricing based on level of service in urban network under fuzzy conditions. Int J Civ Eng 11(4A):281–291
47. Ceylan H, Bell MG (2005) Genetic algorithm solution for the stochastic equilibrium transportation networks under congestion. Transp Res Part B Methodol 39(2):169–185
48. Kubrusly CS, Malebranche H (1985) Sensors and controllers location in distributed systems—a survey. Automatica 21(2):117–128
49. Iqbal M, Naeem M, Anpalagan A, Ahmed A, Azam M (2015) Wireless sensor network optimization: multi-objective paradigm. Sensors 15(7):17572–17620
50. Addakiri K, Khallouki H, Bahaj M (2019) Intelligent urban transport decision analysis system based on mining in big data analytics and data warehouse. In: International conference on artificial intelligence and symbolic computation. Springer, Cham, pp 179–184
51. Ma X, Wang Y, Chen F, Liu J (2012) Transit smart card data mining for passenger origin information extraction. J Zhejiang Univ Sci C 13(10):750–760
52. Sheffi Y (1985) Urban transportation networks, vol 6. Prentice-Hall, Englewood Cliffs, NJ
53. Bertsekas DP (1998) Network optimization: continuous and discrete models. Athena Scientific Belmont
54. Sładkowski A, Pamuła W (2016) Intelligent transportation systems-problems and perspectives, vol 303. Springer
55. Bianco L, Toth P (2012) Advanced methods in transportation analysis. Springer Science & Business Media
56. Ghatee M, Hashemi SM (2009) Traffic assignment model with fuzzy level of travel demand: An efficient algorithm based on quasi-Logit formulas. Eur J Oper Res 194(2):432–451
57. Ghatee M, Hashemi SM (2009) Some concepts of the fuzzy multicommodity flow problem and their application in fuzzy network design. Math Comput Model 49(5–6):1030–1043
58. Bui K-HN, Jung JJ (2019) ACO-based dynamic decision making for connected vehicles in IoT system. IEEE Trans Ind Inform 15:5648–5655
59. Tan Y (2019) A new shortest path algorithm generalized on dynamic graph for commercial intelligent navigation for transportation management. ArXiv Prepr. ArXiv190504469
60. Zeng J, Yao QG, Zhang YS, Lu JT, Wang M (2019) Optimal path selection for emergency relief supplies after mine disasters. Int J Simul Model IJSIMM 18(3):476–487
61. Alawadhi N, Shaikhli IA, Akandari A, Tahir M (2019) Optimal path planning for urban vehicles using internet of things: a new navigation perspective. J Comput Theor Nanosci 16(3):1074–1080

62. Hamza-Lup GL, Hua KA, Peng R, Ho AH (2005) A maximum-flow approach to dynamic handling of multiple incidents in traffic evacuation management. In: Proceedings. 2005 IEEE intelligent transportation systems, 2005, IEEE, pp 1147–1152
63. Ye P, Chen C, Zhu F (2011) Dynamic route guidance using maximum flow theory and its mapreduce implementation. In: 2011 14th International IEEE conference on intelligent transportation systems (ITSC), IEEE, pp 180–185
64. Tan Z (2019) How many passengers can we serve with ride-sharing? ArXiv Prepr. ArXiv 190107906
65. Ghatee M, Hashemi SM (2008) Generalized minimal cost flow problem in fuzzy nature: an application in bus network planning problem. Appl Math Model 32(12):2490–2508
66. Chiu Y-C, Huynh N, Mahmassani HS (2001) Determining optimal locations for variable message signs under stochastic incident scenarios. In: 80th Annual meeting transportation research board, Washington DC, United States, Paper 01-2927
67. Toro-DíAz H, Mayorga ME, Chanta S, Mclay LA (2013) Joint location and dispatching decisions for emergency medical services. Comput Ind Eng 64(4):917–928
68. Ming L, Liang B, Zheng F, Chu F (2019) Stochastic airline fleet assignment with risk aversion
69. Boccia M, Crainic TG, Sforza A, Sterle C (2018) Multi-commodity location-routing: flow intercepting formulation and branch-and-cut algorithm. Comput Oper Res 89:94–112
70. Lee S, Lee J, Na B (2018) Practical routing algorithm using a congestion monitoring system in semiconductor manufacturing. IEEE Trans Semicond Manuf 31(4):475–485
71. Li P, Mirchandani P, Zhou X (2015) Simulation-based traffic signal optimization to minimize fuel consumption and emission: a Lagrangian relaxation approach. In: 94th Annual meeting transportation research board, Washington DC, United States, Paper: 15-2358
72. Kachroo P, Özbay KM (2018) Traffic assignment: a survey of mathematical models and techniques. In: Feedback control theory for dynamic traffic assignment. Springer, pp 25–53
73. Patriksson M, Labbé M (2002) Transportation planning: state of the art, vol 64. Springer Science & Business Media
74. Zhang Y, Qin X, Dong S, Ran B (2010) Daily OD matrix estimation using cellular probe data. In: 89th Annual meeting transportation research board, vol 9. Washington DC, United States, Paper: 10-2472
75. Boyce D (2007) Forecasting travel on congested urban transportation networks: review and prospects for network equilibrium models. Netw Spat Econ 7(2):99–128
76. Kim H, Oh J-S, Jayakrishnan R (2009) Effects of user equilibrium assumptions on network traffic pattern. KSCE J Civ Eng 13(2):117–127
77. Connors RD, Sumalee A (2009) A network equilibrium model with travellers' perception of stochastic travel times. Transp Res Part B Methodol 43(6):614–624
78. Chen Y-W, Tzeng G-H (2001) Using fuzzy integral for evaluating subjectively perceived travel costs in a traffic assignment model. Eur J Oper Res 130(3):653–664
79. Peeta S, Mahmassani HS (1995) System optimal and user equilibrium time-dependent traffic assignment in congested networks. Ann Oper Res 60(1):81–113
80. Bell MG, Cassir C (2002) Risk-averse user equilibrium traffic assignment: an application of game theory. Transp Res Part B Methodol 36(8):671–681
81. Lu C-C, Mahmassani HS, Zhou X (2008) A bi-criterion dynamic user equilibrium traffic assignment model and solution algorithm for evaluating dynamic road pricing strategies. Transp Res Part C Emerg Technol 16(4):371–389
82. Xie J, Nie Y (2019) A new algorithm for achieving proportionality in user equilibrium traffic assignment. Transp Sci 53(2):566–584
83. Li M, Di X, Liu HX, Huang H-J (2019) A restricted path-based ridesharing user equilibrium. J Intell Transp Syst 24:1–21
84. Yu H, Ma R, Zhang HM (2018) Optimal traffic signal control under dynamic user equilibrium and link constraints in a general network. Transp Res Part B Methodol 110:302–325
85. Hazelton ML (1998) Some remarks on stochastic user equilibrium. Transp Res Part B Methodol 32(2):101–108

86. Lam WH-K, Gao ZY, Chan KS, Yang H (1999) A stochastic user equilibrium assignment model for congested transit networks. Transp Res Part B Methodol 33(5):351–368
87. Fisk C (1980) Some developments in equilibrium traffic assignment. Transp Res Part B Methodol 14(3):243–255
88. Maher M (1998) Algorithms for logit-based stochastic user equilibrium assignment. Transp Res Part B Methodol 32(8):539–549
89. Huang H-J, Li Z-C (2007) A multiclass, multicriteria logit-based traffic equilibrium assignment model under ATIS. Eur J Oper Res 176(3):1464–1477
90. Liu Z, Wang S, Meng Q (2014) Toll pricing framework under logit-based stochastic user equilibrium constraints. J Adv Transp 48(8):1121–1137
91. Wen C-H, Koppelman FS (2001) The generalized nested logit model. Transp Res Part B Methodol 35(7):627–641
92. Koppelman FS, Wen C-H (2000) The paired combinatorial logit model: properties, estimation and application. Transp Res Part B Methodol 34(2):75–89
93. Yu Q, Fang D, Du W (2014) Solving the logit-based stochastic user equilibrium problem with elastic demand based on the extended traffic network model. Eur J Oper Res 239(1):112–118
94. Zhang K, Mahmassani HS, Lu C-C (2008) Probit-based time-dependent stochastic user equilibrium traffic assignment model. Transp Res Rec 2085(1):86–94
95. Kitthamkesorn S, Chen A (2013) A path-size weibit stochastic user equilibrium model. Proc Soc Behav Sci 80:608–632
96. Xuyan Q, Huapu L, Yuanyuan W (2005) A K-shortest-paths-based algorithm for stochastic traffic assignment model and comparison of computation precision with existing methods. Proc Eastern Asia Soc Transp Stud 5:1218–1232
97. Van der Zijpp NJ, Catalano SF (2005) Path enumeration by finding the constrained K-shortest paths. Transp Res Part B Methodol 39(6):545–563
98. Raith A, Wang JY, Ehrgott M, Mitchell SA (2014) Solving multi-objective traffic assignment. Ann Oper Res 222(1):483–516
99. Chabini I (1998) Discrete dynamic shortest path problems in transportation applications: complexity and algorithms with optimal run time. Transp Res Rec 1645(1):170–175
100. Chen BY, Lam WH, Sumalee A, Li Q, Tam ML (2014) Reliable shortest path problems in stochastic time-dependent networks. J Intell Transp Syst 18(2):177–189
101. Liu HX, Ban X, Ran B, Mirchandani P (2003) Formulation and solution algorithm for fuzzy dynamic traffic assignment model. Transp Res Rec 1854(1):114–123
102. Pi X, Qian ZS (2017) A stochastic optimal control approach for real-time traffic routing considering demand uncertainties and travelers' choice heterogeneity. Transp Res Part B Methodol 104:710–732
103. Chen W, Chen L, Chen Z, Tu S (2006) Wits: a wireless sensor network for intelligent transportation system. In: First international multi-symposiums on computer and computational sciences (IMSCCS'06), vol 2, pp 635–641
104. Chuan-zhi L, Ru-fu H, Ye H (2008) Method of freeway incident detection using wireless positioning. In: 2008 IEEE international conference on automation and logistics, Qingdao, China, IEEE, pp 2801–2804
105. Fadaei M, Ghatee M, Hashemi SM, (2012) Two new models for the traffic counters location problem to maximize the coverage of paths: tehran highways case-study. J Traffic Eng 13(48):19–23
106. Maher MJ, Zhang X, Van Vliet D (2001) A bi-level programming approach for trip matrix estimation and traffic control problems with stochastic user equilibrium link flows. Transp Res Part B Methodol 35(1):23–40
107. Gartner NH, Al-Malik M (1996) Combined model for signal control and route choice in urban traffic networks. Transp Res Rec 1554(1):27–35
108. Chiou S-W (2005) Bilevel programming for the continuous transport network design problem. Transp Res Part B Methodol 39(4):361–383
109. Tornquist J, Persson JA (2005) Train traffic deviation handling using tabu search and simulated annealing. In: Proceedings of the 38th annual Hawaii international conference on system sciences, Big Island, HI, USA, IEEE, pp 1–10. https://doi.org/10.1109/HICSS.2005.641

110. Ceylan H, Bell MG (2004) Traffic signal timing optimisation based on genetic algorithm approach, including drivers' routing. Transp Res Part B Methodol 38(4):329–342
111. Banks J, Carson II, Nelson BL, Nicol DM (2005) Discrete-event system simulation. Pearson
112. Badiru AB, Sieger DB (1998) Neural network as a simulation metamodel in economic analysis of risky projects. Eur J Oper Res 105(1):130–142
113. Chambers M, Mount-Campbell CA (2002) Process optimization via neural network metamodeling. Int J Prod Econ 79(2):93–100
114. Blanning RW (1975) The construction and implementation of metamodels. Simulation 24(6):177–184
115. Nasrabadi E, Hashemi SM, Ghatee M (2007) An LP-based approach to outliers detection in fuzzy regression analysis. Int J Uncertain Fuzziness Knowl Based Syst 15(04):441–456
116. Dehghan M, Ghatee M, Hashemi B (2008) Some computations on fuzzy matrices: an application in fuzzy analytical hierarchy process. Int J Uncertain Fuzziness Knowl-Based Syst 16(05):715–733
117. Hashemi SM, Ghatee M, Hashemi B (2006) Fuzzy goal programming: complementary slackness conditions and computational schemes. Appl Math Comput 179(2):506–522
118. Rommelfanger H (1996) Fuzzy linear programming and applications. Eur J Oper Res 92(3):512–527
119. Hrušovskỳ M, Demir E, Jammernegg W, Van Woensel T (2018) Hybrid simulation and optimization approach for green intermodal transportation problem with travel time uncertainty. Flex Serv Manuf J 30(3):486–516
120. Sayyadi R, Awasthi A (2018) A simulation-based optimisation approach for identifying key determinants for sustainable transportation planning. Int J Syst Sci Oper Logist 5(2):161–174
121. Rosano M, Demartini CG, Lamberti F, Perboli G (2018) A mobile platform for collaborative urban freight transportation. Transp Res Proc 30:14–22
122. Zhao C, Li S, Wang W, Li X, Du Y (2018) Advanced parking space management strategy design: an agent-based simulation optimization approach. Transp Res Rec 2672(8):901–910
123. Fikar C, Hirsch P, Nolz PC (2018) Agent-based simulation optimization for dynamic disaster relief distribution. Cent Eur J Oper Res 26(2):423–442
124. Yu B, Guo Z, Peng Z, Wang H, Ma X, Wang Y (2018) Agent-based simulation optimization model for road surface maintenance scheme. J Transp Eng Part B Pavement 145(1):04018065
125. Shakibayifar M, Sheikholeslami A, Corman F (2018) A simulation-based optimization approach to reschedule train traffic in uncertain conditions during disruptions. Sci Iran 25(2):646–662
126. Zhao Y, Ioannou PA, Dessouky MM (2018) Dynamic multimodal freight routing using a co-simulation optimization approach. IEEE Trans Intell Transp Syst 20:2657–2667
127. Layeb SB, Jaoua A, Jbira A, Makhlouf Y (2018) A simulation-optimization approach for scheduling in stochastic freight transportation. Comput Ind Eng 126:99–110
128. Niessner H, Rauner MS, Gutjahr WJ (2018) A dynamic simulation—optimization approach for managing mass casualty incidents. Oper Res Health Care 17:82–100
129. Yao B, Yan Q, Chen Q, Tian Z, Zhu X (2018) Simulation-based optimization for urban transportation demand management strategy. Simulation 94(7):637–647
130. Zadeh RB, Ghatee M, Eftekhari HR (2017) Three-phases smartphone-based warning system to protect vulnerable road users under fuzzy conditions. IEEE Trans Intell Transp Syst 19(7):2086–2098
131. Merdrignac P, Shagdar O, Nashashibi F (2016) Fusion of perception and v2p communication systems for the safety of vulnerable road users. IEEE Trans Intell Transp Syst 18(7):1740–1751
132. Regan MA, Oxley JA, Godley ST, Tingvall C (2001) Intelligent transport systems: safety and human factors issues, Royal Automobile Club of Victoria (RACV) Ltd – Report 01/01, Monash University, Australia
133. Anaya JJ, Talavera E, Giménez D, Gómez N, Felipe J, Naranjo JE (2015) Vulnerable road users detection using v2x communications. In: 2015 IEEE 18th international conference on intelligent transportation systems, pp 107–112

134. Guo L, Ge P-S, Zhang M-H, Li L-H, Zhao Y-B (2012) Pedestrian detection for intelligent transportation systems combining AdaBoost algorithm and support vector machine. Expert Syst Appl 39(4):4274–4286
135. Themann P, Kotte J, Raudszus D, Eckstein L (2015) Impact of positioning uncertainty of vulnerable road users on risk minimization in collision avoidance systems. In: 2015 IEEE Intelligent Vehicles Symposium (IV), pp 1201–1206
136. Scholliers J, Bell D, Morris A, García AB (2014) Improving safety and mobility of vulnerable road users through ITS applications. Transp Res Arena 4:14–17
137. Tian W, Lauer M (2017) Tracking vulnerable road users with severe occlusion by adaptive part filter modeling. In: 2017 IEEE international conference on vehicular electronics and safety (ICVES), Vienna, Austria, IEEE, pp 139–144
138. Konforty M, Erez Y, Dor OBB (2010) Alerting a driver to the presence of a pedestrian on a road. U.S. Patent No. 7,777,646. Washington DC, U.S. Patent and Trademark Office
139. Muzammel M, Yusoff MZ, Meriaudeau F (2017) Rear-end vision-based collision detection system for motorcyclists. J Electron Imaging 26(3):033002
140. Hub A, Diepstraten J, Ertl T (2005) Augmented indoor modeling for navigation support for the blind. In: International conference on computers for people with special needs (CPSN), Las Vegas, Nevada, USA, CSREA Press 2005, ISBN 1-932415-78-5, pp 54–62
141. Kafi MA, Challal Y, Djenouri D, Doudou M, Bouabdallah A, Badache N (2013) A study of wireless sensor networks for urban traffic monitoring: applications and architectures. Proc Comput Sci 19:617–626
142. Abadi A, Rajabioun T, Ioannou PA (2014) Traffic flow prediction for road transportation networks with limited traffic data. IEEE Trans Intell Transp Syst 16(2):653–662
143. Karoonsoontawong A, Lin D-Y (2011) Time-varying lane-based capacity reversibility for traffic management. Comput Aided Civ Infrastruct Eng 26(8):632–646
144. Abdou W, Henriet A, Bloch C, Dhoutaut D, Charlet D, Spies F (2011) Using an evolutionary algorithm to optimize the broadcasting methods in mobile ad hoc networks. J Netw Comput Appl 34(6):1794–1804
145. Huang B, Pan X (2007) GIS coupled with traffic simulation and optimization for incident response. Comput Environ Urban Syst 31(2):116–132
146. Levinson D, Chang E (2003) A model for optimizing electronic toll collection systems. Transp Res Part Policy Pract 37(4):293–314
147. Pan S, Ballot E, Fontane F (2013) The reduction of greenhouse gas emissions from freight transport by pooling supply chains. Int J Prod Econ 143(1):86–94
148. Arnold P, Peeters D, Thomas I (2004) Modelling a rail/road intermodal transportation system. Transp Res Part E Logist Transp Rev 40(3):255–270
149. Kuran MŞ, Viana AC, Iannone L, Kofman D, Mermoud G, Vasseur JP (2015) A smart parking lot management system for scheduling the recharging of electric vehicles. IEEE Trans Smart Grid 6(6):2942–2953
150. Lu X-Y, Varaiya P, Horowitz R, Su D, Shladover SE (2011) Novel freeway traffic control with variable speed limit and coordinated ramp metering. Transp Res Rec 2229(1):55–65
151. Joksimovic D, Bliemer M, Bovy P (2005) Dynamic road pricing optimization with heterogeneous users. In: 45th congress of the European regional science association: land use and water management in a sustainable network society, Amsterdam, The Netherlands, European Regional Science Association(ERSA), Louvain-la-Neuve, pp 1–18
152. Liang Z, Jianmin XU, Lingxiang Z (2007) Application of genetic algorithm in dynamic route guidance system. J Transp Syst Eng Inf Technol 7(3):45–48
153. Shang S, Chen L, Wei Z, Jensen CS, Wen J-R, Kalnis P (2015) Collective travel planning in spatial networks. IEEE Trans Knowl Data Eng 28(5):1132–1146
154. Gavalas D, Konstantopoulos C, Pantziou G (2016) Design and management of vehicle-sharing systems: a survey of algorithmic approaches. In: Smart cities and homes. Elsevier, pp 261–289
155. Bejani MM, Ghatee M (2019) Convolutional neural network with adaptive regularization to classify driving styles on smartphones. IEEE Trans Intell Transp Syst 21:543–552

156. Chen L, Englund C (2015) Cooperative intersection management: a survey. IEEE Trans Intell Transp Syst 17(2):570–586
157. Karbowski D, Rousseau A, Pagerit S, Sharer P, Plug-in vehicle control strategy: from global optimization to real time application. In: 22th international electric vehicle symposium (EVS22). Yokohama
158. Onieva E, Milanés V, Villagra J, Pérez J, Godoy J (2012) Genetic optimization of a vehicle fuzzy decision system for intersections. Expert Syst Appl 39(18):13148–13157
159. Han Y, Guan X, Shi L (2011) Optimization based method for supply location selection and routing in large-scale emergency material delivery. IEEE Trans Autom Sci Eng 8(4):683–693
160. Li L, Jiang Z, Duan N, Dong W, Hu K, Sun W (2011) Police patrol service optimization based on the spatial pattern of hotspots. In: Proceedings of 2011 IEEE international conference on service operations, logistics and informatics, Beijing, China, IEEE, pp 45–50
161. Faturechi R, Miller-Hooks E (2014) A mathematical framework for quantifying and optimizing protective actions for civil infrastructure systems. Comput Aided Civ Infrastruct Eng 29(8):572–589
162. Orabi W, El-Rayes K, Senouci AB, Al-Derham H (2009) Optimizing postdisaster reconstruction planning for damaged transportation networks. J Constr Eng Manag 135(10):1039–1048
163. He L, Trépanier M, Agard B (2019) Sampling method applied to the clustering of temporal patterns of public transit smart card users
164. Welch TF, Widita A (2019) Big data in public transportation: a review of sources and methods. Transp Rev 39:1–24
165. Kamel I, Hasnine S, Shalaby A, Habib KN, Abdulhai, B (2019) Evaluating time-based transit fares: integrated model of departure time and travel mode choice within a dynamic multimodal assignment framework. In: 98th Annual meeting transportation research board, Washington DC, United States, Paper: 19-01386
166. Zhang Y, Hu Q, Meng Z, Ralescu A (2019) Fuzzy dynamic timetable scheduling for public transit. Fuzzy Sets Syst 395:235–253
167. Salman A, El-Tawab S, Yorio Z (2019) Expanding coverage of an intelligent transit bus monitoring system via ZigBee radio network. In: Arai K, Bhatia R (eds) Advances in information and communication, FICC 2019, Lecture Notes in Networks and Systems, vol 69. Springer, Cham. https://doi.org/10.1007/978-3-030-12388-8_15
168. Kathuria A, Parida M, Sekhar CR (2019) A review of service reliability measures for public transportation systems. Int J Intell Transp Syst Res 18:1–13
169. Nikolić M, Teodorović D (2019) Mitigation of disruptions in public transit by Bee Colony optimization. Transp Plan Technol 42:1–14
170. Yan Y, Meng Q, Wang S, Guo X (2012) Robust optimization model of schedule design for a fixed bus route. Transp Res Part C Emerg Technol 25:113–121
171. Guerriero F, Pezzella F, Pisacane O, Trollini L (2014) Multi-objective optimization in dial-a-ride public transportation. Transp Res Proc 3:299–308
172. Trépanier M, Tranchant N, Chapleau R (2007) Individual trip destination estimation in a transit smart card automated fare collection system. J Intell Transp Syst 11(1):1–14
173. Bielli M, Bielli A, Rossi R (2011) Trends in models and algorithms for fleet management. Proc Soc Behav Sci 20:4–18
174. Mesbah M, Sarvi M, Currie G (2011) Optimization of transit priority in the transportation network using a genetic algorithm. IEEE Trans Intell Transp Syst 12(3):908–919
175. Skabardonis A (2000) Control strategies for transit priority. Transp Res Rec 1727(1):20–26
176. Mahmassani HS, Kim Y, Jaillet P (2000) Local optimization approaches to solve dynamic commercial fleet management problems. Transp Res Rec 1733(1):71–79
177. Tang M, Gong D, Liu S, Zhang H (2016) Applying multi-phase particle swarm optimization to solve bulk cargo port scheduling problem. Adv Prod Eng Manag, 11(4):299–310
178. Perrier N, Langevin A, Campbell JF (2006) A survey of models and algorithms for winter road maintenance. Part I: system design for spreading and plowing. Comput Oper Res 33(1):209–238

179. Yang N, Schonfeld P, Kang MW (2009) A hybrid methodology for freeway work-zone optimization with time constraints. Public Works Manag Policy 13(3):253–264
180. Deb T, Vishwas N, Saha A (2020) A comparative study on different approaches of road traffic optimization based on big data analytics. In: Performance management of integrated systems and its applications in software engineering. Springer, pp 119–126

Low Power Hilbert Transformer Design Using Multi-objective Seeker Optimization Algorithm

Atul Kumar Dwivedi

Abstract In this paper, a novel design approach using multi-objective evolutionary seeker optimization algorithm has been proposed, in which the Hilbert transformer is designed using half band FIR filter. The proposed technique has been analyzed for Hilbert transformer model in terms of reducing the power consumption, passband error and order simultaneously. The inclusion of power minimization makes the designed Hilbert transformer portable, low power devices thus increasing battery life and less heating effect. The pertinence of the proposed technique was analyzed by comparing the results achieved using the proposed algorithm with other state of the art evolutionary multi-objective algorithms. Using Virtex-7 FPGA and Xilinx X-power analyzer, Power consumption was analyzed. In the present work, a novel EA i.e. hybrid artificial bee colony algorithm has been proposed and further applied for FIR filter design. The filter design task aims at satisfying the dual objectives of meeting the desired frequency domain specifications and power minimization.

Keywords Digital FIR filters · Hilbert transformers · Evolutionary algorithms · Multi-objective seeker optimization · Low power design

1 Introduction

Hilbert transformers (HT) are class of digital filters which has the characteristic to shift the input signal by $\pi/2$ radians phase. The main importance of the HTs is to represent real function into analytic function. HT of a signal x(n) is obtained using summation as given below

$$\mathcal{H}[x(n)] = \lim_{\varepsilon \to 0} \frac{1}{\pi} \sum_{|s-n|>\varepsilon} \frac{x(s)}{n-s} ds \tag{1}$$

A. K. Dwivedi (✉)
Department of Electronics and Communication Engineering, BIET, Jhansi 492010, India
e-mail: atuldwivedi@live.in

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_5

The certain operation of signal processing is simplified by complex signals and is generated by using HTs. It shifts the negative frequencies phase by +90° and positive frequencies phase by −90°. Therefore, the entire frequency band of HTs has unity gain but has a phase shift of −180° at zero frequency. Practically, HTs are used in various applications of digital signal processing like radio engineering, communication systems [1, 2], seismic signals [3], propagation time estimation, vibration analysis [4], medical signal analysis [5] and image processing [6]. HTs can be designed either by using Finite impulse response (FIR) half band filters or infinite impulse response (IIR) half band filters [7]. For synthesizing Hilbert transformer, FIR filters are preferred because FIR filters have exact linear phase, have high stability as compared to infinite impulse response (IIR) filters and are less sensitive to the coefficients quantization [8]. HTs can be easily derived from FIR half band filters [9].

In spite of many improvements in the design techniques for FIR Half band filters, there still exist many challenges faced by the researchers. Though FIR half band filters have many advantages over IIR half band filters but they require higher filter order. The filter order increases with more stringent specification of HTs, i.e., smaller PBE and narrower transition width. Therefore, the minimization of Pass Band Error (PBE), Stop Band Error (SBE) with smaller complexity (order), have been always challenging. In this regard a number of techniques have been developed in last two decades for efficient design of HTs which include using frequency response masking (FRM) based half band filter [10]. In [11], a linear phase FIR filter with piecewise polynomial sinusoidal techniques. Identical sub filters have been used for multiplier less HT. An efficient design of FIR HT requires efficient FIR half band filter. For FIR filter designing, various evolutionary optimization-based algorithms have been reported because of their capability in converging to the global optima. The techniques published in this regard include simulated annealing [12], Genetic Algorithms (GA) [13, 14], Differential evolution (DE) [15–18], Particle swarm optimization (PSO) [19–22], Hybrid differential evolution and PSO (DEPSO) [23], Cat swarm optimization (CSO) [24], orthogonal harmony search algorithm [25], cat swarm optimization [24] and artificial bee colony [26, 27] algorithms. However, because of single objective optimization, a common problem with most of the classical and optimization-based HT design methods, is that they try to meet a certain objective therefore lack in fulfilling another requirement of the HTs. For example, HTs designed with the objective of small error poorly perform in terms of order, i.e., they require higher order FIR half band filter and vice versa. In smaller order FIR half band filter, both PBE and SBE can not be minimized simultaneously [28]. In addition to PBE and SBE this work presents conflicting nature of filter order. The condition of conflicting nature can be resolved by using a Lagrange's multiplier which allows to use different scaling factor for different objectives. However, the choice of scaling factor is again a big problem. Multi-objective optimization, which takes into account all possible solutions between conflicting many objectives, can solve such problems.

Unlike evolutionary optimization based single objective technique for HT design, which provide a set of coefficients, multi-objective technique provides a set of filters, from this set of filters a particular filter can be selected based on the requirement of the application of the HT.

In this work, the objectives are pass band error, stop band error of HT, order of prototype half band FIR filter and power consumption in HTs. The inclusion of power consumption while execution of HT, in addition to the response of HT is motivated by high operating frequency and device count in modern digital signal processing devices cause heating effect resulting in higher packaging and cooling related cost caused because of higher power consumption during HT execution. On the other hand low power devices are portable hence can be easily operated at low power, low cost and at remote places [29]. Power requirement in HTs can be reduced by improving prototype half band FIR filter implementation architecture [30] or by reducing transition activities [31] between filter coefficients in their digital form while execution. The average signal transition activity can be reduced by reducing information theoretic measure entropy [32–34]. The switching activity has been reduced by using gray code addressing in control path of the embedded processor [35]. The switching sensitivity has been measure for prototype FIR filters using entropy in DeBrunner et al. [36]. In the line of these works, in this paper minimization of entropy between consecutive coefficients of the prototype FIR half band filter, has been utilized in HTs design which relates to the power consumption. It has been observed using single objective optimization-based design of HTs optimized for PBE have higher entropy whereas HTs optimized for entropy have larger PBE. HTs with smaller order have larger PBE and vice versa. Therefore, the HT design problem with low power can be solved using multi-objective optimization. Evolutionary optimization approaches are characterized by population and natural selection determines the new population therefore these approaches are potential candidates for providing solution to multi-objective optimization problems.

Motivated by the capabilities multi-objective evolutionary optimization, in this work, a recently developed multi-objective seeker optimization algorithm (MOSOA) [37]. MOSOA is a meta-heuristic search control parameter which mimics the social exchange behavior found in a group of seekers. Classical MOSOA has been improved by adding a chaotic factor to improve its convergence and referred as chaotic multi-objective seeker optimization algorithm(c-MOSOA). The applicability of the proposed c-MOSOA has been evaluated by designing low power HTs. The response of the designed HTs have been compared with the classical MOSOA and other multi-objective evolutionary optimization approaches, i.e., non-dominated sorting genetic (NSGA-II) [38, 39], multi objective particle swarm optimization (MOPSO) [40] and multi-objective differential evolution (MODE) [41, 42] algorithms.

The major contributions of this work are firstly evolutionary seeker optimization-based HT design, secondly, formulation of HT as a multi-objective optimization problem, thirdly, validation of low power consumption using FPGA and Xilinx X-power analyzer.

The designed filters have been synthesized using Xilinx ISE 14.7. The power has been analyzed using X-power analyzer. The results of power consumption obtained using the proposed technique are compared with without power consumption results in other multi-objective evolutionary optimization algorithms.

The remaining sections of the manuscript are organized as follows. In the next section, formulation of HT as a multi-objective optimization problem has been discussed. Section 3 presents MOSOA and c-MOSOA step by step. Section 4 presents convergence of proposed c-MOSOA using KKT conditions. Section 5, describes the simulation and comparison of designed HTs with the state-of-the-art multi-objective optimization algorithms, and finally Sect. 6 concludes the paper.

2 Hilbert Transformer as a Multi-objective Optimization

The impulse response $h(n)$ for $n = 0$ to $L - 1$ of HT can be represented as

$$h(n) = \begin{cases} \frac{2}{n\pi} \sin^2\left(\frac{n\pi}{2}\right) & for \quad n \neq 0 \\ 0 & for \quad n = 0 \end{cases} \tag{2}$$

It can be observed from (2) that HTs are unstable as the impulse response is not absolutely summable. However, approximations to the ideal HTs can be obtained using half band FIR filters. Further the impulse response of HTs is anti-symmetric for positive and negative values of therefore FIR Hilbert transformers can be designed either by using FIR half band filters of type III or type IV. Consider a half band FIR filter of length N with real impulse response coefficients $h_{HB}(n)$, the transfer function of it is given by

$$h_{HB}(\mathrm{n}) = \sum_{n=0}^{N-1} h_{HB}(n) \tag{3}$$

In z domain, (3) can be represented as

$$H_{HB}(z) = \sum_{n=0}^{2M} h_{HB}(n) z^{-n} \tag{4}$$

where 2M is the order of FIR half band filter. In this work, a HT is designed using a half-band filter. Firstly, all the sample having value ½ at n = M, is replaced by the value 0. For this ½ is subtracted from all the coefficients. The subtraction of the coefficient by ½ shifts $H(\omega)$ by ½ downwards. Here, HT is obtained by a shift of $\frac{\pi}{2}$ horizontally rightwards. This shift is completed by multiplying remaining coefficients $h(n)$ by $(-j)^{-n}$. The transfer function of HTis given as

$$\mathcal{H}(z) = 2 \sum_{\substack{n=0 \\ n \neq M}}^{2M} h_{HB}(n)(jz)^{-n} = 2\left[H_{HB}(jz) - \frac{1}{2}(jz)^{-M}\right] \tag{5}$$

Thus the impulse response of half band FIR filter is related to the impulse response of HT is given by

$$\hbar(\mathrm{n}) = \begin{cases} 0; & \mathrm{n} = 2\mathrm{k}+1, \\ 2(-1)^{\mathrm{k}-1}\mathrm{h_{HB}}(\mathrm{n}); & \mathrm{n} = 2\mathrm{k} \ \text{ with } \ \mathrm{k} = 0, 1, 2, \ldots, \mathrm{M} \end{cases} \tag{6}$$

In this work, to obtain optimized HTs, the design problem has been framed as multi-objective optimization. In order to formulate HT as a multi-objective optimization problem, two different objective functions related to minimization of PBE and filter order has been formulated. The generalized the multi-objective optimization problem, with three objectives of HT design, can be stated as

$$minimize \quad J(h) = \{j_1(h), j_2(h), j_3(h)\} \tag{7}$$

subject to:$h \in H$,where $(h_1, \ldots, h_n)$ represent decision variables (HT prototype filter coefficients), $H \subset O$, H is the variable space, O is the objective space and $J\colon H \to O$, $J(h)$ is the objective vector. The proposed multi-objective algorithm has been applied to find the set of pareto (globally) optimal HT design solutions, which combinedly constitute the pareto set (PS). The set of all pareto objectives vectors, known as pareto front. The pareto front is represented as

$$PF = \{\ J(h) \in O | h \in PS\} \tag{8}$$

As discussed in the introduction, the primary requirements of a HT is smaller PBE at smaller filter order. However, both the objectives can not be achieved simultaneously. In this regard, in this paper, HTs have been designed by considering a multi-objective optimization with the two conflicting objectives. The first objective has been formulated by using an objective function which quantifies the deviation in the response of the designed HT from the no error in the pass band. Similarly the second objective function quantifies the corresponding order of the filter. The objective functions are represented as

$$J_1(h) = |(max|E(\omega)| - \delta_{pm}) - 1)| \quad for \ \ \omega \le \omega_p \tag{9}$$

$$J_2(h) = M(order\ of\ the\ prototype\ half\ band\ FIR\ filter,\ N-1) \tag{10}$$

where δ_{pm} are required maximum PBE, E(ω) is the deviation between desired and designed frequency response.

$$E(\omega) = \left[H_{dHB}\left(e^{j\omega}\right) - H_{IHB}\left(e^{j\omega}\right)\right] \tag{11}$$

$H_{dHB}\left(e^{j\omega}\right)$ and $H_{IHB}(e^{j\omega})$ are the frequency responses of the designed and ideal prototype half band filters for HTs. The desired response for the prototype half band low pass filter (LPF), is given as

$$H_{dHB}\left(e^{j\omega}\right) = \begin{cases} 1 & for \;\; 0 \le \omega \le \omega_p \\ 0 & otherwise \end{cases} \tag{12}$$

As discussed in the previous section, motivated by the importance of low power HT design, the third objective function for HT design has been formulated for power minimization using entropy of the half band FIR filter coefficients.

$$J_3 = \sum_{i=1}^{N} E(Cf_i, Cf_{i+1}) \tag{13}$$

where $[Cf_1, Cf_2, Cf_3 \ldots Cf_{N+1}$ are the coefficients represented in their IEEE 754 floating point representation, N is the length of the prototype filter. As entropy is a measure of the randomness carried by set of discrete events observed over time, the information content of the system can be measured by weighted sum of the information contents C_i by its occurrence probability P_i.

$$E(p) = \sum_{i=1}^{m} p_i \log_2 \frac{1}{p_i} \tag{14}$$

where p_i is the event occurrence probability which is related to information content C_i as

$$C_i = \log_2\left(\frac{1}{p_i}\right) \tag{15}$$

since $0 \le p_i \le 1$, the logarithm term is non negative, therefore $C_i \ge 0$. Therefore (13) reflects the overall switching in the binary values of the successive coefficients of the prototype half band FIR filter. For a filter having smaller entropy J_3 will have smaller switching activity hence the power consumption while HT execution will be low. The switching activity in a dynamic digital system is directly related to dynamic power consumption and given as

$$P_{dy} = \alpha_{0\leftrightarrow 1} C_L V_{DD}^2 f_{clk} \tag{16}$$

where $\alpha_{0\leftrightarrow 1}$ is the node switching activity factor, C_L is the load capacitance, V_{DD} is the supply voltage and f_{clk} is the operating clock frequency of the system. Thus in the present work multi-objective design problem has been formulated by considering three objectives i.e. J_1, J_2 and J_3. In the next section proposed c-MOSOA for multi-objective filter design has been discussed and compared with classical MOSOA.

3 Chaotic Multi-objective Seeker Optimization Algorithm

In this section firstly, classical multi objective MOSOA has been discussed. Then it is extended to the proposed c-MOSOA is presented. In this work, Pareto dominance based selection scheme is used for MOSOA. The scalar concept of optimality is not directly applicable to the multi-objective optimization problems. Therefore, in multi-objective optimization based techniques, a notion of Pareto optimality is applied. The concept of Pareto optimality in HT design (7) can be stated as: a HT prototype half band filter coefficient vector h^* is said to be Pareto optimal for a multi-objective filter design problem if all other vectors $h \in H$ have a higher value for at least one of the objective functions J_i, with $i = 1, 2, \ldots, n$ or have the same value for all the objective functions. A Pareto optimum can be defined as [43].

1. A point h^* is said to be weaker Pareto optimum for the multi-objective optimization problem if and only if there is no $h \in H$ such that $J_i(h) < J_i(h^*)$ for all $i \in \{1, 2, \ldots, n\}$.
2. A point h^* is said to be a strict Pareto optimum or a strict valid solution for the multi-objective optimization problem if and only if no $h \in H$, such that $J_i(h) < J_i(h^*)$ for all $i \in \{1, \ldots, n\}$ with at least one strict inequality.

3.1 MOSOA

In this section the fundamental steps of the classical MOSOA have been outlined as below:

1. The initial population of seeker at iteration t, is generated by D dimensional position vectors.

$$x_{ij}(t) = \left[x_{i1}(t), x_{i2}(t), \ldots, x_{ij}(t), \ldots, x_{iD}(t)\right] \tag{17}$$

where $i = 1, 2, \ldots, S$. x_{ij} is the jth element in the population and S is the population size. The seeker position x_{ij} is best if the fitness obtained using objective functions (7) is high. The fitness $f(x_{ij})$ is given as

$$f(x_{ij}) = \begin{cases} \frac{1}{J(x_{ij})}; & if \quad J(x_{ij}) > 0 \\ 1 + abs\big(J(x_{ij})\big); & if \quad J(x_{ij}) \le 0 \end{cases} \tag{18}$$

2. The total population is randomly categorized into three subpopulations. In all sub-populations, for ith seeker at iteration t, a search direction and step length vector are represented as.

$$d_i(t) = \left[d_{i1}(t), d_{i2}(t), \ldots, d_{ij}(t), \ldots, d_{iD}(t)\right] \quad (19)$$

and

$$\alpha_i(t) = \left[\alpha_{i1}(t), \alpha_{i2}(t), \ldots, \alpha_{ij}(t), \ldots, \alpha_{iD}(t)\right] \quad (20)$$

respectively, where $d_i(t) \in \{-1, 0, 1\}$ and $\alpha_{ij}(t) \geq 0$.

3. Using step length and direction the jth element of ith seeker position is updated by

$$x_{ij}(t+1) = x_{ij}(t) + \alpha_{ij}(t) \cdot d_{ij}(t) \quad (21)$$

The value of overall search direction $\alpha_{ij}(t)$ is calculated from the behavior of seekers. Seekers have two extreme types of cooperative behavior. One is egotistic and another altruistic. The direction associated with this kind of seeker is called egotistic direction, which is given as

$$d_{i,ego}(t) = sgn\left(p_{i,best}(t) - x_i(t)\right) \quad (22)$$

However, the seekers of altruistic nature co-operate explicitly with each other and adjust their behaviors in response to others. Seekers can have a local best behavior $(l_{best}(t))$ or global best behavior $g_{best}(t)$. Hence each seeker i is associated with two optional altruistic directions which are represented as (Fig. 1)

$$d_{i,atl1}(t) = sgn\left(g_{i,best}(t) - x_i(t)\right) \quad (23)$$

and

$$d_{i,alt2}(t) = sgn\left(l_{i,best}(t) - x_i(t)\right) \quad (24)$$

Moreover, seekers also have the properties of proactiveness. Seekers are able to exhibit a goal-directed behavior. In addition, future behavior can be predicted and guided by past behavior. To represent this, each seeker is associated with an empirical direction called proactiveness direction.

$$d_{i,pro}(t) = sgn(x_i(t_1) - x_i(t_2)) \quad (25)$$

where $t_1, t_2 \in \{t, t-1, t-2\}$ and $x_i(t_1)$ is better than $x_i(t_2)$. According to human rational judgment, the actual search direction of ith seeker, *i.e.,*$d_i(t)$ is a compromise among the empirical directions (19), (20), (21) and (22).

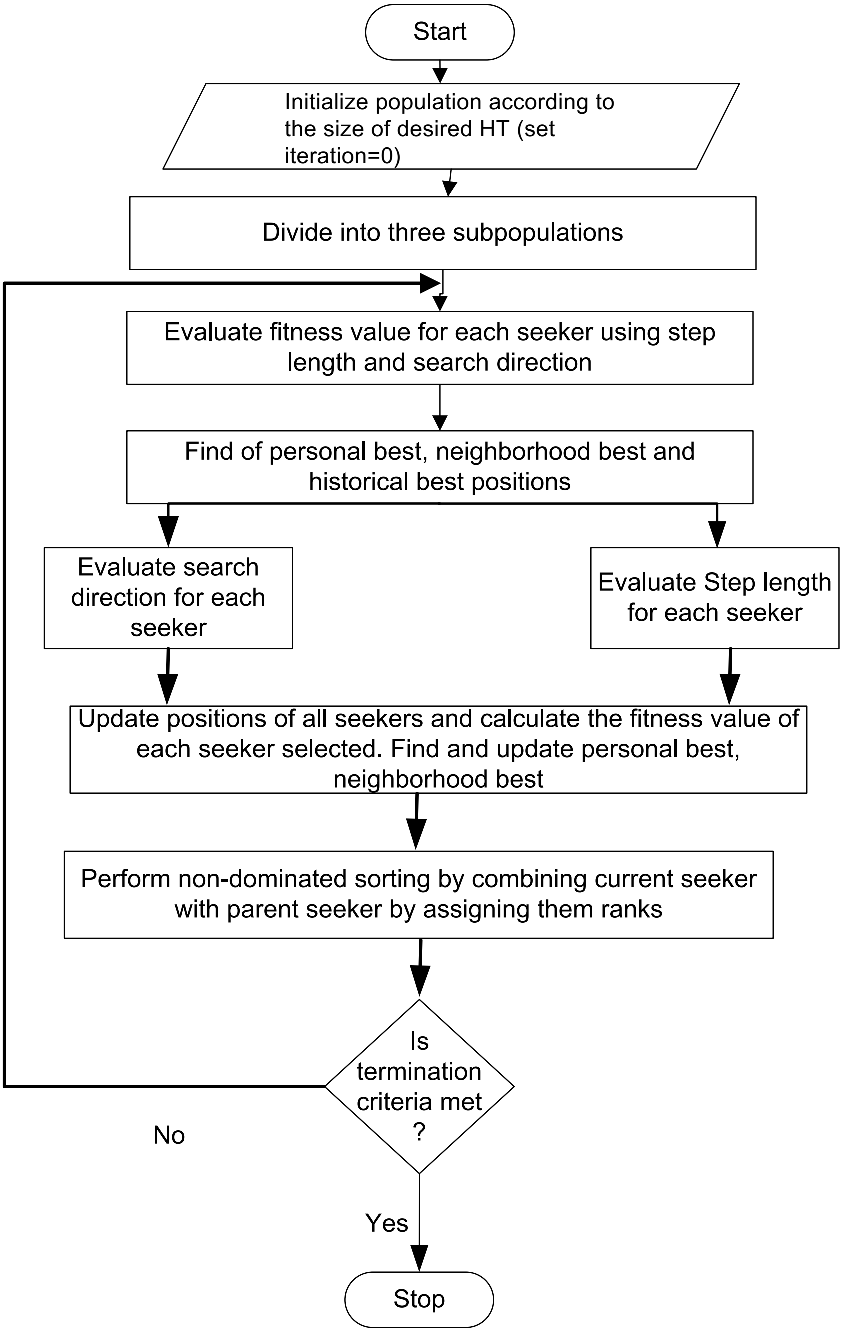


Fig. 1. Flow chart of proposed MOSOA

$$d_{i,j}(t) = \begin{cases} 0 & if \;\; r_j < p_j^{(0)} \\ 1 & if \;\; p_j^{(0)} < r_j \le p_j^{(0)} + p_j^{(1)} \\ -1 & if \;\; p_j^{(0)} + p_j^{(1)} < r_j \le 1 \end{cases} \tag{26}$$

where $i = 1, 2, \ldots, s$, $j = 1, 2, \ldots, D$. r_j is a uniform random number in [01] and $p_j^{(m)}$, ($m \in \{0, 1, -1\}$ is defined as follows: in the set$\{d_{ij,ego}, d_{ij,alt1}, d_{ij,alt2}, d_{ij,pro}\}$, let $num^{(x)}$ be such that$x \in \{-1, 0, 1\}$. Then,

$$p_j^{(x)} = \frac{num^{(x)}}{4} \tag{27}$$

The step length in (21) is decided based on fuzzy systems. To design the fuzzy system, fitness values of all the seekers are sorted in decreasing order and turned into the sequence numbers from 1 to s as the inputs of fuzzy reasoning. The linear membership function is used in the conditional part (fuzzification) , *i.e.*, {1, 2, ..., *s*} is presented as

$$\mu_i = \mu_{max} - \frac{s - I_i}{s - 1}(\mu_{max} - \mu_{min}) \tag{28}$$

where I_i is the sequence number of $x_i(t)$ after sorting the fitness values and μ_m axis, the maximum membership degree value that is equal to or a little less than 1.0. In this work, μ^{max} is considered as 0.96. In the action part (defuzzification), the bell membership function

$$\mu(\alpha_{ij}) = e^{-\alpha_{ij}^2/\left(2\delta_j^2\right)} \tag{29}$$

is used for *j*th element of *i*th seeker's step length. For the bell function, the membership degree values of the input variables beyond $[-3\delta_j \; 3\delta_j]$ are less than 0.0111 ($\mu \pm 3\delta_j$). The Parameter δ_j is the *j*th element of the array $\delta = [\delta_1, \delta_2, \ldots, \delta_D]$, which is given by

$$\delta = \omega_{lin} \cdot abs(x_{best} - x_{rand}) \tag{30}$$

where the term inside absolute provides a array such that each element of the array is the absolute value of the corresponding element of the input array. Further, the parameter ω is used to decrease the step length with increasing time step so as to gradually improve the search quality. In the present work, ω is linearly decreased from 0.9 to 0.1 during an iteration of the algorithm. In (30), best seeker and at random chosen seeker are represented as x_{best} and x_{rand}. The step length of seeker is calculated by

$$\alpha_{ij} = \delta_j \sqrt{-\log(rand(\mu_i, 1)} \tag{31}$$

where δ_j is the jth element of vector δ. Term inside square root, introduces ranmomicity in each element of α_{ij} and improve local search capability.

4. Since all the three sub-populations search individually based on their own information they can get trapped into local minima. In MOSOA this situation is avoided using non- dominated sorting approach. In this first step, combined population is obtained using $R_t = X_t \cup Y_t \cup Z_t$ where X_t, Y_t and Z_t are three sub-population.
5. After inter-subpopulation learning operation, the iteration is incremented by 1. The process from step 2 to 4 is repeated till one of the stopping criterions is met. Post convergence the algorithm in this work, a set of filter coefficient is obtained which removes noise from an image till maximum extent.

3.2 c-MOSOA

An evolutionary optimization algorithm must be efficient in exploration. Usually these have wider exploration in early stage whereas smaller exploration in the later stage. Based on these criterions, in the classical MOSOA, a linearly decreasing inertial weight ω_{lin} is considered in (30). However, to maintain exploration capability linearly varying weight is found to be ineffective. Therefore, with the aim of increasing the search space in addition to faster convergence. Chaotic searching behavior has been incorporated by considering chaotic weight

$$\omega_{lin,c}(k) = \omega_{lin}(k) \times \omega_c(k) \tag{32}$$

The chaotic term $\omega_{lin,c}(k)$ is evaluated using the widely used chaos theory [44] as

$$\omega_c(k+1) = \mu(k) * \omega_c(k) \times (1 - \omega_c(k)) \tag{33}$$

where $\mu(k)$ is the parameter that determines the extent of chaos. The initial condition for (33) is randomly chosen, i.e., $0 \leq \omega_c(k) \leq 1$. The chaotic weight includes ergodicity, irregularity and randomness in MOSOA.

Except change in inertia weight, remaining all the steps of MOSOA and c-MOSOA are same. Because of more diversity represented by chaotic probabilistic inertia weight, they perform better as compared to their continuous versions. In order to compare the proposed approach with other state of the art multi-objective optimization approaches (MOEAs) are also considered for filter design. Some of the other MOEA are non-dominated sorting genetic algorithm (NSGA), multi-objective particle swarm optimization (MOPSO) and multi-objective differential evolution (MODE). In the next section, these algorithms have been discussed in brief.

4 Other Multi-objective Optimization Algorithms Used

Three other state of the art multi-objective optimization algorithm considered for comparison, i. e., NSGA-II, MOPSO and MODE.

4.1 NSGA-II

NSGA was proposed in [45] was one of the earlier proposed multi-objective algorithms. The algorithm is used for classification of individuals according to their dominance using ranking. Each individual in the population is assigned a rank on the basis of non-domination. Each dominance class is assigned with a dummy fitness value proportional to population size. Since the process of Pareto ranking is repeated in each run therefore this algorithm is not found to be very inefficient. There is lack of elitism in the classical NSGA and there is need of specifying a sharing parameter. An improved version of this technique has been proposed, which is known as NSGA-II [38].

4.2 MOPSO

This algorithm is based on the study of bird behavior [40]. In MOPSO the initial population is initialized from random vectors. External archive is also used by MOPSO to store non-dominated solutions. A special mutation operator is incorporated to enhance the exploration capability of the particles. MOPSO is able to solve multi-modal non-convex optimization problems easily.

4.3 Mode

The multi-objective differential evolution algorithm focus on the concept of developing the populations over the generations, they are called as individuals [41, 42]. The individuals are encoded as

$$x_{ig} = \{x_{1g}, x_{2g}, \ldots, x_{Dg}\}$$

where i = 1,2,3, ..., Np. A mutated vector v_{ig} is generated corresponding to the target vector x_{ig}. The evolution is processed till the best value is obtained starting from random values.

$$v_{i,g} = x_{i,g} + F(x_{best,g} - x_{i,g}) + F(x_{r1,g} - x_{r2,g})$$

In MODE the crossover is performed using a random number for j = 1 to D to generate a target vector $u_{i,g}$.

$$u_{j,i,g} = \begin{cases} V_{j,i,g} & \text{if } (rand_{i,j}(0,1) < C_r \\ x_i, j, g & \text{otherwise} \end{cases}$$

where $i = 1, 2, 3, \ldots, Np$. The external archive in MODE is saved using non-dominated sorting-based selection.

5 Results and Discussion

The efficacy of the proposed Hilbert Transformer algorithm has been analyzed in two phases. In first phase, minimization of PBE and minimization of order has been evaluated. In the second phase, minimizing power consumption using entropy has been applied with an additional objective i.e.

Based on reported works [24, 46, 47], for the prototype half band FIR filter of proposed HT, a large number of pilot runs were executed, the initial parameters for MOSOA and c-MOSOA are selected as number of iterations 500, population size 50, limits for filter coefficients −1 to 1. The filters have been designed and simulated using MATLAB/SIMULINK 8.5. Further implemented using Xilinx ISE 14.7 on device-xc7vx485t-2ffg1761. The dynamic power consumption after implementation, has been analyzed using Xilinx X-power analyzer.

To evaluate the effectiveness of the proposed chaotic MOSOA, a low pass half band FIR filter design following specifications are used: normalized pass band frequency = 0.45 rad/s, normalized stop band frequency = 0.55 rad/s. Coefficients of filters are updated iteratively with the aim of minimizing J_1 and J_2. Figure 2 depicts the Pareto front obtained post convergence of c-MOSOA with two objectives and compared with MOSOA. In the Pareto front, each circle represents a set of filter coefficients.

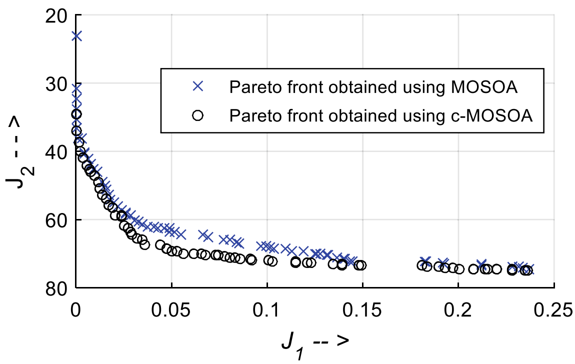


Fig. 2 Pareto front obtained for LPF design using MOSOA and c-MOSOA

From Fig. 2, the dominating solution of proposed technique c-MOSOAas compared to the conventional MOSOA is clearly reflected. The first filter provides min. PBE, the second filterprovides min. SBE and the third maintains a compromise between minimum PBE and order (J_1 and J_2). The magnitude response of the Hilbert transform design using corresponding filters are given in Fig. 3. These responses are compared with the magnitude responses of the respective filters obtained using MOSOA in Table 4.

It can be observed from Fig. 3 that the Hilbert transformer designed from small PBE filter coefficients is having less PBE whereas the Hilbert transformer designed using small order filter coefficients has larger SBE.

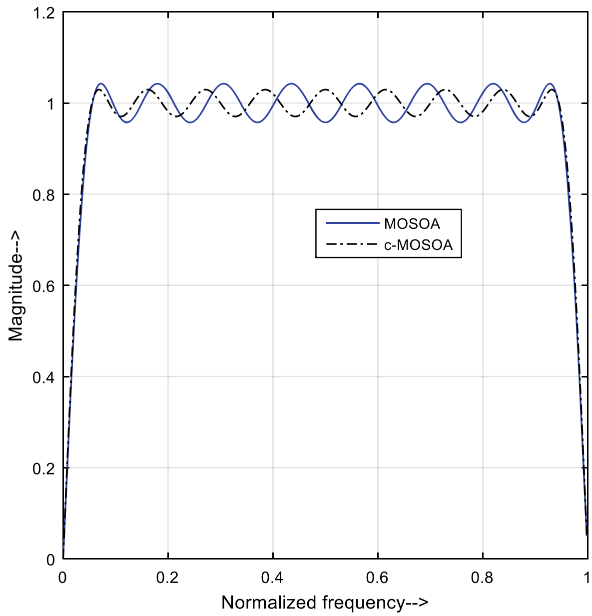
The proposed c-MOSOA algorithm has also been compared with other state of the art multi-objective evolutionary techniques: NSGA-II [38, 39], MOPSO [40] and MODE [48]. In Table 1, selected control parameters are shown. The algorithms are compared to meet the desired frequency domain specifications. The comparison has been shown in Table 2. The proposed c-MOSOA is found to outperform other multi-objective algorithms.

Considering the advantage of low power consumption, as discussed earlier, the third objective has been included in the multi-objective optimization. All the control parameters settings have been considered same as earlier. The three-dimensional Pareto front is shown in Fig. 4. To clarify the relation of J_3 with other objective functions J_1 and J_2, two other Pareto fronts have been depicted in Fig. 4(b) and 4(c) respectively. The efficiency of the quantum inspired multi-objective cat swarm optimization over the classical approach is clearly reflected. The filter coefficients obtained after convergence are shown in Table 3. The frequency responses of the filters at corners of Pareto optimal front is depicted in Fig. 5. Figure 5 represents Hilbert transformer designed by filter coefficients when J_1 is minimum. The responses of Hilbert transformers in Fig. 5b–d are obtained by filter coefficients from pareto optimal front when J_2, J_3 and $J_1 + J_2 + J_3$ respectively. The performance comparison is clearer in two-dimensional Pareto front as compared to three dimensional.

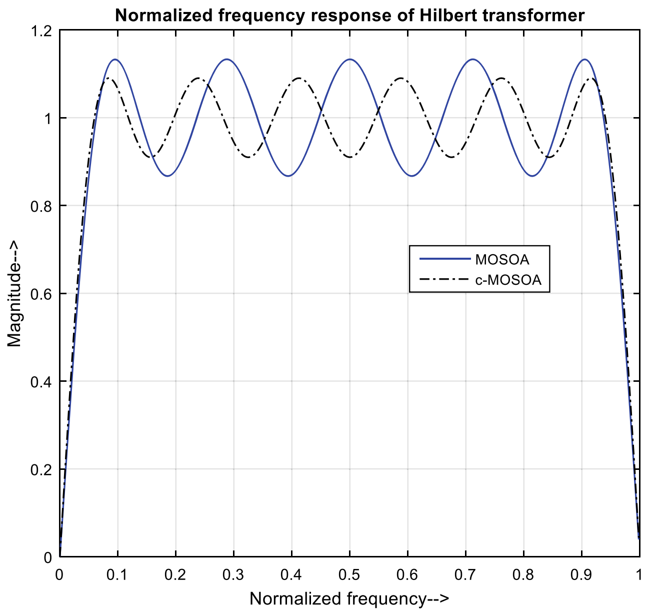
It is clearly visible from the Pareto optimal fronts figure that c-MOSOA performs better than classical MOSOA. The tradeoff between frequency domain specifications and power consumption is clearly reflected from Fig. 4a–c.

In filter execution amount of power consumed using the proposed algorithm has been compared with the MOSOA in Table 4. The effect of the objective function J_3 inclusion is clearly observed in the form of reduction in power consumption during filter execution.

In this paper, low pass filter design has been discussed with two and three objective functions. It can be observed that using multi-objective optimization; the filters can be designed with smaller errors. The designed filters have been compared with the classical multi-objective algorithms for filter designing. It has been observed that the approaches which are effective for low pass filters are equally effective in designing other filters also. Hence, the proposed quantum algorithm can serve as an excellent optimizer for Hilbert transformer design using prototype half band FIR filters (Fig. 5).

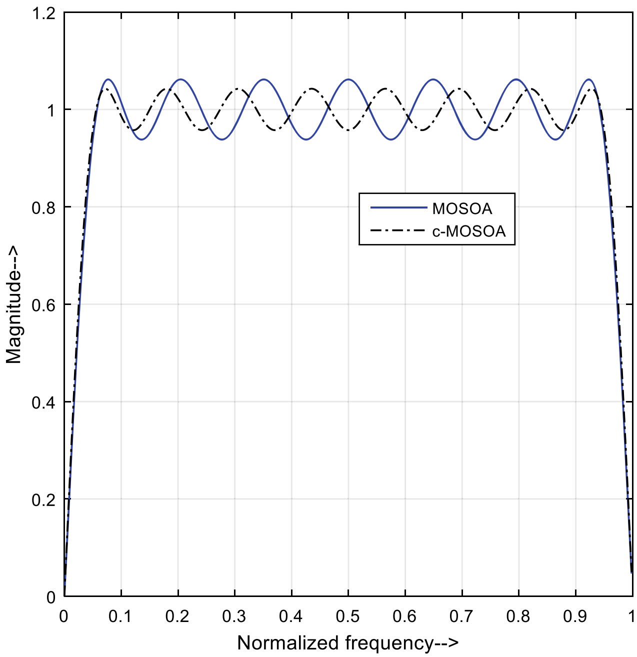


(a)



(b)

Fig. 3 Normalized magnitude responses of Hilbert transformers design of FIR filters for **a** minimum PBE, **b** minimum order, **c** compromised between minimum PBE and order



(c)

Fig. 3 (continued)

Table 1 Parameters for the applied multi-objective algorithms NSGA-II, MOPSO and MODE

Parameter	Algorithms		
	NSGA-II	MODE	MOPSO
No. of executions	100	100	100
Population size	100	50	50
Crossover rate	0.80	–	–
Mutation rate	0.07	–	–
Selection	Tournament		
Selection probability	1/3	–	–
C_1, C_2	–	–	1.5, 1.5
V_i^{min}, V_i^{max}	–	–	0.01, 1.0
C_r, F		0.4, 0.5	

6 Conclusions

In this work, a novel approach for Hilbert transformer designing using chaotic MOSOA has been proposed. The main objective was not only to minimize the errors in pass band at smaller filter order but also reduce power consumption during

Table 2 Comparison of Q-MOSOA with other multiobjective optimization approaches

Algorithm	J_1		J$_2$		Between (J_1 and J_2)	
	PBE	SBE	PBE	SBE	PBE	SBE
NSGA-II	0.059329	0.079769	0.059269	0.079789	0.091049	0.039329
MOPSO	0.039609	0.090479	0.079009	0.059969	0.089609	0.049389
MODE	0.029559	0.092329	0.090149	0.039949	0.089789	0.039869
MOSOA	0.019029	0.097189	0.094639	0.019129	0.069459	0.069459
c-MOSOA	0.009909	0.096549	0.096329	0.009909	0.059789	0.059779

execution of Hilbert transformer. In the previously reported works of HTs optimization, the only concentration was on minimizing errors or transition width of HTs. While the present work is motivated to reduce power loss, minimization of power was considered as one of the objectives. Using FIR filters, HT design has been performed. Further, the filter design has been formulated as a multi-objective optimization problem and solved using chaotic MOSOA i.e. c-MOSOA. The use of multi-objectives allows one to select a HT from a set based on requirements or application. The suitability of the proposed algorithm has been validated by comparing it with the classical MOSOA and existing multi-objective optimization evolutionary algorithms. c-MOSOA is found to outperform all other algorithms in meeting the specifications. A tradeoff between pass band error minimization and power consumption is observed in the pareto front post convergence i.e. it is practically impossible to design a HT which minimizes both simultaneously for lower order HT. The tradeoff avoids the usage of classical optimization methods involving single objective. Further, to verify the applicability of the proposed algorithm, for real time applications, the designed filters have been implemented using FPGA Virtex-7(device-xc7vx485t-2ffg1761). Both the simulated and experimental results conform the usefulness of c-MOSOA for designing HT with minimum PBE, order and power consumption.

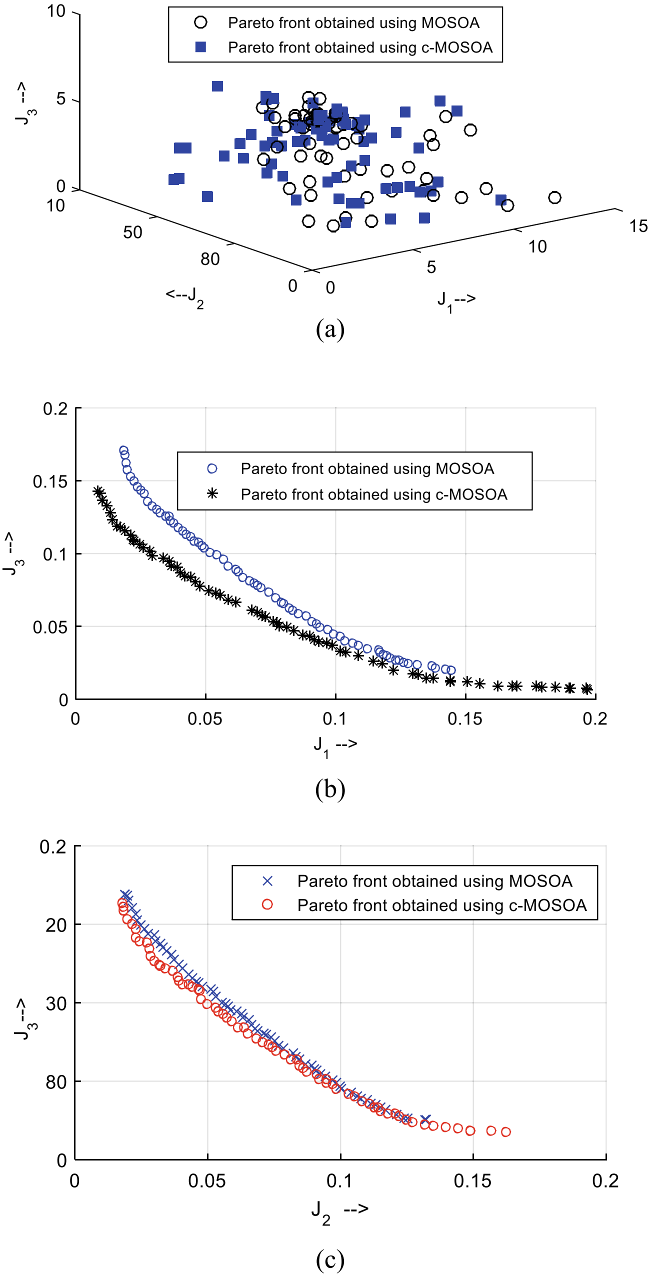
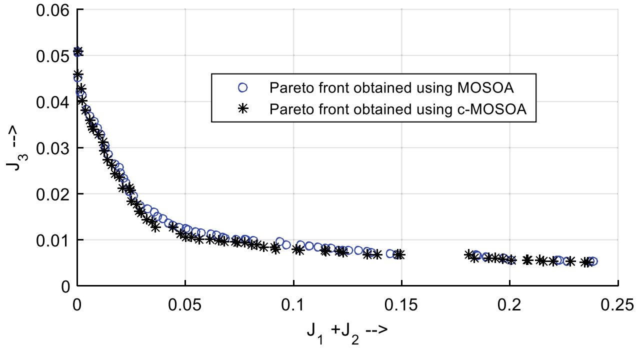


Fig. 4 Pareto front obtained using MOSOA and c-MOSOA after considering J_3 as third objective function for (**a**) J_1, J_2 and J_3 (**b**) J_1 and J_3 (**c**) J_2 and J_3 (**d**) $J_1 + J_2$ and J_3

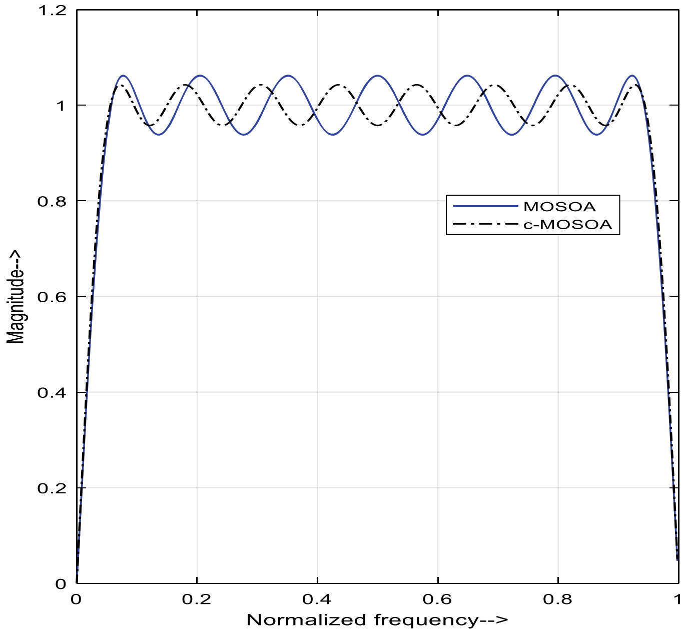


(d)

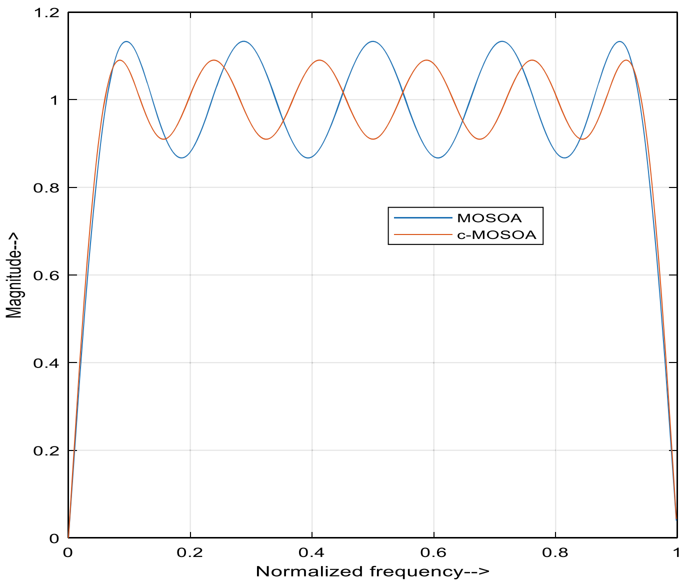
Fig. 4 (continued)

Table 3 Comparison of the proposed Q-MOSOA with other evolutionary multiobjective algorithms

Technique	Parameters											
	$\min(J_1)$			$\min(J_2)$			$\min(J_3)$			$\min(J_1 + J_2 + J_3)$		
	PBE	SBE	EN	PBE	SBE	EN	PBE	SBE	EN	PBE	SBE	EN
NSGA-II	0.020	0.223	1.22	0.521	0.036	2.33	12.00	6.120	1.88	0.110	0.161	2.12
MOPSO	0.038	0.185	2.50	0.492	0.020	2.71	11.00	6.950	1.40	0.122	0.140	1.10
MODE	0.042	0.184	3.55	0.533	0.018	2.60	6.040	8.020	1.31	0.070	0.072	2.90
MOABC	0.094	0.059	2.12	0.426	0.307	1.70	0.222	0.168	0.10	0.195	0.017	1.10
M-MOABC	0.018	0.1712	2.13	0.143	0.020	2.32	0.020	0.164	0.13	0.110	0.035	1.32

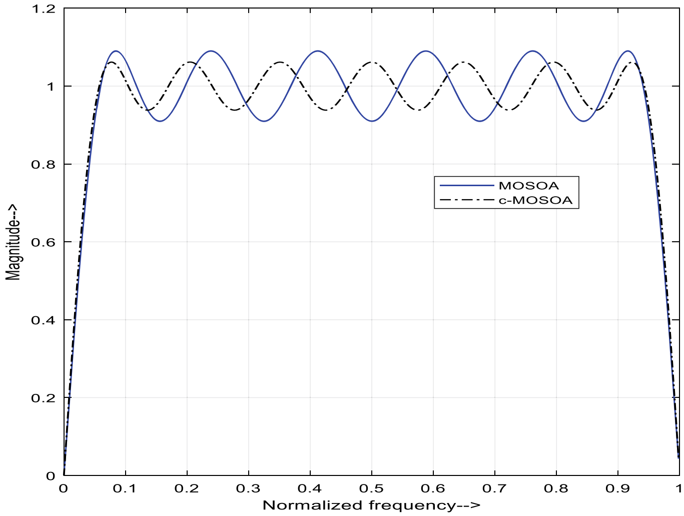


(a)

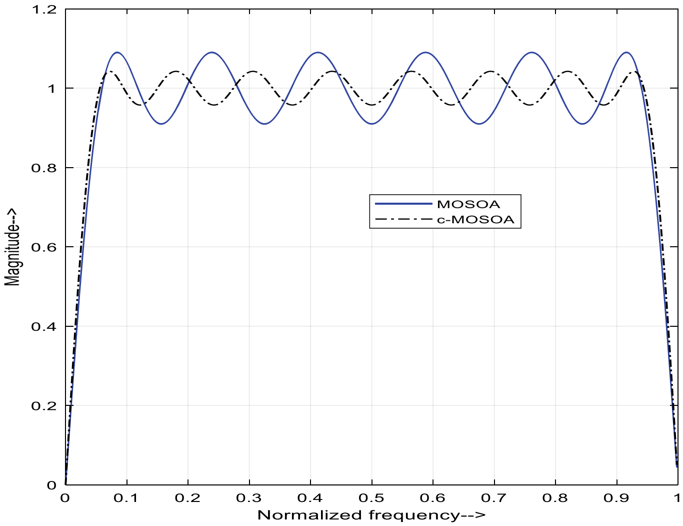


(b)

Fig. 5 Normalized frequency responses of hilbert transformers designed using FIR filters. (**a**) Minimum PBE, (**b**) minimum order, (**c**) minimum entropy, **d** tadeoff between minimum PBE, order and entropy



(c)



(d)

Fig. 5 (continued)

Table 4 Comparison of power consumed during filter execution using MOSOA and c-MOSOA

Filter Name	MOSOA		c-MOSOA	
	EN	PC (mW)	EN	PC (mW)
Table 1 (J1)	0.510	0.090	0.590	0.081
Table 2 (J2)	0.230	0.071	2.500	0.0910
Table 3 (J1)	1.131	0.150	1.381	0.070
Table 3 (J2)	2.660	0.110	1.751	0.081
Table 3 (J3)	0.150	0.0351	0.121	0.040

References

1. Gdeisat M, Burton D, Lilley F, Arevalillo-Herráez M (2016) Opt Commun 359:200
2. Gomes JGRC, Petraglia A, Trans IEEE (2002) Circuits Syst II Analog Digit Signal Process 49:177
3. Van Spaendonck R, Fernandes F (2002) 64th EAGE conference and exhibition, vol 1, p 1
4. Feldman M (2011) Mech Syst Signal Process 25:735
5. Klingspor M (2015) Hilbert transform: mathematical theory and applications to signal processing. Linkopings University
6. Wang S, Yan K, Xue L (2017) Opt Commun 383:537
7. Antoniou A (2006) Digital signal processing: signals, systems, and filters, 2nd edn. McGraw-Hill Professional
8. Samadi S, Igarashi Y, Iwakura H (1999) IEEE Trans Signal Process 47:1946
9. Lim YC, Yu YJ (2005) IEEE Trans Signal Process 53:2595
10. Lim YC, Yu YJ, Saramäki T (2005) IEEE Trans Circuits Syst I Regul Pap 52:2444
11. Lehto R, Saramäki T, Vainio O (2010) Circuits Syst Signal Process 29:25
12. Radecki J, Konrad J, Dubois E, Trans IEEE (1995) Circuits Syst II Analog Digit Signal Process 42:424
13. Lu HC, Tzeng ST (2000) Signal Process 80:497
14. Suckley D (1991) IEE Proc Circuits Dev Syst 138:234
15. Chandra AS, Chattopadhyay S (2012) 2012 International conference on communication devices intelligent systems. IEEE, pp 425–428
16. Karaboga N, Cetinkaya B (2006) Circuits. Syst Signal Process 25:649
17. Luitel B, Venayagamoorthy GK (2008) IEEE Congress evolutionary computation. IEEE World Congress on computational intelligence. IEEE, pp 3954–3961
18. Zhao Q, Meng G (2010) Internatioanl conference on information science and management engineering. IEEE, pp 177–180
19. Ababneh JI, Bataineh MH (2008) Digit. Signal Process 18:657
20. Boudjelaba K, Ros F, Chikouche D (2014) Circuits. Syst Signal Process 33:3195
21. Najjarzadeh M, Ayatollahi A (2008) IEEE international symposium on signal processing and information technology. IEEE, pp 129–132
22. Fang W, Sun J, Xu W, Liu J, Fang W, Sun J, Xu W, Liu J (2006) First international conference on innovative computing, information and control, vol I. IEEE, pp 615–619
23. Durbadal V, Rajib M, Vasundhara, Mandal D, Kar R, Ghoshal SP (2014) Nat Comput 13:55
24. Saha SK, Ghoshal SP, Kar R, Mandal D, Kumar S, Prasad S, Kar R, Mandal D, Saha SK, Ghoshal SP, Kar R, Mandal D (2013) ISA Trans 52:781
25. Saha SK, Dutta R, Choudhury R, Kar R, Mandal D, Ghoshal SP (2013) Sci World J 2013:1
26. Dwivedi AK, Ghosh S, Londhe ND (2015) Annual IEEE India Conference. IEEE, pp 1–6
27. Dwivedi AK, Ghosh S, Londhe ND (2016a) IET Signal Process 10:955
28. Dwivedi AK, Ghosh S, Londhe ND (2016b) Eng Appl Artif Intell 55:58

29. Nemani M, Najm FN (1996) IEEE Trans Comput Des Integr Circuits Syst 15:588
30. Arslan T, Erdogan AT, Horrocks DH (1996) Microelectronics J 27:731
31. Nemani M, Najm FNFN, Member S, Najm FNFN (1996) IEEE Trans Comput Des Integr Circuits Syst 15:588
32. Ramprasad S, Shanbhag NR, Hajj IN (1999) IEEE Trans Very Large Scale Integr Syst 7:359
33. Shanbhag NR, Trans IEEE (1997) Circuits Syst II Analog Digit Signal Process 44:935
34. Sotiriadis P-P, Tarokh V, Chandrakasan A (2003) Tit 49:790
35. Su CL, Tsui CY, Despain M (1994) IEEE Des Test Comput 11:24
36. DeBrunner VE, DeBrunner LS, Stephanie C, Hu X (2004) 3rd IEEE Signal Processing Education Workshop. In: 2004 IEEE 11th Digital Signal Processing Workshop 2004. IEEE, pp 97–101
37. Kumar D, Kamwa I, Samantaray SR, Gener IET (2015) Transm Distrib 9:1195
38. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) IEEE Trans Evol Comput 6:182
39. Fiandaca G, Fraga ESE, Brandani S (2009) Eng Optim 41:833
40. Coello CACC, Pulido GTT, Lechuga MSS (2004) Evol Comput IEEE Trans 8:256
41. Babu BV, Jehan MML (2003) Congress on evolutionary computation 2003. CEC ’03, vol 4, p 2696
42. Robič T, Filipič B (2005) Lecture notes in computer science, pp 520–533
43. Wang Y, Leung Y-W (2000) IEEE Trans Syst Man Cybern Part C Appl Rev 30:293
44. May RM (2004) The theory of chaotic attractors, vol 1, p 1
45. Srinivas N, Deb K (1995) Evol Comput 2:221
46. Pradhan PM, Panda G (2012) Expert Syst Appl 39:2956
47. Sharafi Y, Khanesar MA, Teshnehlab M (2013) 3rd IEEE International Conference on Computer Control and Communication. IC4 2013
48. Varadarajan M, Swarup KS, Gener IET (2008) Transm Distrib 2:720

Metaheuristics Applied to Blood Image Analysis

Ana Carolina Borges Monteiro, Reinaldo Padilha França, Vania V. Estrela, Navid Razmjooy, Yuzo Iano, and Pablo David Minango Negrete

Abstract The growing use of digital image processing techniques focused on health is explicit, helping in the solution and improvements in diagnosis, as well as the possibility of creating new diagnostic methods. The blood count is the most required laboratory medical examination, as it is the first examination made to analyze the general clinical picture of any patient, due to its ability to detect diseases, but its cost can be considered inaccessible to populations of less favored countries. In short, a metaheuristic is a heuristic method for generally solving optimization problems, usually in the area of combinatorial optimization, which is usually applied to problems for which no efficient algorithm is known. Digital Image Processing allows the analysis of an image in the various regions, as well as extract quantitative information from the image; perform measurements impossible to obtain manually; enable the integration of various types of data. Metaheuristic techniques have come to be great tools for image segmentation for digitally segmenting containing red blood cells, leukocytes, and platelets under detection and counting optics. Metaheuristics will benefit to computational blood image analysis but still face challenges as cyber-physical systems evolve, and more efficient big data methodologies arrive.

A. C. B. Monteiro (✉) · R. P. França · P. D. M. Negrete
School of Electrical and Computer Engineering (FEEC), University of Campinas—UNICAMP, Av. Albert Einstein, 400, Barão Geraldo, Campinas, SP, Brazil
e-mail: monteiro@decom.fee.unicamp.br

R. P. França
e-mail: padilha@decom.fee.unicamp.br

P. D. M. Negrete
e-mail: pablodavid218@gmail.com

V. V. Estrela · Y. Iano
Department of Telecommunications, Fluminense Federal University (UFF), Niterói, Brazil
e-mail: vania.estrela.phd@ieee.org

Y. Iano
e-mail: yuzo@decom.fee.unicamp.br

N. Razmjooy
Department of Electrical Engineering, University of Tafresh, Tafresh, Iran
e-mail: navid.razmjooy@hotmail.com

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_6

Keywords Metaheuristics · Image processing · Blood image analysis · Optimization · Computational intelligence · Image description

1 Introduction

Clinical analysis laboratories have a variety of methods and tests that help doctors diagnose diseases of all kinds. Very common, blood tests are the starting point for the patient to receive accurate information about any disease that affects him, and also about the current state of his health. The blood count is the most common and most used test for the initial screening of some diseases, from which it is possible to know about the physiology of the patient from his blood. Bone marrow is the site of blood cell formation: leukocytes (white blood cells), red blood cells (red blood cells) and platelets. The primary function of the red blood cells is the transport of oxygen, the leukocytes are the defense of the organism, and the platelets are the blood coagulation [1].

The blood count provides an analysis of the three major blood components (erythrocytes, leukocytes, and platelets) consisting of red blood cell count, hemoglobin, hematocrit, red cell, leukocyte index (global and differential) and platelet count. Figure 1 depicts the blood analysis rationale and types of analytical procedures. Thus, the test evaluates the quantity and quality of red blood cells (red blood cells that carry oxygen and nutrients to the body), leukocytes (white blood cells, which act on the immune system), and platelets (which modulate coagulation) [2].

It is a paramount diagnostic and control test for hematological and systemic diseases. It is routinely indicated for the evaluation of anemia, hematological malignancies, infectious and inflammatory reactions, follow-up of drug therapies, and evaluation of platelet disorders. In general, it identifies diseases that mess up the composition of the blood, such as leukemia, bacterial or viral infections. Allergies

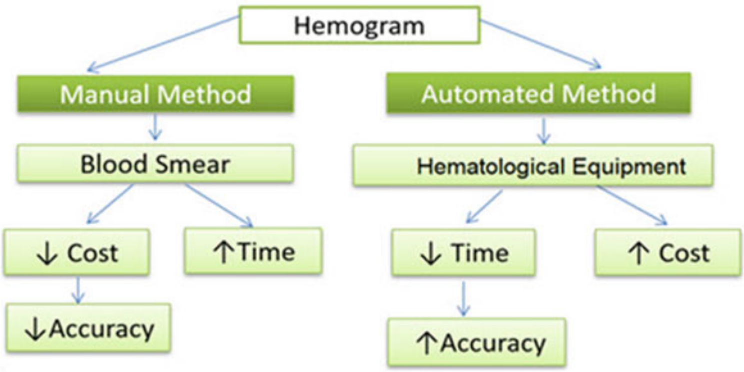


Fig. 1 Blood count methodologies [1–5]

and bleeding can also be detected with it. The blood count guides the differentiation between viral and bacterial infections, parasitic infections, inflammation, intoxication and neoplasms through global and differential leukocyte counts and their morphological evaluation. The blood count is also used to ensure that one can undergo surgery. Additionally, even for how to check the body's reaction to specific treatments [3].

In clinical analysis laboratories, a careful process of collecting, storing, transporting samples, performing, analyzing, typing, and releasing results is fundamental for the results of these tests to bring a correct interpretation of the patient's clinical status. As well as an impeccable standardization in all steps of the laboratory procedures is of fundamental importance to obtain the accuracy and precision of the results. All of this must also be associated with proper facilities, calibrated equipment, and skilled personnel. It follows that this process is insightful and involves intense human labor. Even considering the high cost of equipment, which in addition to needing periodic maintenance, in many scenarios of underdeveloped countries this reality is nonexistent [4]. Figure 2 displays the whole blood analysis pipeline for a typical pathology laboratory. Since parts of this laboratory can be geographically distributed, then it can be associated to a cyber-physical system [7].

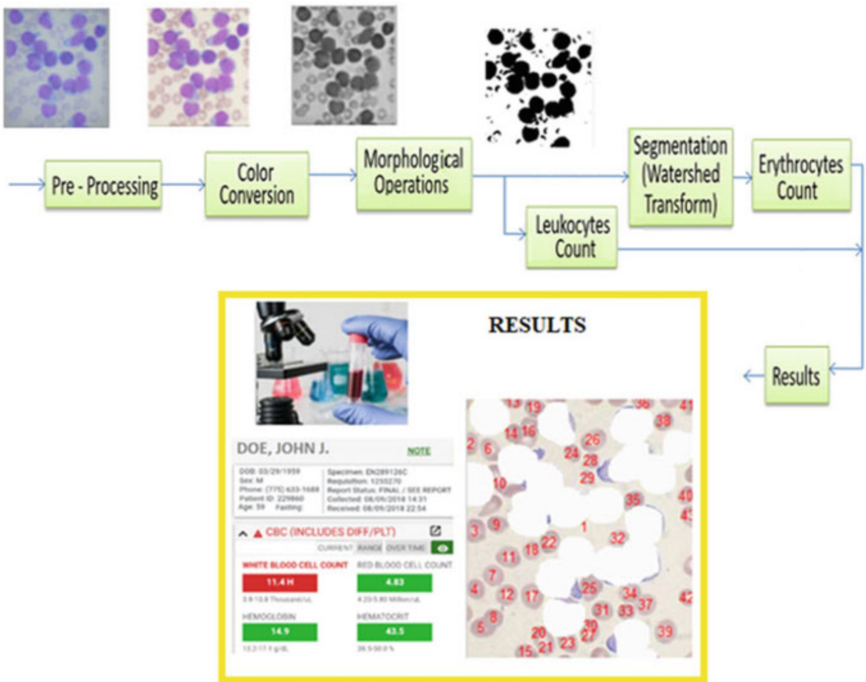


Fig. 2 Blood-processing pipeline [1–7]

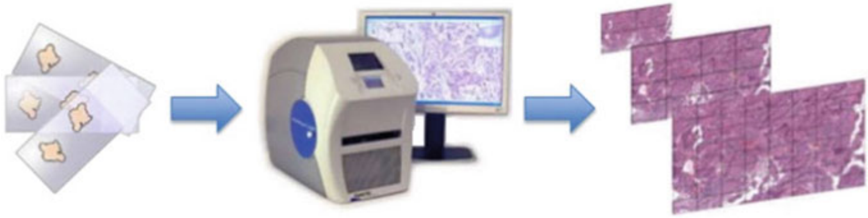


Fig. 3 Whole slide imaging [35, 36]

During the last decades, there has been a significant technological evolution in the accomplishment of these exams, and the manual techniques can and have been replaced by automated systems that present more precision in the results and in a shorter time interval. These innovations have changed the routine of laboratories, making them more efficient and agile, and presenting a better quality of results. Since the automation of the processes of these exams has been advantageous, ensuring the reliability of the hematological tests in all phases, because the standardization guarantees the efficiency of the results [5].

Technological advances have brought reliability to blood count results. Nonetheless, in order to offer the physician greater possibilities for the correct interpretation of these tests, new methods can be developed with more exceptional technological support. Given this scenario, metaheuristics provide more excellent stability and precision, avoiding errors, through artificial intelligence training, an even greater range of efficiency can be achieved [1–5, 8, 9].

In this sense, and because of the medical fields' dependence on new developments, it is clear that it is necessary to develop new technology-based methods, which are cheaper, thus having a wide range, such as the use of digital blood cell imaging., and through image segmentation algorithms that meet the efficiency and reliability criteria for blood cell detection and counting [10].

Recently, metaheuristic techniques have gained prominence and relevance, where they have been used to assist in the optimization problem solving, are high-level procedures designed to search, generate or select a heuristic, which provides a sufficiently right solution for a given problem. A metaheuristic has to coordinate local search procedures with higher-level strategies, to create a process that can escape local minimums and perform a robust search in the solution space. In short, a metaheuristic is a process that combines concepts of exploration and exploitation, seeking in large portions of the search space the refinement for a promising solution [11–13].

Metaheuristics are applied to solve problems about which there is little information, but once a solution is offered, candidates can be tested. However, they have no guarantees of optimality. Techniques based on metaheuristics are usually inspired by some biological strategy or behavior of nature, consisting of simple local searches and complex learning procedures, even considering that they do not have specific domains and can be applied to any problem. Briefly described, these are strategies that guide the search process in search of near-optimal solutions [12].

Metaheuristics have emerged as possible algorithms for dealing with complex optimization problems, which are difficult to solve using traditional methods. In this context, there are several types such as Genetic Algorithms (GAs) [14, 15], Fuzzy Logic (FL), Particle Swarm Optimization (PSO), Deep Learning [16], Ant Colony Optimization [17, 18], and Differential Evolution [19], all regarded as highly promising techniques for optimization and being used in solving various problems in real-time. Still considering that many times in image processing one finds irregular forms, difficult to recognize and classify, either by overlap or even the actual contour of an object of interest in an image. Where all these problems are overcome by the use of global stochastic optimization metaheuristic techniques designed to solve complex optimization problems, yet taking machine learning into account as a strong ally in this regard. Thus, the application of metaheuristics in image segmentation is efficient [20].

Therefore, this chapter aims to discuss metaheuristics in discrete-event simulation, categorizing, and synthesizing the potential of technologies that involves thematic. Thus, Sect. 2 explains the blood analysis via image digital processing. Metaheuristics in blood image analysis are explored in Sect. 3. Section 4 shows some case studies. Lastly, Sect. 5 outlines future trends in technology. Finally, Sect. 6 concludes the chapter.

2 Blood Analysis via Image Processing

The computer vision seeks to help solve highly complex problems, seeking to mimic human cognition and the ability of humans to make decisions according to the information contained in the image. In this field is the medical image analysis, which is a fusion of visual examinations and information, since, with them, it is possible to perform segmentation and identification of areas, regions, and elements in Image, as well as to perform Extraction of Geometry Characteristics of interest [21].

Digital image processing consists of a set of techniques for capturing, representing and transforming images with the aid of a computer. The use of these techniques makes it possible to extract and identify information from the images and to improve the visual quality of certain structural aspects, facilitating human perception and automatic interpretation employing machines [22].

In biological imaging, it is commonly used to verify the contour found that the cells present were correctly classified, even in the case of small regions, in the specific case of blood images. The analysis or interpretation of images aims to obtain a description that contains enough information to distinguish between different objects of interest, reliably and requiring minimal human intervention [23].

A digital image is described by an $N \times M$ matrix of positive integer pixel ($p(x, y)$) values, which indicates the tone at each position $[x, y]$ of the image. Image analysis is typically based on the shape, texture, gray levels, or colors of the objects in the images. An inherent difficulty in the process of image analysis is its multidisciplinary character, in which several domains of knowledge are commonly needed to satisfactorily solve the problem, such as computational geometry, visualization, science, psychophysics, statistics, information theory, and many others [24, 25].

This means that for each sampled pixel, the digital image usually has an integer value corresponding to a shade of gray, or entry number in a color table, or 3 integers corresponding to the bands R (red), G (green) and B (blue). The increasing advance of digital technology, associated with the development of new algorithms, has allowed expanding the number of pathology applications [24, 25].

Medical diagnostics can be aided with the use of digital imaging, where it is already known that various fields of medicine have benefited from improved diagnostics through imaging, in particular, oncology, in particular, cases blood imaging. The analysis and interpretation of these images make it easier, for example, to identify lesions or regions affected by cancer, allowing physicians greater accuracy and faster diagnosis as well as better planning of treatments and surgeries [26].

A digital image processing system usually possesses a set of steps to yield a result from the problem domain. Knowledge about this specific problem domain is encoded in an image processing system in the form of a knowledge base. This knowledge base is dependent on the solution found, whose size and complexity can vary significantly and can be used to guide the communication between processing modules in order to perform this particular task [27].

Normally, a digital image is the result of the acquisition process, which may generally present imperfections or degradations resulting from lighting conditions or devices' characteristics. A preprocessing step ameliorates the image quality by applying techniques for noise attenuation, contrast or brightness correction, and smoothing specific image properties [28].

In sequence, generally, the segmentation step performs the extraction and identification of areas of interest contained in the image, which relies on the detection of discontinuities (edges) or similarities (regions) in the image. Thus, recognition or classification is the process that assigns an identifier or label to image objects based on the characteristics provided by their descriptors [27–33].

The blood test is one of the most widely used clinical analysis procedures for the full spectrum of anomalies it can detect. Where through the blood cell count, which is the subject of interest of these examinations, in reality, a manual count is made by an operator who microscopically examines an eventually treated or colored sample under the microscope. The shape of blood cell contours, for example, may aid in the diagnosis of anemia by automatically counting cells in a blood sample. Which from image processing, this task, as described above, is much more efficient and can reach high levels of accuracy due to the technological content [1–3].

Thus, image processing techniques are usually based on mathematical methods that allow the quantitative description of images from various sources. An image can be described regardless of what it represents and considering all its parameters which have a two-dimensional or topological feature. In short, in a typical 2-D image, each object defined in this 2-D space has surface measurements, lengths, thickness, perimeters, position, among other features that can be measured digitally, and then deduced statistical quantities in a digitally automatic way [27, 34].

Whole Slide Imaging (WSI) creates a material that can be stored in hematological databases as part of PACS [37–42]. These histological repositories can contain images from a broad range of modalities, multidimensional imageries, as well as asemantic data. These collections permit evidence-based analysis, health training, and research. Hematological information retrieval needs suitable procedures to explore the groups of images with the possibility of handling characteristics similar to the case(s) of concern. Content-based Image Retrieval (CBIR) is related to image queries to complement the traditional text-based access to images via visual features, e.g., color, texture, and shape, as adequate criteria to perform searches [40, 43]. Medical CBIR is in its infancy when applied to highly multidimensional and multimodality healthcare data. Contemporary biomedical CBIR methodologies are 2-D image retrieval, 3-D or higher-dimensional image retrieval, and semantic (non-image information) to enrich the blood element retrieval from assorted research groups' datasets. Blood smear images can be organized into several categories as a framework for accessing different pathologies, aiming to the features and modalities of the knowledge gathered during laboratory exams, medical image retrieval, and the use of metadata.

3 Metaheuristics in Blood Image Analysis

In short, metaheuristics may encounter useful or even optimal solutions to a given problem, consisting of the iterative application of a secondary heuristic (local search), having some mechanism to escape from great places (valleys). Metaheuristics relying on population search with various solutions are those that maintain a set of reasonable solutions and combine them to produce even better solutions, and usually do not perform refinement procedures, i.e., local search, and so, must maintain a set of current solutions [44].

3.1 Genetic Algorithms

Genetic Algorithms (GA) are optimization and search methods based on the evolutionary mechanisms of living beings, being algorithms based on the theory of naturalist Darwin, who states that individuals more adapted to their environment have a higher chance of surviving and generating offspring. These GAs belong to the class of probabilistic algorithms, but they are not purely random search methods because they combine directed and stochastic search elements [45].

These algorithms employ a terminology originated from the theory of natural evolution and genetics, having a logic of operation relative to an individual of a population is represented by a single chromosome, containing the coding, genotype, of a candidate to solve the problem, phenotype. A chromosome is usually implemented in the form of a vector as a list of attributes, where each component is known as a gene. Thus, GAs are particularly applied to complex optimization problems, either problem with many constraints or conditions that cannot be represented mathematically; whether they have characteristics that need to be combined in search of the best solution or with several parameters; or problems with large search spaces [14].

They may straightforwardly solve complex problems and had their methodology based on a set of some bit strings, 0s and 1s, called individuals. Thus, the system evolves until the best chromosome meets a specific problem, even without knowing the characteristics of the problem that is being solved. Furthermore, this solution is encountered in an automatic and unsupervised way, and the only information given to the system was adjustments to the chromosome. Several works in digital image processing encompass the identification and classification of regions, which, although indispensable, involves a high degree of complexity. Consequently, GA models for image classification based on the genetic evolution of association rules from the color and texture attributes of a training sample set can be used [15].

Just as in 2008 an application of a GA in conjunction with a Support Vector Machine (SVM) was applied to the recognition of blood cells from the aspirated bone sample image. In this research, the focal task of the GA was to select the features for the SVM for targeted recognition and final classification, which showed that applying GA was a powerful tool for selecting diagnostic features, leading to a significant improvement in overall system accuracy. Since it has been taken into account that the counting and evaluation of blood cells in the bone marrow of patients are very informative in clinical practice, there are different cell lines in the bone marrow, the most important of which are the granulocytic and lymphocytic series, white blood cells, and erythrocytic, which is red blood cells. Still considering the geometric features that describe different aspects of cell geometry and using descriptive parameters for radius, perimeter, area, as well as symmetry, is the difference between lines that are perpendicular to the central axis and the cell boundary. Therefore, in this context, it was seen that the application of GA in the selection of characteristics for recognition of neighboring blood cells. The advantage of GA was that it increased blood cell recognition accuracy by more than 25% (in relative terms) [46].

From the previous facts, it was possible, and it is clear the viability of the use of GAs in image classification. GA image classification opens the way for a wide range of possibilities in the most diverse knowledge arenas, including the identification and classification of regions with color or hyperspectral images [14, 15, 45, 47].

3.2 Fuzzy Logic

Fuzzy Logic (FL) can be defined as a tool capable of capturing vague information, has applied logic to deal with situations where there is a certain degree of uncertainty, often achieving better results than classical logic, it fills a gap between human communication and computational systems, generally described in a natural language and convert them to a numerical format, easily manipulated by computers today. Fuzzy Set theory is the FL building block, and is better suited to addressing information imperfections than probability theory [48].

While some systems operate using classical logic, where facts/events are true (1) or false (0), there is nothing between these two options; When using fuzzy logic, there are new possibilities, as it is possible to use the values between 0 (false) and 1 (true). Traditionally, a logical proposition has two extremes: either it is utterly true or it is entirely false. Nevertheless, in FL, a premise varies in degree of truth from 0 to 1, which leads to being partially true or partially false. These associations come about through membership functions, which can be triangular, trapezoidal and sigmoidal [49].

The automatic detection of blood components done in 2014 by [50] is a significant landmark in the field of hematology. Image segmentation allowed grouping the blood components so that they could be processed separately. This method classified and segmented blood constituents from microscopic images aka Whole Slide Imaging (WSI) for automatic analysis, using a general and automatic diffuse approach. During preprocessing, fuzzy sets were automatically calculated based on the histogram peaks relative to the green channel of the RGB image and the Euclidean distance between the leukocyte nucleus centroids and the remaining pixels, where 530 microscopic smear images were processed. The results were compared with the results of manual segmentation by experts and with the accuracy rates of other approaches surveyed. During processing, the fuzzy process associates the degree of the pertinence of the gray level of each pixel in the regions defined in the histogram with the proximity of the centroid of the leukocyte nucleus closest to the pixel. The postprocessing lessens false positives with the segmentation of leukocytes, considering the included nucleus and cytoplasm, erythrocytes, and blood plasma. Likewise, fuzzy rules are applied to the image, ensuing four classification regions: leukocyte nuclei, leukocyte cytoplasm, erythrocytes, and blood plasma. Demonstrating average accuracy rates of 95.06% for blood plasma, 97.31% for leukocytes and 95.39% for erythrocytes [50].

3.3 Particle Swarm Optimization (PSO)

PSO refers to a family of population metaheuristics for the optimization of functions built on mechanisms for simulating the social behavior of animals, such as bird flocks. It uses as a metaphor for the mode of influence on ways of thinking and acting among human beings: a particle set navigating through the solution space. In this structure, each particle establishes its trajectory by relating its past experiences with those of its neighbors, as well as other particles with which they communicate with [51, 52].

PSO has the strengths of being a robust, flexible, simple, and highly distributable technique with small memory requirements, low processing power, and fast convergence to the optimum. On the other hand, its negative point is the rapid loss of diversity and premature solution convergence to extrema [53].

In short, the PSO aims to simulate the swarm behavior found in nature in certain types of animal species, where, when moving from one point to another, there is always one or more pack leaders impelling the movement of the others. Thus, PSO is considered as an evolutionary computation technique, although it brings in a different metaphor to the evolution of species [51–53].

In 2005, a resource extraction algorithm constructed on PSO for hyperspectral images [47] focused on data visualization to extract the resources that generate the best visualization of an area covered by the blood (blood smear). This research had a binary PSO version for choosing a subset of wavelengths in the near-infrared region. In this procedure, the optical blood absorption physiognomies allow extracting some visual information. A linear image transformation assists features' selection, employing two transformation equations that will pick up three and four bands. The transformed image was evaluated using four different aptitude criteria: entropy, Euclidean distance, contrast, and correlation. Experimental results showed that entropy better assessed the amount of visual evidence under the blood layer, and the four-band transformation formed the better viewing, as well as enhanced images of extracted features, revealed decent views under the layer of spilled blood [54].

3.4 Watershed Transform

The Watershed Transform (WT) is an image segmentation inspired by the division of surfaces in watersheds, having several forms of definition and algorithms. WT proposes a metaheuristic to the image segmentation problem, interpreting them as surfaces, where each pixel corresponds to a position, and the gray levels determine the altitudes. This notion prompts the necessity to identify watersheds, defined by regional minimums and their domain regions. Intuitively, the WT tries to find the points on a surface where a drop of water can drip to two different regional minimums [55].

In 2017, the need for reliability, practicality, and agility focused on the creation of new health tools was considered, taking into account the blood cell count as an indispensable exam that diagnoses various diseases. Being proposed an algorithm for counting and recognition of Red Blood Cells (RBC) and White Blood Cells (WBC) using Matlab software, using the Watershed Transform in the segmentation and counting of these cells, where its results showed the best performance of 34% in execution time, as well as a computational performance of 1.98 s [56].

In 2018 it was researched that blood tests have direct help in detecting various types of diseases using erythrocyte count (erythrocytes) and leukocytes (leukocytes).

Red blood cell and leukocyte evaluation have a direct impact on the diagnosis of leukemia, anemia, viral, parasitic, and viral infections. In this context and given the importance and applicability of Watershed Transform and Morphological Operations, it is possible to perform medical image segmentation. Thus, the developed WT-MO algorithm was used for detection, segmentation, and counting of these blood cells focusing on the criteria of efficiency and reliability. The Watershed Transform-based WT-MO algorithm was highly accurate (93%), running on different hardware platforms with average and processing time of fewer than 3 s per sample. Thus, the

WT-MO algorithm was considered accurate and reliable and can be applied as a third methodology for performing laboratory tests accelerating medical diagnosis [1].

In 2019 it was seen that it is becoming more common to use engineering techniques in health areas aimed at solving simple problems or even creating new diagnostic methods. Hough Transform has also been studied, which has been used as a tool for the segmentation of blood smear images for counting blood cells. However, he noted that the Watershed Transform was applied more efficiently to the same function. Based on this, a methodology based on the Hough Transform was developed focusing on the detection and counting of erythrocytes and leukocytes and subsequent comparison with a methodology developed with Watershed Transform called WT-MO [57].

3.5 Deep Learning

Deep Learning is a type of machine learning that trains computers to perform tasks such as humans, which includes speech recognition, image identification, and changes. Instead of organizing the data to be executed using standard equations, either Deep Learning sets the basic values about the data and trains them or pattern recognition methodologies takes care of various processing steps. It is an emerging theme within the machine learning subcategory that relates to neural networks to ameliorate things. This is fast becoming one of the most studied and sought-after fields within modern computer science [58].

Deep learning techniques have enhanced the ability of computers to classify, recognize, detect and describe (in a word, understand). Deep learning can classify images, recognize speech, detect objects, and describe the content. It performs the training of a computational model so that it can decipher the natural language. This model relates terms and words to infer meaning from enormous amounts of data [59].

It takes much computational power to solve deep learning problems because of the iterative nature of their algorithms, their complexity that grows as the number of layers increases, and the massive amounts of data required to train networks. The dynamic nature of deep learning methods presents an excellent opportunity to introduce more dynamic behaviors to analytics, given their ability to continually ameliorate and adapt to changes in the underlying information pattern [60].

Using deep learning techniques, it is possible to classify WBC images as long as the density of leukocytes in our bloodstream provides a glimpse of any potential risks that the body may be facing as well as visualizing the state of the immune system. In particular, a dramatic change in cell count is usually a sign that the body is being affected by an antigen, just as a variation in a specific type of leukocyte is usually correlated with a specific type of antigen [61].

To do this kind of work with this technique, a data set consisting of n leukocyte images is required (it is worth noting that the larger the number of images, the higher will be their accuracy since artificial training will be on top of this dataset). This dataset should be labeled by subject matter experts to make training accuracy more refined. Thus, training is performed by gauging a preprocessing to normalized this data set and then processing the artificial training [59, 60].

In 2018 was implemented deep learning algorithms, especially Convolutional Neural Networks (CNN), bring considerable benefits to the medical field, where a large number of images can be processed and analyzed. This algorithm aimed to classify blood cells, being one of the most challenging problems in blood diagnosis, so a CNN-based framework for automatic classification of blood cell images into cell subtypes was developed. A dataset of 13,000 blood cell images with their subtypes was used, showing high results regarding the classification [4].

4 Case Studies

While heuristics generate viable solutions of good quality, but without quality assurance, being understood as an approximate method, it is designed based on the structural properties and characteristics of these problems, with reduced complexity about that of conventional exact algorithms [62, 63].

In summary, metaheuristics are techniques used in situations where closed mathematical or logical modeling is challenging to get. Computational intelligence entails the combination of basic heuristics at a higher structure level. These methods are optimization tools and viable for solving complicated or extensive problems, where a large number of variables and constraints make the usage of exact methodologies unfeasible. The combination of random choices and historical medical knowledge assimilated by the method guides exploration of the search space to the adequate neighborhoods, which circumvents premature stoppages in poor local optimal locations. Hybrid methods frequently employ a strategy that guides or modifies a heuristic to produce solutions that surpass the quality of those commonly encountered [64–67].

It was made a review based on research on metaheuristic papers in conjunction with discrete-event technology exploring a historical review and applicability of techniques data in the last 5 years, with emphasis on publications and indexing in renowned databases.

In 2014, the application of biological methods and systems together with the design of engineering systems and modern technologies were studied, combining mathematical and metaheuristic algorithms to solve and apply these metaheuristics in medical image segmentation, which plays a fundamental role in medical image analysis for computer-aided diagnosis and classification [68].

In 2015 a hybrid approach to medical image segmentation was studied, where its combined region and border-based information with prior knowledge introduced using deformable registration, using logic and algorithms based on metaheuristics [69].

In 2016 it was studied the qualitative classification of milled rice grains, being developed a machine vision system combined with some metaheuristic classification approaches, where the presented results can be used for the development of an efficient system for fully automated classification and classification of milled rice grains [70].

In 2017, the problem of tracking methods and its management of changes in object appearance, such as changes in illumination, occlusions, scale and pose variation during the tracking process, was studied, and a non-occlusion object tracking method was proposed, coupled with a simple adaptive looking model using a metaheuristic approach, applying the proposal in some videos showing satisfactory results [71].

In 2018, recent rapid advances in medical imaging and automated image analysis have been studied to make significant improvements in our understanding of life and disease processes as well as our ability to provide high-quality healthcare in many processing procedures. Low-level images involve different methods, nature-based or bio-inspired metaheuristic algorithms can find an almost ideal and global solution faster than other traditional in the intelligence of nature [72].

Since the logic for a heuristic method lacks complete solid mathematical knowledge about its behavior, i.e., generally vis-à-vis time consumption, it can be applied to this particular problem, which is designed to have two common features that can produce a satisfactory solution within a reasonable time [64–67].

Metaheuristic derivatives indicate a higher level, composed of several generic heuristics that adapt and are directed to the general optimization of a problem and may contain different heuristic procedures in its structure, being understood as a set of concepts that can be used in the definition of heuristic methods., which can be applied to a wide range of different problems. A strategy that tries efficiently explore the space for workable solutions to this problem, where one has specific knowledge of the problem and can be used as a heuristic to assist in the process [65, 66].

5 Future Trends

Trends in metaheuristics are probabilistic and population-based algorithms of individuals such as the Ant Colony Optimization (ACO) meta-heuristic, which mimics the way real ants find the shortest path between their nest and a source of food. Where tasks are assigned to an ant colony without a central management unit. ACO is a metaheuristic to encounter approximate solutions to severe optimization problems, wherein the artificial algorithm ants build a solution to a combinatorial optimization problem by traversing a fully connected construction graph. Through the cooperative and self-adaptive behavior of artificial ants, which pose a specific problem, emerges an intelligent search system, concurrently collect the necessary information about the structure of a problem, stochastically make their decisions and build a set of solutions. The information required in each decision-making step may include pheromone concentration, problem-specific information, and heuristic function values [17].

ACO can be used for edge detection-oriented digital image processing as one pixel is connected to each pixel that touches one of its edges or corners; Thus, an artificial ant cannot move to a pixel if it is not connected to the pixel where the ant is currently located, so an ant can only move to an adjacent pixel. Where strategies such as artificial ants are distributed over the image, aiming at a final pheromone matrix that reflects edge information, each element in the pheromone matrix corresponds directly to a pixel in the image and indicates whether a pixel is a border or not [18].

As more specific techniques for deep learning architectures are still very popular in image processing, such as Multilayer Perceptron Networks, where Perceptron is a simple algorithm for performing binary classification; that is, it predicts whether the entry belongs to a particular interest category or not, whether it is a cat or not, for example. A Perceptron is a linear classifier; that is, it is an algorithm that classifies the input by separating two categories with a straight line. The input is usually a resource vector $\boldsymbol{x}$ multiplied by weights $\boldsymbol{w}$ and added to a bias b [73].

Just like Convolutional Neural Networks (ConvNets or CNNs) are deep artificial neural networks to classify images, group them by similarity (photo search), and perform object recognition within scenes, provided these algorithms that can identify faces, individuals, street signs, carrots, platypuses and many other aspects of visual data. Just as object recognition has recently been studied, it involves the processing of visual properties through semantic information, considering a computational and cognitive approach to modeling these visual and semantic properties through a neural network [16, 74, 75].

Blob classification and clustering are very intensive computational tasks in terms of computational load. Metaheuristics are also used with dimensionality reduction methods [76–79].

Several new soft computing methods appear every day and can help improve the processing of blood smear images. Some examples:

- World Cup Optimization [80];
- Grey wolf optimization [81];
- Cuckoo search [82];
- Artificial immune systems [83];
- Artificial life (also see digital organism) [84];
- Synergistic fibroblast optimization [85];
- Self-organizing maps [86];
- Bees algorithm [87];
- Whale algorithm [88];
- Firefly algorithm [89]; and
- Harmony search [90].

6 Conclusions

As can be seen throughout this chapter, metaheuristic techniques have had various applications focused on digital medical image processing, as well as demonstrated their use in conjunction with digital image processing techniques, having positive characteristics as the implementation and the exceptional efficiency of these techniques.

Developing methodologies based on techniques based on digital image processing has great potential for solving various problems and challenges present in the medical field, whether related to new diagnostic methods or the cheapening of existing methodologies. As can be seen in the present study, GAs, Fuzzy Logic, PSO, Watershed Transform, Deep Learning techniques as well as Ant Colony Optimization trends have great potential for detecting and counting blood cells, can be seen as steps towards the effective reduction of the complexity of medical examinations, where through technology generate social as well as economic benefits.

References

1. Monteiro ACB, Yuzo I, França RP (2017) Detecting and counting of blood cells using watershed transform: an improved methodology. In: Brazilian technology symposium. Springer, Cham
2. Monteiro ACB, Yuzo I, França RP (2017) An improved and fast methodology for automatic detecting and counting of red and white blood cells using watershed transform. In: VIII Simpósio de Instrumentação e Imagens Médicas (SIIM)/VII Simpósio de Processamento de Sinais da UNICAMP
3. Monteiro ACB et al (2018) Methodology of high accuracy, sensitivity and specificity in the counts of erythrocytes and leukocytes in blood smear images. In: Brazilian technology symposium. Springer, Cham (2018)
4. Monteiro ACB et al (2018) A comparative study between methodologies based on the Hough transform and watershed transform on the blood cell count. Brazilian technology symposium. Springer, Cham
5. Monteiro ACB et al (2019) Medical-laboratory algorithm WTH-MO for segmentation of digital images of blood cells: a new methodology for making hemograms. Int J Simul Syst Sci Technol 20(Suppl 1):19.1–19.5 (5p. 4)
6. Sahastrabuddhe AP, Ajij SD (2016) Blood group detection and RBC, WBC counting: an image processing approach. IJECS 5:10
7. Estrela VV, Saotome O, Loschi HJ, Hemanth DJ, Farfan WS, Aroma RJ, Saravanan C, Grata EGH (2018) Emergency response cyber-physical framework for landslide avoidance with sustainable electronics. Technologies 6:42. https://doi.org/10.3390/technologies6020042
8. Razmjooy N, Estrela VV, Loschi HJ (2019) A study on metaheuristic-based neural networks for image segmentation purposes. In: Data science, pp 25–49
9. Razmjooy N, Estrela VV, Loschi HJ (2019) A survey of potatoes image segmentation based on machine vision. In: Razmjooy N, Estrela VV (eds) Applications of image processing and soft computing systems in agriculture. IGI Global, Hershey, pp 1–38. https://doi.org/10.4018/978-1-5225-8027-0.ch001
10. Estrela VV et al (2019) Health 4.0: applications, management, technologies and review. Med Technol J 2(4):262–276. https://doi.org/10.26415/2572-004X-vol2iss1p262-276
11. Blum C, Roli A (2003) Metaheuristics in combinatorial optimization: overview and conceptual comparison. ACM Comput Surv (CSUR) 35(3):268–308
12. Nesmachnow S (2014) An overview of metaheuristics: accurate and efficient methods for optimisation. Int J Meta 3(4):320–347
13. Gendreau M, Jean-Yves P (2010) Handbook of metaheuristics, vol 2. Springer, New York
14. Kramer O (2017) Genetic algorithm essentials, vol 679. Springer
15. Mirjalili S (2019) Genetic algorithm. In: Evolutionary algorithms and neural networks. Springer, Cham, pp 43–55
16. Hemanth DJ, Estrela VV (2017) Deep learning for image processing applications. In: Advances in parallel computing series, vol 31. IOS Press. ISBN 978-1-61499-821-1 (print). ISBN 978-1-61499-822-8 (online)

17. López-Ibáñez M, Stützle T, Dorigo M (2016) Ant colony optimization: a component-wise overview. In: Handbook of heuristics, pp 1–37
18. Dorigo M, Stützle T (2019) Ant colony optimization: overview and recent advances. In: Handbook of metaheuristics. Springer, Cham, pp 311–351
19. Li Y, Zhan Z, Gong Y, Chen W, Zhang J, Li Y (2015) Differential evolution with an evolution path: a deep evolutionary algorithm. IEEE Trans Cybernet 45:1798–1810
20. Sörensen K, Sevaux M, Glover F (2018) A history of metaheuristics. In: Handbook of heuristics, pp 1–18
21. Dubois G (2018) Modeling and simulation: challenges and best practices for industry. CRC Press (2018).
22. Birkfellner W (2016) Applied medical image processing: a basic course. CRC Press (2016)
23. Robertson S et al (2018) Digital image analysis in breast pathology—from image processing techniques to artificial intelligence. Transl Res 194:19–35
24. Stearns SD, Donald RH (2016) Digital signal processing with examples in MATLAB. CRC Press
25. Nixon M, Aguado A (2019) Feature extraction and image processing for computer vision. Academic Press
26. de Azevedo-Marques PM et al (eds) Medical image analysis and informatics: computer-aided diagnosis and therapy. CRC Press
27. Sebesta RW (2016) Concepts of programming languages. Pearson Education India
28. McAndrew A (2015) A computational introduction to digital image processing. Chapman and Hall/CRC
29. Kothari S, Phan JH, Stokes TH, Wang MD (2013) Pathology imaging informatics for quantitative analysis of whole-slide images. J Am Med Inform Assoc 20(6):1099–1108
30. Fernandes SR, Estrela VV, Saotome O (2014) On improving sub-pixel accuracy by means of B-spline. In: Proceedings of the 2014 IEEE international conference on imaging systems and techniques (IST). https://doi.org/10.1109/IST.2014.6958448
31. Ghaznavi F, Evans A, Madabhushi A, Feldman M (2013) Digital imaging in pathology: whole-slide imaging and beyond. Ann Rev Pathol 8:331–359
32. Goacher E, Randell R, Williams BJ, Treanor D (2017) The diagnostic concordance of whole slide imaging and light microscopy: a systematic review. Arch Pathol Lab Med 141(1):151–161
33. Kaur S, Kaur P (2016) An edge detection technique with image segmentation using ant colony optimization: a review. In: Proceedings of the 2016 online international conference on green engineering and technologies (IC-GET), pp 1–5
34. Tan L, Jean J (2018) Digital signal processing: fundamentals and applications. Academic Press
35. Sucaet Y, Waelput W (2014) Digital pathology. Springer. https://doi.org/10.1007/978-3-319-08780-1
36. Ferrer-Roca O, Marcan F, Vidal M, Ruckhaus E, Fernández-Baíllo R, Santos X, Álvarez-Marquina A, Iglesias E (2011) Grid technology in telepatology and personalised treatment. In: Kldiashvili E (ed) Grid technologies for e-health: applications for telemedicine services and delivery. IGI Global, Hershey, pp 117–128. https://doi.org/10.4018/978-1-61692-010-4.ch006
37. Franca RP, Iano Y, Monteiro ACB, Arthur R, Estrela VV (2019) Betterment proposal to multipath fading channels potential to MIMO systems, In: Iano Y et al (eds) Proceedings of the 4th Brazilian technology symposium (BTSym'18). Smart innovation, systems and technologies, vol 140. Springer. https://doi.org/10.1007/978-3-030-16053-1_11
38. Kriegel H, Kröger P, Zimek A (2009) Clustering high-dimensional data: a survey on subspace clustering, pattern-based clustering, and correlation clustering. TKDD 3:1:1–1:58
39. Dragan D, Ivetic D (2009) Architectures of DICOM based PACS for JPEG2000 medical image streaming. Comput Sci Inf Syst 6:186–203
40. Estrela VV, Herrmann AE (2016) Content-based image retrieval (CBIR) in remote clinical diagnosis and healthcare. In: Cruz-Cunha M, Miranda I, Martinho R, Rijo R (eds) Encyclopedia of e-health and telemedicine. IGI Global, Hershey, pp 495–520. https://doi.org/10.4018/978-1-4666-9978-6.ch039

41. Cruz BF, de Assis JT, Estrela VV, Khelassi, A (2019) A compact SIFT-based strategy for visual information retrieval in large image databases. Med Technol J 3(2):402–412. https://doi.org/10.26415/2572-004X-vol3iss2p402-412
42. Chen L, Papandreou G, Kokkinos I, Murphy K, Yuille AL (2016) DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Trans Pattern Anal Mach Intell 40:834–848
43. Gupta S, Girshick RB, Arbeláez PA, Malik J (2014) Learning rich features from RGB-D images for object detection and segmentation. In: Proceedings of the 2014 ECCV
44. Rabadi G (ed) Heuristics, metaheuristics and approximate methods in planning and scheduling, vol 236. Springer
45. Kurniasih J, Utami E, Raharjo S (2019) Heuristics and metaheuristics approach for query optimization using genetics and memetics algorithm. In: Proceedings of the 2019 1st international conference on cybernetics and intelligent system (ICORIS), vol 1. IEEE, pp 168–172
46. Costin HN, Thomas MD (2018) computational intelligence re-meets medical image processing. Methods Inf Med 57(05/06):270–271
47. da Silva FD, Estrela VV, Matos LJ (2011) Hyperspectral analysis of remotely sensed images. In: Sustainable water management in the tropics and subtropics—and case studies in Brazil, vol 2. University of Kassel. ISBN 978-85-63337-21-4
48. De Silva CW (2018) Intelligent control: fuzzy logic applications. CRC Press
49. De Barros LC, Rodney CB, Weldon AL (2017) Biomathematical modeling in a fuzzy environment. In: A first course in fuzzy logic, fuzzy dynamical systems, and biomathematics. Springer, Berlin, Heidelberg, pp 237–269
50. Osowski S et al (2008) Application of support vector machine and genetic algorithm for improved blood cell recognition. IEEE Trans Instrum Meas 58(7):2159–2168
51. Du K-L, Swamy MNS (2016) Particle swarm optimization. Search and optimization by metaheuristics. Birkhäuser, Cham, pp 153–173
52. de Jesus MA, Estrela VV, Saotome O, Stutz D (2018) Super-resolution via particle swarm optimization variants. In: Hemanth J, Balas V (eds) Biologically rationalized computing techniques for image processing applications. Lecture notes in computational vision and biomechanics, vol 25. Springer. https://doi.org/10.1007/978-3-319-61316-1_14
53. Marini F, Beata W (2015) Particle swarm optimization (PSO). A tutorial. Chem Intell Lab Syst 149:153–165
54. Vale AMPG et al (2014) Automatic segmentation and classification of blood components in microscopic images using a fuzzy approach. Rev Bras Eng Bioméd 30(4):341–354
55. Romero-Zaliz R, Reinoso-Gordo JF (2018) An updated review on watershed algorithms. In: Soft computing for sustainability science. Springer, Cham, pp 235–258
56. Monteiro ST et al (2005) Feature extraction of hyperspectral data for under spilled blood visualization using particle swarm optimization. Int J Bioelectrom 7(1):232–235
57. Monteiro ACB, Yuzo I, França RP (2018) Proposal of a medical algorithm based on the application of digital image processing and visual communication techniques. SET Int J Broadcast Eng 4:9
58. Jordan MI, Mitchell TM (2015) Machine learning: Trends, perspectives, and prospects. Science 349(6245):255–260
59. Goodfellow I, Yoshua B, Aaron C (2016) Deep learning. MIT Press
60. LeCun Y, Yoshua B, Geoffrey H (2015) Deep learning. Nature 521(7553):436–444
61. Tiwari P et al (2018) Detection of subtype blood cells using deep learning. Cogn Syst Res 52:1036–1044
62. Glover F, Cotta C (2019) An overview of meta-analytics: the promise of unifying metaheuristics and analytics. In: Business and consumer analytics: new ideas. Springer, Cham, pp 693–702
63. Datta S, Sandipan R, Davim JP (2019) Optimization techniques: an overview. optimization in industry. Springer, Cham, pp 1–11
64. Cuevas E, Espejo EB, Enríquez AC (2019) Introduction to metaheuristics methods. In: Metaheuristics algorithms in power systems. Springer, Cham, pp 1–8
65. Bhattacharyya S (ed) Hybrid metaheuristics for image analysis. Springer

66. Hussain K et al (2018) Metaheuristic research: a comprehensive survey. Artifi Intell Rev, pp 1–43
67. Fernandez SA et al (2018) Metaheuristics in telecommunication systems: network design, routing, and allocation problems. IEEE Syst J 12(4):3948–3957
68. Sahoo A, Satish C (2014) Meta-heuristic approaches for active contour model based medical image segmentation. Int J Adv Soft Comput Appl 6(2)
69. Mesejo P et al (2015) Biomedical image segmentation using geometric deformable models and metaheuristics. Comput Med Imaging Graph 43:167–178
70. Zareiforoush H et al (2016) Qualitative classification of milled rice grains using computer vision and metaheuristic techniques. J Food Sci Technol 53(1):118–131
71. Sardari F, Moghaddam ME (2017) A hybrid occlusion free object tracking method using particle filter and modified galaxy based search meta-heuristic algorithm. Appl Soft Comput 50:280–299
72. Costin HN, Deserno TM (2018) Computational intelligence re-meets medical image processing. Methods Inf Med 57(05/06):270–271
73. da Silva IN et al (2017) Multilayer perceptron networks. In: Artificial neural networks. Springer, Cham, pp 55–115
74. Vedaldi A, Karel L (2015) Matconvnet: convolutional neural networks for MATLAB. In: Proceedings of the 23rd ACM international conference on multimedia. ACM
75. Razmjooy N, Estrela VV (2019) Applications of image processing and soft computing systems in agriculture. IGI Global. https://doi.org/10.4018/978-1-5225-8027-0
76. Coelho AM, Assis JT, Estrela VV (2009) Error concealment by means of clustered blockwise PCA. In: 2009 picture coding symposium. IEEE, pp 1–4. https://doi.org/10.1109/PCS.2009.5167442
77. Coelho AM, Estrela VV (2012) EM-based mixture models applied to video event detection. In: Principal component analysis—engineering applications. IntechOpen. https://doi.org/10.5772/38129
78. Ravi V, Naveen N, Pandey M (2013) Hybrid classification and regression models via particle swarm optimization auto associative neural network based nonlinear PCA. Int J Hybrid Intell Syst 10:137–149
79. Miranda V, Martins JD, Palma V (2014) Optimizing large scale problems with metaheuristics in a reduced space mapped by autoencoders—application to the wind-hydro coordination. IEEE Trans Power Syst 29:3078–3085
80. Razmjooy N, Ramezani M, Estrela VV (2019) A solution for Dubins path problem with uncertainties using world cup optimization and Chebyshev polynomials. In: Iano Y, Arthur R, Saotome O, Vieira Estrela V, Loschi H. (eds) Proceedings of the 4th Brazilian technology symposium (BTSym'18). Smart innovation, systems and technologies, vol 140. Springer
81. Heidari AA, Pahlavani P (2017) An efficient modified grey wolf optimizer with Lévy flight for optimization tasks. Appl Soft Comput 60:115–134
82. Rajabioun R (2011) Cuckoo optimization algorithm. Appl Soft Comput 11:5508–5518
83. Coello CA, Cortés NC (2005) Solving multiobjective optimization problems using an artificial immune system. Genet Program Evolvable Mach 6:163–190
84. Kanakubo M, Hagiwara M (2007) Speed-up technique for association rule mining based on an artificial life algorithm. In: 2007 IEEE international conference on granular computing (GRC 2007), pp 318–318
85. Dhivyaprabha TT, Subashini P (2017) Performance analysis of synergistic fibroblast optimization (SFO) algorithm. In: 2017 IEEE international conference on current trends in advanced computing (ICCTAC), pp 1–7
86. Majumder A, Behera L, Venkatesh KS (2014) Emotion recognition from geometric facial features using self-organizing map. Pattern Recogn 47:1282–1293
87. Karaboga D, Basturk B (2007) A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. J Global Optim 39:459–471

88. Mirjalili SM, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51–67
89. Tilahun SL, Ong HC (2012) Modified firefly algorithm. J Appl Math, 467631:1–467631:12
90. Mahdavi M, Fesanghary M, Damangir E (2007) An improved harmony search algorithm for solving optimization problems. Appl Math Comput 188:1567–1579

Optimal Bidding Strategy for Power Market Based on Improved World Cup Optimization Algorithm

Navid Razmjooy, Anand Deshpande, Mohsen Khalilpour, Vania V. Estrela, Reinaldo Padilha, and Ana Carolina Borges Monteiro

Abstract Power companies in the world-wide have been restructuring their electric power systems from a vertically integrated entity to a deregulated and open-market environment. In the past, electric utilities usually look for maximizing the social welfare of the system with distributional equity as their main operational criterion. The operating paradigm was based on achieving the least-cost system solution while meeting reliability and security margins. This often resulted in investments in generating capacity operating at very low capacity factors. Decommissioning of this type of generating capacity was a natural outcome when the vertically integrated utilities moved over to deregulated market operations. This paper proposes an optimizing base and load demand relative binding strategy for generating the power pricing of different units in the investigated system. Afterward, the congestion effect in this biding strategy is investigated. The described systems analysis is implemented on 5 and 9 bus systems and the optimizing technique in this issue is a new improved version of the world cup optimization algorithm. Simulation results have been compared

N. Razmjooy (✉)
Department of Electrical and Control Engineering, Tafresh University, 39518 79611 Tafresh, Iran
e-mail: navid.razmjooy@hotmail.com

A. Deshpande
Department of Electronics and Communication Engineering, Angadi Institute of Technology and Management, Belgaum, India
e-mail: deshpande.anandb@gmail.com

M. Khalilpour
Young Researchers Club, Ardabil Branch, Islamic Azad University, Ardabil, Iran
e-mail: m.khalilpour66@gmail.com

V. V. Estrela
Department of Telecommunications, Fluminense Federal University (UFF), Rio de Janeiro, Brazil
e-mail: vania.estrela.phd@ieee.org

R. Padilha · A. C. B. Monteiro
School of Electrical and Computer Engineering (FEEC), University of Campinas—UNICAMP, Av. Albert Einstein—400, Barão Geraldo, Campinas—SP, Brazil
e-mail: padilha@decom.fee.unicamp.com

A. C. B. Monteiro
e-mail: monteiro@decom.fee.unicamp.com

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_7

with the standard world cup optimization algorithm. Finally, examined systems are simulated by using the *Power World* software. Experimental results show that the proposed technique has a good superiority compared with the world cup optimization algorithm for congestion management purposes.

Keywords World cup optimization algorithm · Chaos theory · Power market · Local marginal price · Power flow · Congestion management

1 Introduction

In recent years, many electric utilities worldwide have been forced to change their method of business from vertically integrated functioning to open-market systems. In developing countries, the main issues have been high demand growth associated with inefficient system management and irrational tariff policies, among others [1]. This affects the availability of a capital investment in generation and transmission systems. In such a situation, many countries were forced to restructure their power sectors under the pressure from international funding agencies. On the other hand, in developed countries, the driving force has been performed to the electricity providers to provide the customers with electricity at lower prices and to offer them greater choice in purchasing electricity [2].

In front of the days restructuring, the power grid used to be operated by vertically integrated utilities, that has control over both generation and transmission appliances [3].

There are several methods for congestion management. One of these methods is the capacitance auction. Independent system operator auctions are some of the determined transmission generally in a short time and typically are transmissions which happens congestion to them. In the pool markets, load flow and locational marginal pricing (LMP) have been adopted for congestion management [4].

The Independent System Operator (ISO) is a regulating entity autonomous from the side of electric companies that optimizes the overall system operation. Spot pricing theory is used for economic generation and load dispatch. Under the pool system, locational prices are computed by the marginal cost of optimal power flow solutions [5, 6].

The main idea in this paper is to propose an optimized *Power Transfer Distribution Factors* (PTDF) for optimized designing of the IEEE 5-bus and IEEE 9-bus systems for congestion management purposes.

Here, a newly developed version of the world cup optimization algorithm based on chaos theory has been utilized. This improvement is performed for improving the premature convergence of the algorithm.

The designed algorithm is then utilized for investigation the problem and also to analyze the effect of each unit to the local marginal price. The results of the proposed method are compared with some different state of the art methods to show its efficiency. After optimizing the system, the achieved values are utilized to simulate the system using *Power World* software to check the results [7, 8]. In this study improved version of the world cup optimization (WCO) algorithm is utilized for optimal bidding strategy in the power market. The evaluation is applied before and after congestion management.

2 Problem Formulation

The power load flow (P_{ij}) through the transmission line ($i - j$), is a function of the line reactance (X_{ij}), the voltage magnitude (V_i, V_j) and the phase angle between the sending and receiving end voltages ($\delta_i - \delta_j$) is achieved as the following equation.

$$p_{ij} = \frac{V_i V_j}{x_{ij}} \sin\left(\delta_i - \delta_j\right) \tag{1}$$

The *Transmission Line Relief* (TLR) sensitivity values at all the load buses for the most overloaded transmission line are regarded and used for calculating the essential load curtailment for the alleviation of the transmission congestion. The TLR sensitivity at a bus k for a congested line $i - j$ is S_{ij}^k and is achieved as follows:

$$S_{ij}^k = \frac{\overline{\Delta P_{ij}}}{\Delta P_k} \tag{2}$$

The excess power flow on transmission line $i - j$ is obtained by:

$$\overline{\Delta P_{ij}} = P_{ij} - \overline{P_{ij}} \tag{3}$$

where, P_{ij} is the actual power flow through transmission line $i - j$ and $\overline{P_{ij}}$ is the flow limit of transmission line $i - j$ [9, 10]. The new load P_k^{new} at the bus k can be achieved as follows:

$$P_k^{new} = P_k - \frac{S_{ij}^k}{\sum_{i=1}^N S_{ij}^l} \overline{\Delta P_{ij}} \tag{4}$$

where, P_k^{new} is the load after curtailment at bus k, P_k is the load before curtailment at bus k, S_{ij}^k is the power flow sensitivity on line $i - j$ due to load change at bus k, and N describes the total number of load buses.

3 Improved World Cup Optimization Algorithm

3.1 World Cup Optimization Algorithm

Optimization is the art of finding the best acceptable solution for a given minimization or maximization problem. The algorithm should be considered all the system dynamics and its constraints. Meta-heuristic algorithms are some kinds of optimization algorithms that are inspired by nature, physics, or human's social reactions and are used to solve several optimization problems. In most cases, these algorithms can be adopted in combination with other algorithms to reach the optimal solution or exit from the local optimum.

Some of these algorithms are genetic algorithm [11–15], particle swarm optimization [16, 17], quantum invasive weed optimization [18], world cup optimization algorithm [19], and variance reduction of Gaussian distribution [20] which have been designed for solving different complicated problems.

In recent years, Razmjooy et al. introduced a new meta-heuristic technique inspired by FIFA world cup competitions. They called the method world cup optimization algorithm (WCO). The method is then utilized in several methods [19, 21–25] to and the results showed the system good efficiency.

The competition is mathematically modeled to provide an optimization tool for obtaining the global optimum solution. The main idea in WCO is to challenge all the teams with each other to find the best and strongest team. Each team that achieves the best value will be the best solution.

WCO algorithm, like most of the other meta-heuristic algorithms, has two important parts of exploitation (like team's ranking and playoff) and exploration (like random surprising teams which there is no expectation to have a good solution).

In the standard model, WCO starts by a random number of populations which are called teams to reach the global optimal solution.

By considering N, the number of variable dimensions (N_{var}) and M number of continents for an optimization problem,

$$Continent = \begin{bmatrix} x_{c1,1} & x_{c2,1} & \cdots & x_{cM,1} \\ x_{c1,2} & x_{c2,2} & \cdots & x_{cM,2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{c1,N_{\text{var}}} & x_{c2,N_{\text{var}}} & \cdots & x_{cM,N_{\text{var}}} \end{bmatrix} \tag{5}$$

where, $x_{i,j}$ shows the ith team of the jth country.

The scoring of the teams for ranking has been performed by the score function (f_r) as follows:

$$f_r(continent_i) = f_r(x_{ci,1}, x_{ci,2}, \ldots, x_{i,N_{\text{var}}}) \tag{6}$$

An important parameter in the WCO algorithm is “Rank”. Rank has a great impact on the solution. After obtaining the rank scores, the first n potent teams are included in the first seed, the other n weaker teams are included in the second seed, and the others have been classified as the first and second seeds hierarchically.

The rank scoring in WCO is modeled as follows:

$$Rank = \frac{(\beta \times \sigma + \overline{X})}{2} \tag{7}$$

where, β is a parameter for increasing or decreasing the impact of standard deviation in the interval [0, 1] and,

$$\overline{X} = \frac{1}{n} \sum_{i=1}^{n} X_i \tag{8}$$

$$\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (X_i - \overline{X})^2} \tag{9}$$

where, n, $\overline{X}$ and σ are the number population, the mean value and the standard deviation of the continent X, respectively.

In the WCO algorithm, after raising the first seed directly to the second stage of competitions, the challenge has been started. The competition commences among each team separately in their seeds and the winner of the competition improves its score (rank) and move to the next stage of the competition.

Another important parameter in the WCO algorithm is the “Play-Off”. After getting the two strong teams in each team to the next level of competition, the rest teams should be discarded. In the meantime, the third team of each seed can have a second chance to return to the competition (Play-Off).

For updating the solution for the next iteration (the next competition), it uses previous information on the competition scores and the team’s ranking. To do so, WCO uses two-part vector:

$$Pop = [X_{Best}, X_{Rand}] \tag{10}$$

where, Pop is an $N \times M$ matrix which describes the new population, X_{Rand} describes a random value in the problem interval constraints, and X_{Best} is:

$$\frac{1}{2} \times ac \times (Ub - Lb) < X_{Best} < \frac{1}{2} \times ac \times (Ub + Lb) \tag{11}$$

where, *Ub* and *Lb* describe the higher and the lower bounds of the problem constraints and *ac* is the accuracy parameter between *Lb* and *Ub*.

3.2 Improved World Cup Optimization Algorithm

Chaos theory is a concept that tells us all random values have a determined relation with each other. There are some complicated dynamical and nonlinear systems whit chaos nature with randomly and unpredictable behavior.

The main idea behind this theory is to analysis the highly sensitive dynamic systems which have been affected by even very small variations, i.e. a small change in the condition, make big changes in the system behavior.

By considering this definition, a large diversity can be generated by the population for the global optimum improvement and for escaping from the local optimum [26, 27]. A simple definition of chaos behavior can be described as follows:

$$\begin{aligned} &CM_{i+1}^{j} = f(CM_i^j) \\ &j = 1, 2, \ldots, k \end{aligned} \tag{12}$$

where, k is the map dimension and $f(CM_i^j)$ is the chaotic model generator function.

The main advantage of using chaos theory in the WCO algorithm is that can escape from sticking at the local optimum point. It can also prevent premature convergence. In the proposed IWCO algorithm, the parameter X_{Rand} is modeled based on the Singer mechanism [26, 27]. To implement this mechanism, the unknown scale factor (γ) is converted into regular formulated value as follows:

$$\begin{aligned} X_{Rand,\,k+1} = 1.07 \times (7.9 \times X_{Rand,\,k} - 23.3 \times X_{Rand,\,k}^2 + 28.7 \times X_{Rand,\,k}^3 \\ - 13.3 \times X_{Rand,\,k}^4) \end{aligned} \tag{13}$$

The flowchart of the presented IWCO algorithm is shown in Fig. 1.

4 Bidding Strategy

Bidding strategy is described by suggesting transactions in the market [28]. Energy selling prices by the power generation units and energy purchasing prices by the clients are recommended to the power market. The whole market zones try to maximize the social welfare index. Bidding strategy with no congestion can be presented as below:

$$Min\ f_k = \min\left(\sum_{j=1}^{NG} C_j(p_j) - \sum_{i=1}^{NL} B_i(d_i)\right) \tag{14}$$

where

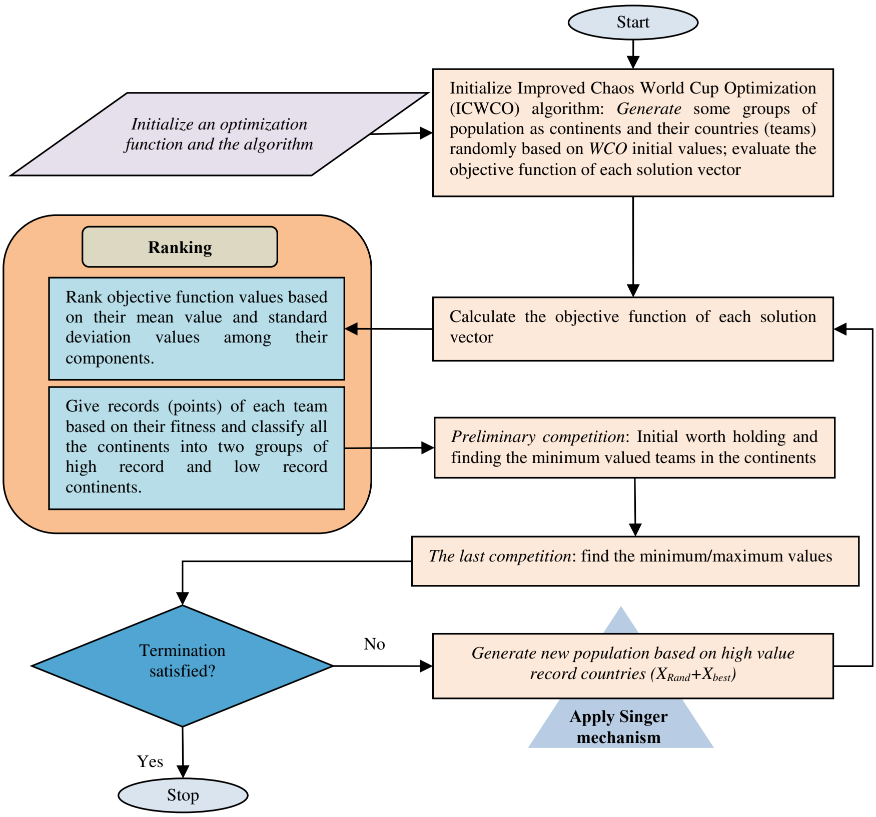


Fig. 1 The flowchart diagram of the proposed IWCO algorithm

$$B_i(d_i) = b_i d_i^t - 1/2 d_i^t p_{jj} \tag{15}$$

$$C_j(j) = b_i p_j^t - 1/2 d_j^t p_{jj} \tag{16}$$

$$subject.to \rightarrow \sum_{j=1}^{NG} p_j = \sum_{i=1}^{NL} d_i \tag{17}$$

$$p_j^{\min} \le p_j \le p_j^{\max}$$

where, i describes customers load index, j is generators index, NL is the number of consumption loads, NG is the number of generators, B_i is the ith customer's profit function, P_j is the delivery power of jth unit, C_j is the jth generator's cost function and d_j is the quantity of consumption power in the jth unit.

5 Case Studies

5.1 5 Bus System Data

In the following, the simulation results of the proposed method are applied to two different case studies to show the system capability.

Since the electricity market has been deregulated, the participants have several choices to develop their standing in the market. A set of different bidding strategies may be adopted by the participants in order to maximize their profit. In this study, by neglecting the losses in the load flow, the following purpose is considered:

$$Min \Delta p^t \omega \tag{18}$$

$$|p_{ij}| \leq |p_{ij}^{\max}| \tag{19}$$

$$\sum_{i \in NG} \Delta p_i = 0 \tag{20}$$

$$p_j^{\min} \leq p_j \leq p_j^{\max} \tag{21}$$

where, ω is a weight matrix (in this study, ω is advised a 3×3 uniform identity matrix), and $\Delta p = [\Delta p_1 \ldots \Delta p_n]$.

The required power of consumption loads vector is described based on loads of power differences consumption.

In this section, 5 bus systems by 3 generator units, 2 customers (loads), and 6 transmission lines is analyzed. Table 1 illustrates the Cost functions and profits to the presented system are described in. For a suitable analysis of the costs, two different states are demonstrated to the system.

For analyzing methods, the transmission lines are first assumed without congestion and the relative analysis on the 5 bus system has been implemented. Afterward, optimized congestion management has been applied and the analysis executed to

Table 1 Related data for cost functions in 5 bus system

Generators	Cost function	p_j^{min}
Genco.1.	$0.003(p_1)^2 + 7.2p_1 + 640$	0 (MW)
Genco.2.	$0.002(p_2)^2 + 6.3p_2 + 360$	0 (MW)
Genco.3.	$0.004(p_3)^2 + 6.8p_3 + 120$	0 (MW)
Loads	Profit function	Peak load
Customer1.	$110d_1 - 0.18(d_1)^2$	150 (MW)
Customer2.	$120d_2 - 0.16(d_2)^2$	60 (MW)

the system. Bidding values and obtained cost using IWCO and the standard WCO algorithms are illustrated in Table 2.

Table 2 shows higher social welfare for the proposed improved WCO algorithm than the standard WCO algorithm.

Table 3 shows the line flows before and after congestion management on bus 5. From Table 3, inline 5, the line flow is about 6 MW more than the limited value which is decreased to 45 MW after congestion management.

From Table 3 it is observed that after bidding strategy, line 5 has some overloads. Simulation results show that in order to remove the congestion, the output power of generator 3 is decreased. In brief, generators in bus 2 for consumption load ensuring are faced to genesis growing. Also, the acquired social welfare index in the IWCO is desirable rather than the standard WCO. Notice that transmission lines losses are also calculated in this analysis. Table 4 shows the simulation results.

Table 2 Social welfare index before congestion management

Algorithm	IWCO		WCO	
	Output (Gen (MW))	Cost ($)	Output (Gen (MW))	Cost ($)
Generators				
Gen.1	92.3274	1330.3303	115.6827	1513.0573
Gen.2	64.8405	776.9037	43.9748	640.9088
Gen.3	52.831	490.4152	50.340	472.4844
Total cost for generators	210 (MW)	2597.6492 ($)	210 (MW)	2626.4505 ($)
Customers				
Cust.1	150 (MW)	16,095	150 (MW)	16,095
Cust.2	60 (MW)	6624	60 (MW)	6624
Total benefit for customers	210 (MW)	22,719 ($)	210 (MW)	22,719 ($)
Social welfare ($)	20,121.35		20,092.54	

Table 3 Line flow before and after congestion management for the 5 bus system

# Line	Line flow after bidding strategy (MW)	Line flow limit (MW)	Line flow after congestion management (MW)
1	46.6	50	43.4
2	24.1	150	13.3
3	19.9	50	17
4	21.6	100	20.7
5	55.9	50	44.9
6	128.4	150	129.3

Table 4 Simulation results of the social welfare index after congestion management for 5 bus system

Algorithm	IWCO			WCO		
Generator No.	Gen.1	Gen.2	Gen.3	Gen.1	Gen.2	Gen.3
Output generators without congestion management (MW)	92.3274	64.8405	52.831	115.6827	43.9748	50.340
Output generators with congestion management (MW)	30.5743	90.8562	80.1937	35.5470	95.4652	85.2842
Curtailed output (MW)	−61.7531	26.01	27.3627	−80.357	51.4904	34.9392
Total cost for generators ($)	2502.88			2608.40		
Total benefit for customers ($)	22,719			22,719		
Social welfare ($)	20,216.12			20,110.6		

5.2 9 Bus System Data

Since the electricity market has been deregulated, the participants have a variety of choices to improve their standing in the market. A set of different bidding strategies may be adopted by the participants in order to maximize their profit. In this system assessment, by spot the losses in the load flow, the purpose is:

$$\min F = \frac{1}{F_L \times F_G} \tag{22}$$

$$F_L = \prod_{i=1}^{k} F_{lineflow} \tag{23}$$

$$F_G = \prod_{q=1}^{p} F_{generationpower} \tag{24}$$

where, k is the number of transmission line and p is the number of generators in the 9 bus system.

The requirement power of consumption loads vector is described by consumption loads power differences.

In this case study, the 9 bus system structure by 3 generator units, 3 customers (loads), and 6 transmission lines is analyzed. Cost functions and profits to the presented system are illustrated in Table 5.

For a suitable analysis, at first, transmission lines before congestion assumed and the relative analysis on the 9 bus system implemented. Afterward, congestion also

Table 5 Related data for 9 bus system

Generators	Cost function	p_j^{min}	p_j^{max}
Genco.1.	$0.001562(p_1)^2 + 7.92p_1 + 560$	0 (MW)	200 (MW)
Genco.2.	$0.00194(p_2)^2 + 8.5p_2 + 310$	0 (MW)	150 (MW)
Genco.3.	$0.00482(p_3)^2 + 7.97p_3 + 78$	0 (MW)	150 (MW)
Loads	Profit function	Peak load	
Customer1.	$100d_1 - 0.175(d_1)^2$	125 (MW)	
Customer2.	$110d_2 - 0.15(d_2)^2$	100 (MW)	
Customer3.	$90d_3 - 0.14(d_3)^2$	90 (MW)	

implemented and the analysis executed on the system again. Bid values and attaining cost using WCO and IWCO are described in Table 6 and the results show a higher performance for the IWCO algorithm toward standard WCO.

Table 7 illustrates the line flow before and after congestion management at 9 bus system. From Table 7, inline 3, the line flow is about 50 MW which is more than the limited value. The line flow is then decreased to 47.6 MW after congestion management.

From Table 7 it is observed that after bidding strategy, line 3 has found some overloads. Simulation results show that in order to remove the congestion, the output of generator 2 is decreased. In brief generators in bus 3 for consumption load ensuring are faced to genesis growing. Also, the acquired social welfare index in the IWCO

Table 6 Simulation results of the social welfare index after congestion management for 9 bus system

Algorithm	IWCO		WCO	
	Output (Gen (MW))	Cost ($)	Output (Gen (MW))	Cost ($)
Generators				
Gen.1	151.4101	1794.97	136.1708	1667.4361
Gen.2	110.1704	1198.38	135.9245	1412.85
Gen.3	88.4195	820.38	77.9047	728.15
Total cost for generators	350	2734.73	350	3808.43
Customers				
Cust.1	125 (MW)	9765	125 (MW)	9765
Cust.2	100 (MW)	6000	100 (MW)	6000
Cust.3	90 (MW)	6966	90 (MW)	6966
Total benefit for customers	315 (MW)	22,731	315 (MW)	22,731
Social welfare ($)	19,996.27		18,922.57	

Table 7 Line flow before and after congestion management for the 9 bus system

# Line	Line flow after bidding strategy	Line flow limit	Line flow after congestion management (MW)
1	105	150	123.6
2	17.7	50	19.4
3	52.9	50	47.6
4	146.3	150	135.9
5	204.2	250	197.1
6	48.2	150	42

Table 8 Simulation results of the social welfare index after congestion management for 9 bus system

Algorithm	IWCO		WCO	
Generator No.	Gen.1	Gen.2	Generator No.	Gen.1
Output generators without congestion management (MW)	151.4101	110.1704	88.4195	136.1708
Output generators with congestion management (MW)	166.372	50.25	150	177.43
Curtailed output (MW)	14.9619	−59.9204	61.5805	41.2592
Total cost for generators ($)	2615.36		2783.54	
Total benefit for customers ($)	22,731		22,731	
Social welfare ($)	20,115.64		19,947.46	

is desirable rather than the standard WCO. Notice that transmission lines losses are also calculated in this analysis. Table 8 shows the simulation results for the 9 bus system.

6 Simulation Results

6.1 5 Bus System

Simulation results of 5 bus system using *Power World* software shows that the line transmission 5 (the line that is between bus 2 and bus 4) has overload which by declining the value of generated power on generator 3 (to 65 MW) and increasing the generated power of generator 2, congestion of line 5 (using the sensitivity decreasing method) is decreased (rather than the flow power of transmission lines). Figures 2 and 3 show the results.

As you can see in Fig. 3, by increasing the output power of generator 2 into 100 MW congestion in the transmission line (the line which is between bus 2 and

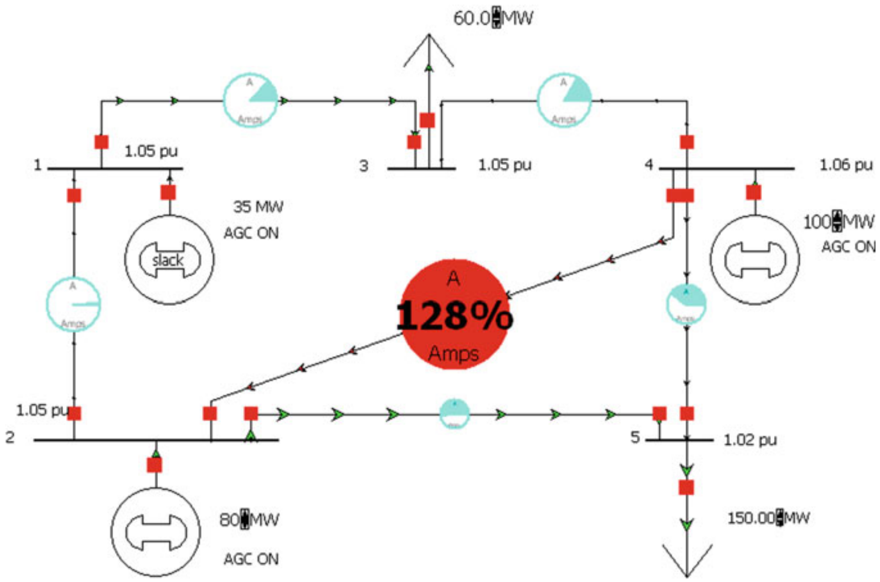


Fig. 2 Mimic diagram of 5 bus system and power flow before congestion management-Power World software

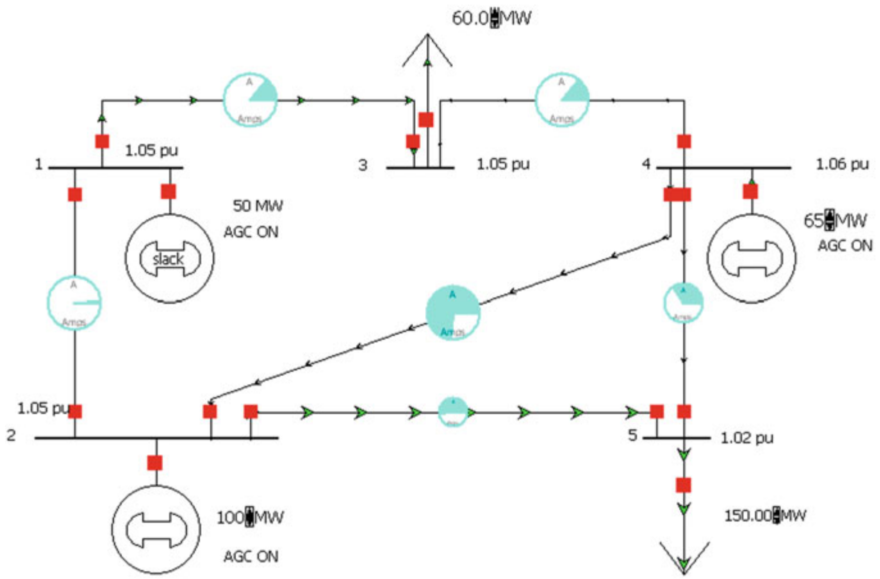


Fig. 3 Mimic diagram of 5 bus system and power flow after congestion management-Power World software

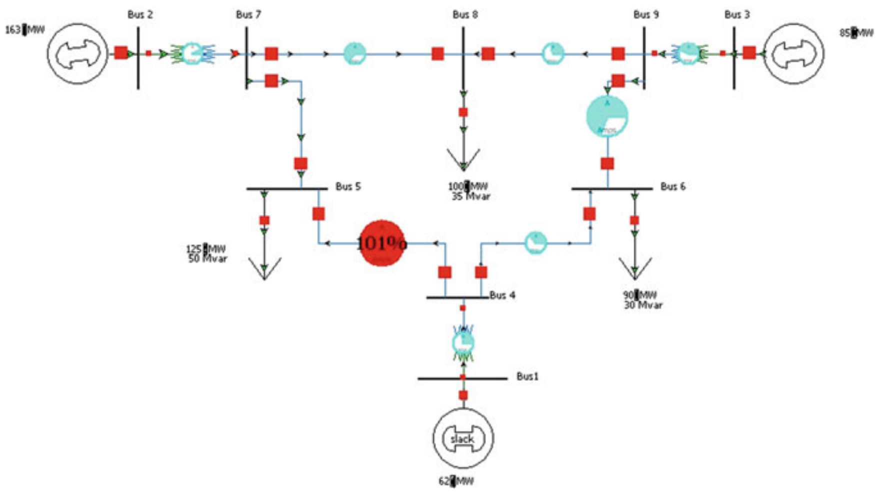


Fig. 4 Mimic diagram of 9 bus system and power flow before congestion management-Power World software

bus 4) is decreased. And the output power value on generator 3 is raised for the consumption load security.

6.2 9 Bus System

Simulation results of 9 bus system using *Power World* software shows that the line transmission 3 (the line that is between bus 4 and bus 5) has overload which by declining the value of generated power on generator 2 (to 153 MW) and increasing the generated power of generator 3, congestion of line 3 (using the sensitivity decreasing method) is decreased (rather than the flow power of transmission lines). Figures 4 and 5 show the analysis.

As can be seen in Fig. 4, by increasing the output power of generator 3 of 85–105 MW congestion in the transmission line (the line which is between bus 4 and bus 5) is decreased and the output power value on generator 3 is raised for the consumption load security.

7 Conclusions

Depending on the structure and objectives of the electricity market, different congestion management methods are put into practice. Effective congestion management will help mitigate the effects of market power in electricity markets. In this work,

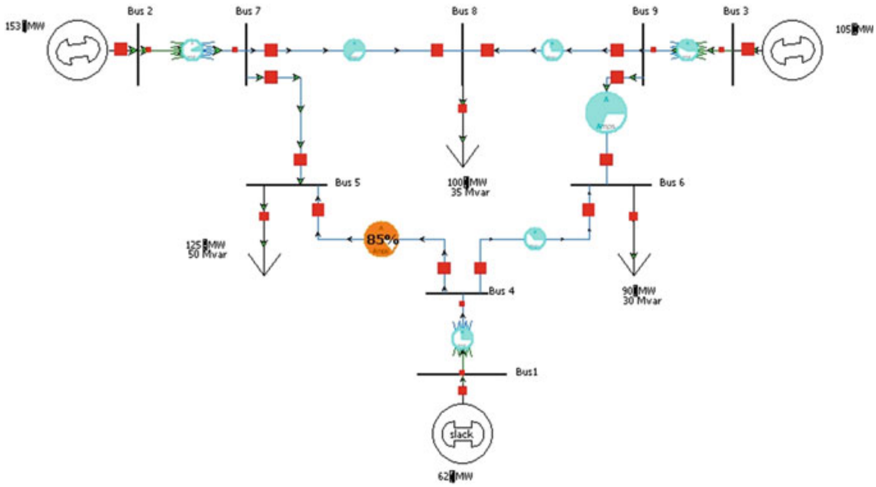


Fig. 5 Mimic diagram of 9 bus system and power flow after congestion management-Power World software

the application of the Invasive Weed Optimization algorithm (IWCO) is presented to congestion management in the bidding strategy; experimental results are compared by the Genetic Algorithm (WCO). It is considered that in IWCO, the output power of generators and also social welfare indexes have more performance than the WCO. These two algorithms are implemented on 5 and 9 bus systems and transmission line losses are envisaged. With regard to more boundaries and to problems of power flow implementation by Gauss–Seidel and Newton–Raphson methods, power world software is used. The OPF in *Power World* simulator provides the ability to optimally dispatch the generation in an area or group of areas while enforcing the transmission line limits. Final results show that using IWO in power flow systems and for congestion management purposes is a proper technique.

References

1. Khanabadi M, Fu Y, Liu C (2018) Decentralized transmission line switching for congestion management of interconnected power systems. IEEE Trans Power Syst 33:5902–5912
2. Jabir H, Teh J, Ishak D, Abunima H (2018) Impacts of demand-side management on electrical power systems: a review. Energies 11:1050
3. Boroojeni KG, Amini MH, Iyengar S, Rahmani M, Pardalos PM (2017) An economic dispatch algorithm for congestion management of smart power networks. Energy Syst 8:643–667
4. Deng L, Li Z, Sun H, Guo Q, Xu Y, Chen R et al (2019) Generalized locational marginal pricing in a heat-and-electricity-integrated market. IEEE Trans Smart Grid
5. Madhankumar S, Karthikeyan SP, Manmohan A, Prasad A (2018) Impact of generator/demand constraints on market clearing price under pool market with elastic demand. In: 2018 national power engineering conference (NPEC), pp 1–6

6. Vakesan K, Krishnamoorthy V (2018) Application of artificial neural networks for short term price forecasting in deregulated power market. Int J Control Autom 11:53–62
7. Wang Y, Cui X, Wang C, Ning X (2019) GIC calculation of Gansu power grid based on Power World. J Phys Conf Ser 022110
8. Monisha B, Balamurugan K (2018) Establishing contingency analysis with FACTS devices using power world simulator. Trends Renew Energy 4:114–131
9. Ma F, Luo X, Litvinov E (2016) Cloud computing for power system simulations at ISO New England—experiences and challenges. IEEE Trans Smart Grid 7:2596–2603
10. Wu M, Huang W, Zhang FQ, Luo X, Maslennikov S, Litvinov E (2017) Power plant model verification at ISO New England. In: 2017 IEEE power & energy society general meeting, pp 1–5
11. Aghdam HN, Ghadimi N, Ataei A (2011) Adjusting PID controllers coefficients to control fuel cell using genetic algorithm. In: 2011 10th international conference on environment and electrical engineering, pp 1–5
12. Ghadimi N (2012) Genetically tuning of lead-lag controller in order to control of fuel cell voltage. Sci Res Essays 7:3695–3701
13. Holland JH (1992) Genetic algorithms. Sci Am 267:66–73
14. Mirjalili S (2019) Genetic algorithm. In: Evolutionary algorithms and neural networks. Springer, pp 43–55
15. Mousavi BS, Soleymani F (2014) Semantic image classification by genetic algorithm using optimised fuzzy system based on Zernike moments. SIViP 8:831–842
16. Moallem P, Razmjooy N (2012) Optimal threshold computing in automatic image thresholding using adaptive particle swarm optimization. J Appl Res Technol 10:703–712
17. Razmjooy N, Ramezani M (2016) Training wavelet neural networks using hybrid particle swarm optimization and gravitational search algorithm for system identification
18. Razmjooy N, Ramezani M (2014) An improved quantum evolutionary algorithm based on invasive weed optimization. Indian J Sci Res 4:413–422
19. Razmjooy N, Khalilpour M, Ramezani M (2016) A new meta-heuristic optimization algorithm inspired by FIFA world cup competitions: theory and its application in PID designing for AVR system. J Control Autom Electr Syst 27:419–440
20. Namadchian A, Ramezani M, Razmjooy N (2016) A new meta-heuristic algorithm for optimization based on variance reduction of Gaussian distribution. Majlesi J Electr Eng 10:49
21. Bandaghiri PS, Moradi N, Tehrani SS (2016) Optimal tuning of PID controller parameters for speed control of DC motor based on world cup optimization algorithm. Parameters 1:2
22. Razmjooy N, Madadi A, Ramezani M (2017) Robust control of power system stabilizer using world cup optimization algorithm. Int J Inf Secur Syst Manag 5:7
23. Razmjooy N, Shahrezaee M (2018) Solving ordinary differential equations using world cup optimization algorithm
24. Razmjooy N, Sheykhahmad FR, Ghadimi N (2018) A hybrid neural network–world cup optimization algorithm for melanoma detection. Open Med 13:9–16
25. Shahrezaee M (2017) Image segmentation based on world cup optimization algorithm. Majlesi J Electr Eng 11
26. Yang D, Li G, Cheng G (2007) On the efficiency of chaos optimization algorithms for global optimization. Chaos Solitons Fractals 34:1366–1375
27. Rim C, Piao S, Li G, Pak U (2018) A niching chaos optimization algorithm for multimodal optimization. Soft Comput 22:621–633
28. Niu H, Baldick R, Zhu G (2005) Supply function equilibrium bidding strategies with fixed forward contracts. IEEE Trans Power Syst 20:1859–1867

Speed Control of a DC Motor Using PID Controller Based on Improved Whale Optimization Algorithm

Navid Razmjooy, Zahra Vahedi, Vania V. Estrela, Reinaldo Padilha, and Ana Carolina Borges Monteiro

Abstract In this paper, a new optimized method is introduced for the optimal control of a DC motor based on a proportional-integral-derivative (PID) controller. In this study, an improved version of the whale optimization algorithm has been adopted for optimal selection of the PID controller parameters for optimal control of the DC motor speed along with minimum settling time. Unlike the other control algorithms, the PID controller can give more accurate and stable control by tuning the process outputs based on the history and rate of change of the error signal. The proposed approach has a premier specification, including easy application, stable convergence characteristics and high-efficiency computational performances. The DC motor designing by optimized PID controller is modeled based on the MATLAB platform. The results of the proposed method are compared with the standard whale optimization algorithm to show the proposed algorithm's efficiency. Final results show that the proposed approach is better in improving the speed loop response stability, the steady-state error is decreased, and the disturbances do not affect the performances of driving motor with no overtaking.

Keywords Optimal control · DC motor · PID controller · Whale optimization algorithm · Improved

N. Razmjooy (✉)
Department of Electrical and Control Engineering, Tafresh University, 3951879611 Tafresh, Iran
e-mail: navid.razmjooy@hotmail.com

Z. Vahedi
Department of Electrical Engineering, Ardabil Branch, Islamic Azad University, Ardabil, Iran
e-mail: zahra23vahedi@gmail.com

V. V. Estrela
Department of Telecommunications, Fluminense Federal University (UFF), Niterói, Brazil
e-mail: vania.estrela.phd@ieee.org

R. Padilha · A. C. B. Monteiro
Department of Biosystems Engineering, Faculty of Agriculture Technology and Natural Resources, Moheghege Ardebili University, Ardabil, Ardabil, Iran
e-mail: padilha@decom.fee.unicamp.com

A. C. B. Monteiro
e-mail: monteiro@decom.fee.unicamp.com

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_8

1 Introduction

One of the principal advantages of a DC motor is the high start torque characteristics, high response performance and easier to be linear control [1]. Typically, the speed of a DC motor can be tuned to a great extent so as to provide easy control and high performance [2]. There are several methods for controlling the DC motor speed at its executing various tasks like PID Controller [3], Fuzzy Logic Controller [4], or the combination between them like PID-Particle Swarm Optimization [5], the optimal Fuzzy Logic controller using the different strategy [6], etc. [7].

PID controllers are widely used in industrial plants due to their simplicity and robustness. With appending some zeros to a closed-loop transfer function using a differential controller, the transient response will be improved and with appending some poles to it using an integral controller, the steady-state error will be decreased [8]. Selecting the proper tuning parameters is essential to have a good performance of PID controllers. Several methods have been developed to determine the PID controller parameters for single input single output (SISO) systems. Ziegler-Nichols (Z-N) [9], the Cohen-Coon method [10], are some well-known examples of these methods. Ziegler and Nichols utilized transient response characteristics of given plant rules for adjusting the PID controllers [11].

Jin et al. [12] proposed a method based on the model reduction method for designing PID controllers. Tumari et al. [13] presented a model-free approach for optimizing the parameters of a controller. Leva et al. [14] described several PID controller methods for industrial control systems. Adjusting of PID is performed by some expert humans empirically which is often an expensive and difficult activity. Evolutionary Algorithms like GA and PSO have demonstrated their superiority in achieving better results by improving the characteristics and efficiency of the steady-state. In this study, an improved version of the whale optimization algorithm is proposed for the optimal design of the PID controller in a DC motor. A comparison is then performed between the proposed method and the standard whale optimization algorithm to show the superiority of the proposed method. One of the most important factors on DC motors stability is their output speed.

Whale optimization algorithm (WOA) is a new optimization technique that belongs to this category of the so-called nature-inspired algorithm and is based on the whales hunting process [15].

In the whale optimization algorithm, the fitness function of a given optimization problem is depended on the whales' hunting process to achieve effective optimal solutions. The achieved results from the whale optimization algorithm are finally compared with the standard whale optimization algorithm. Experimental results show that for this purpose, the WOA has higher performance.

2 Model of Brushed DC Motor

Typically, a DC motor consists of a *stator*, a *rotor,* and a *commutator*. The stator is the cover of the motor and contains a magnet, bearing, etc. The rotor is the movable component of the motor and comprises a coil of wire through which current flows. The wire coil through the rotor connects to the commutator and captures current through brushes. The commutator warranties that the current flow in the appropriate direction while the rotor turns. DC motor procreates torque directly from DC power covered to the motor by handling internal commutation, static permanent or electromagnets, and rotating electrical magnets. Profits of a brushed DC motor include high reliability, low initial cost, and easy motor speed control. Figure 1 shows the equivalent model of the brushed DC motor with a PID controller.

The transfer function of a PID controller is presented as follows:

$$PID = K_p + \frac{K_i}{s} + K_d s \tag{1}$$

where, K_p is the proportional tuning constant, K_i is the integral adjusting constant, and K_d is the derivative tuning constant. The process of specifying the PID controller parameters K_p, K_i, and K_d to get high and consistent performance characteristics is known as controller adjusting. A low-pass filter (LPF) is also applied to noise elimination. The real model of the DC motor for Simulink is shown in Fig. 2.

In this paper, an efficient and fast adjusting method based on a new improved version of the whale optimization algorithm is proposed to find the optimal parameters of the PID controller so that the desired system features are convinced. To illustrate the presented method efficiency, the step responses of the closed-loop

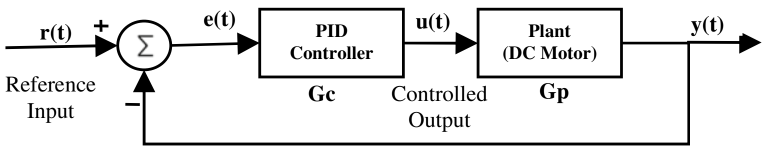


Fig. 1 Equivalent circuit of DC motor with PID Controller

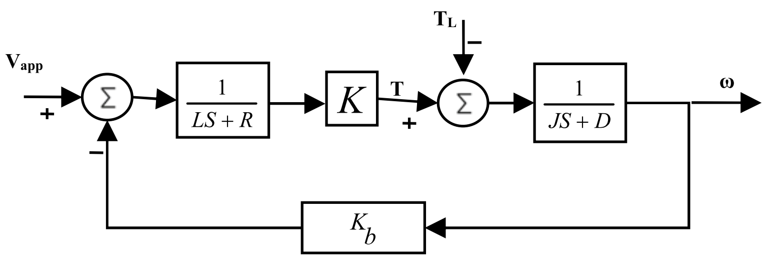


Fig. 2 Model of the DC Motor in Simulink

Table 1 Parameters of the motor

0.4 Ω	Ra
2.7 H	La
0.0004 kg.m^2	J
0.0022 N m s/rad	B
15e−3 kg m/A	K
0.05 V s	Kb

system are compared with the standard whale optimization algorithm based PID. The parameters of the motor used for simulation are presented in Table1.

3 Improved Whale Optimization Algorithm

3.1 The Standard Whale Optimization Algorithm

Humpback whales are some kinds of the largest whales in the world. An adult is about the size of a school bus. Small fish groups are the favorite food for them. The most interesting thing about humpback whales is their particular way of hunting. This exploratory behavior is known as the bubble feeding method. Humpback whales prefer to hunt bunches of small creatures or fish near the water's surface. It has been observed that this exploration is accomplished by generating index bubbles along a circle or paths [16–18].

The WOA is one of the nature-inspired optimization algorithms that is inspired by the bubble net hunting process of the humpback whales and can be used in different optimization problems [19–21].

The algorithm starts with a random vector of variables as the whale's population to find the global solution for the optimization problem. The bubble-net feeding process of the humpback whale is A mathematically modeled as follows:

$$Z(t+1) = \begin{cases} Z^*(t) - AD, & p < 0.5 \\ D' e^{bl} \cos(2\pi t) + Z^*(t), & p \geq 0.5 \end{cases} \tag{2}$$

$$D' = \left| C Z^*(t) - Z(t) \right| \tag{3}$$

$$A = 2ar - a \tag{4}$$

$$C = 2r \tag{5}$$

where, l describes a random variable in the interval [−1, 1], a is a decent integer from 2 to 0 linearly over the iteration, r and p describe random constants in the interval [0, 1], b defines the logarithmic shape of the spiral motion, t is the current iteration, and D' describes the distance of ith whale from the best solution.

Here, the convergence of the method will be guaranteed if $|Z| > 1$. the algorithm exploration is an improvement by the following formula:

$$D' = |CZ_{rand}(t) - Z(t)| \tag{6}$$

$$Z(t+1) = \begin{cases} Z_{rand}(t) - AD, & p < 0.5 \\ D'e^{bl}\cos(2\pi t) + Z_{rand}(t), & p \geq 0.5 \end{cases} \tag{7}$$

The main idea of using the whale optimization algorithm is that although it is a new optimization algorithm, it has been used for different applications due to its good exploration capability. One problem of the whale optimization algorithm is its premature convergence.

3.2 Improved Whale Optimization Algorithm (IWOA)

In this study, chaos theory has been for improving the system efficiency in terms of convergence. This conception has various applications in science such as mathematics and physics. Its simple meaning is rooted in human early perceptions of the universe. In chaos theory, complex systems have a purely turbulent appearance and, as a result, appear irregular and random, while they may be subordinate to a given process with a specific mathematical formula [22, 23]. A simple formulation for the chaos behavior is illustrated below:

$$CM_{i+1}^{j} = f\left(CM_i^j\right), \quad j = 1, 2, \ldots, k \tag{8}$$

where, k describes the map dimension and $f\left(CM_i^j\right)$ represents the chaotic model generator function. Using chaos behavior can improve the system convergence and speed which improves the population diversity to escape from the local optimum trap [22, 23]. In this subsection, an improved version of the WOA is utilized based on the Singer mechanism [24–26]. To implement this mechanism, the unknown scale factor (γ) is converted into regular formulated value as follows:

$$X_{Rand,k}^{i+1} = 1.07\Big(7.9 \times X_{Rand,k}^{i} - 23.3 \times \left(X_{Rand,k}^{i}\right)^2 + 28.7 \times \left(X_{Rand,k}^{i}\right)^3 - 13.3 \times \left(X_{Rand,k}^{i}\right)^4\Big) \tag{9}$$

The flowchart of the presented IWOA is shown in Fig. 3.

Table 2 illustrates some significant advantages and disadvantages of the proposed IWOA.

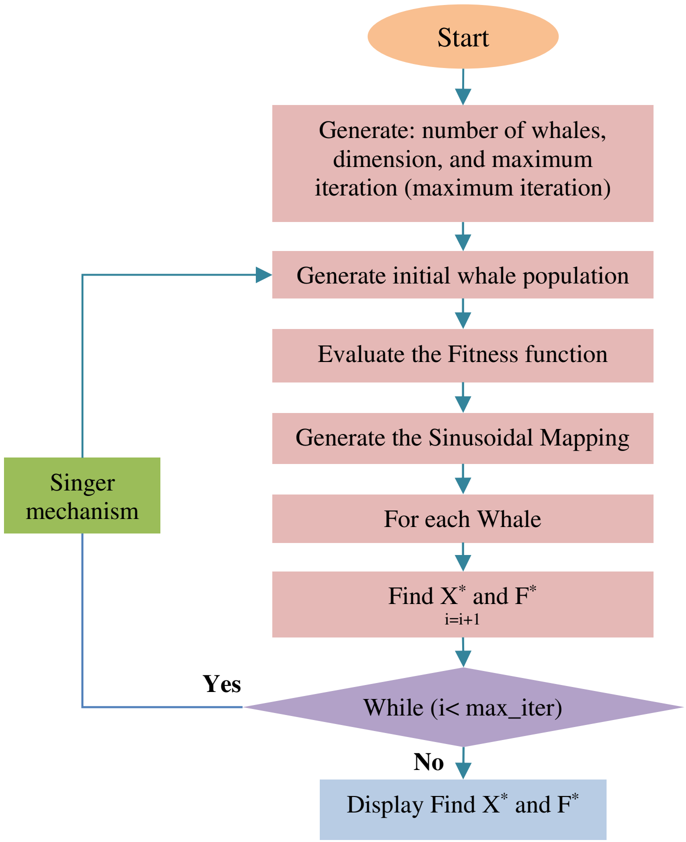


Fig. 3 The block diagram of the proposed improved whale optimization algorithm

Table 2 Advantages and disadvantages of IWOA

Advantages	Disadvantage
Has higher convergence speed	Has the long computational time
Can be robust	Initial value settings are required
Have higher probability and efficiency in finding the global optima	Ability to search for local is weak
Can be efficient for solving problems presenting difficulty to find accurate mathematical models	Has a high dimensional problem
Can be used for large problems	Have a difficult theoretical analysis

According to the results, the proposed Chaos Whale Optimization Algorithm has a simple structure and the only difference between the original WOA and the Chaos WOA is that utilizes a logistic map for updating the algorithm. As a result, the convergence speed is improved. However, the proposed algorithm has the limitation of the high computational effort in terms of methodology and application.

4 Fitness Function

The general fitness function for the PID control system is as follows:

$$ISTSE = 10 \times \int t_{sim}^2 \times e^2 dt + (Os)^2 \tag{10}$$

where, t_{sim} is the simulation time, the variable $e(t)$ represents the tracking error which is the difference between the purposed input value and the actual output. This error signal will be sent to the PID controller and the controller calculates both the derivative and the integral of this error signal and OS is the overshoot. In the PID control designing approaches, the most popular performance criteria are integrated absolute error (IAE), the integrated of time weight square error (ITSE), the integrated squared of time weight and square error (ISTSE), and integrated of squared error (ISE) that can be evaluated analytically in the frequency domain.

These five integral performance metrics in the frequency domain have their own profits and drawbacks. For example, the drawback of the IAE and ISE metrics is that their minimization can result in a response with respectively small overshoot but a long settling time because of the ISE performance criterion weights all errors equally independent of time. However, the ITSE performance metric can dominate the drawbacks of the ISE metric. The derivation processes of the analytical formula are complex and time-consuming. The formulation for the IAE, ISE, and ITSE performance criterion formulas are as follows:

$$IAE = \int |e(t)| dt \tag{11}$$

$$ISE = \int e^2(t) dt \tag{12}$$

$$ITSE = 1000 \times \int t.e^2(t) dt \tag{13}$$

$$ISTSE = 10000 \times \int t^2 e^2(t) dt \tag{14}$$

$$FD = (14 \times OS^2) + (t_s^2) \tag{15}$$

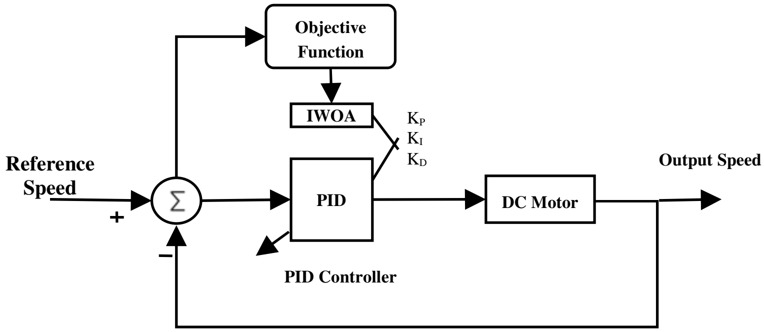


Fig. 4 The block diagram of the proposed PID Controller with IWOA

Table 3 IWOA and WOA PID controller parameters

Characteristic	K_P	K_I	K_D
IWOA-PID	1.5767	0.4297	0.0574
WOA-PID	1.4358	1.6726	0.0416

5 Proposed IWOA Based PID Controller

In this work, a PID controller based on an improved version of the whale optimization algorithm is employed to achieve the optimal parameters of the DC Motor speed control system.

The mechanism of the PID controller with IWOA is shown in Fig. 4. IWOA is applied to the fitness function in order to control the speed of the DC motor. A group of good parameters including *K* and *Resistance* determines a proper step response that will result in performance criteria minimization in the time domain. Table 3 illustrates the performance of the IWOA and standard WOA based PID controllers.

6 Simulation Results

To evaluate the performance of the proposed IWOA-PID controller and its robustness, different operating points are tested in the time domain. To prove the efficiency of the proposed method, a comparison is performed with the designed PID controller with the standard WOA method. Controller Time domain performance is simulated using the model given in Sect. 2. To do so, the operating points of Table 4 are used by considering the changes in *electrical resistance* and *K* parameters. Evaluating operation points are shown in Table 4.

The Output speed of DC motor system per operating points without and with considering the step disturbance's effectiveness is shown in Figs. 5, 6, 7, 8, 9, 10, 11 and 12.

Table 4 Operation points

Case no	R_a (Resistance)	K
1	0.3	0.012
2	0.4	0.015
3	0.1	0.013
4	0.2	0.010

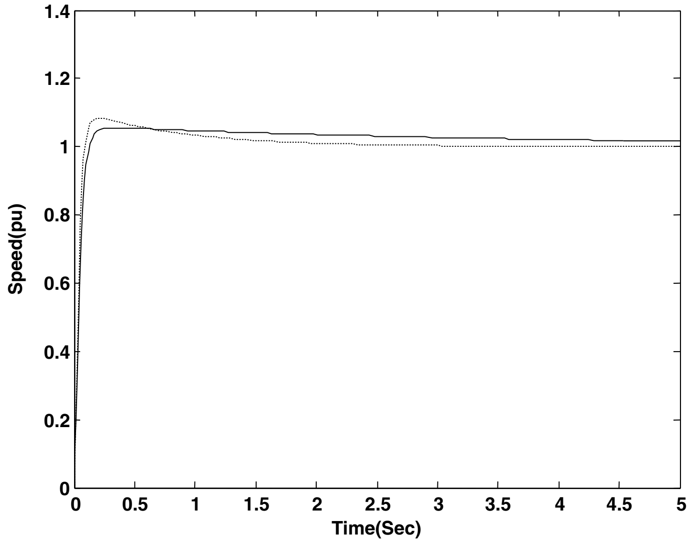


Fig. 5 Output speed system for the operating point (1): solid (IWOA), dashed (WOA)

From the simulation results, it is obvious that the proposed system achieved its stability for all of the operation points. it also results that the WOA-PID controller could reduce the oscillation damping in an admissible value, but increases the damping time and setting time; whereas IWOA-PID controller has a good performance in both time and oscillation damping value.

Tables 5, 6, 7 and 8 present the numerical results for each performance indexes. The value of the indexes' efficiency in IWOA-PID is less than WOA-PID. Therefore, overshoot, settling time and motor speed refraction are reduced by the proposed approach. With regard to the steady-state condition changes and with describing different assessment points, the PID controller is implemented. For the system robustness analysis, simulation is applied for $R_a = (0.1 \ldots 0.4)$, and $K = (0.01 \ldots 0.15)$.

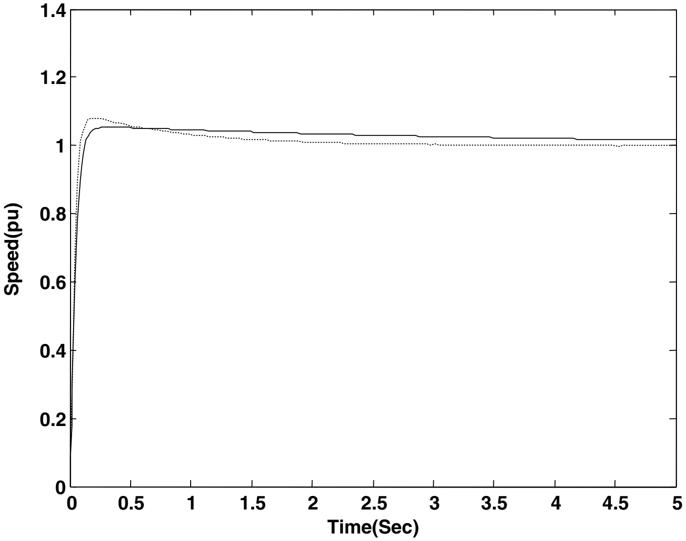


Fig. 6 Output speed system for the operating point (2): solid (IWOA), dashed (WOA)

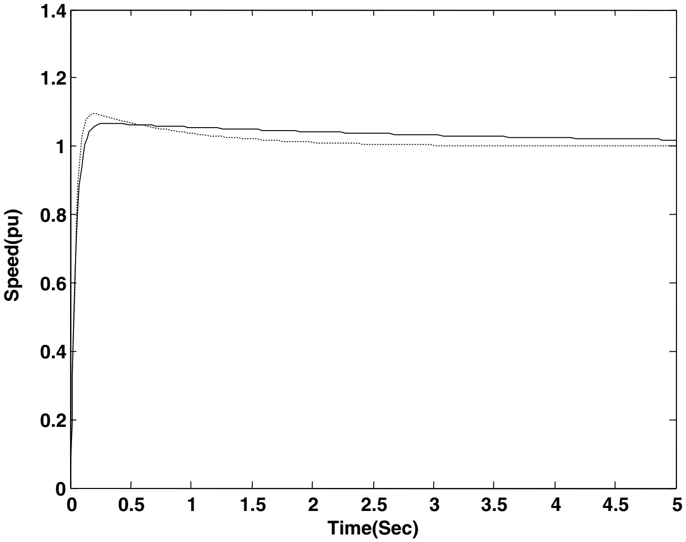


Fig. 7 Output speed system for the operating point (3): solid (IWOA), dashed (WOA)

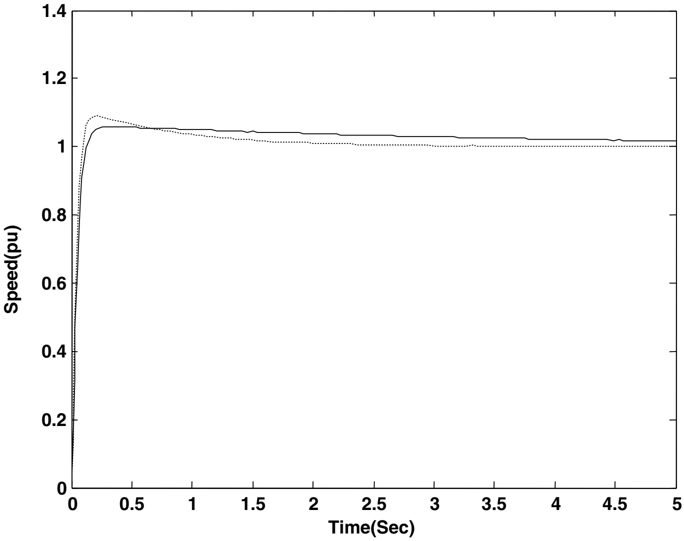


Fig. 8 Output speed system for the operating point (4): solid (IWOA), dashed (WOA)

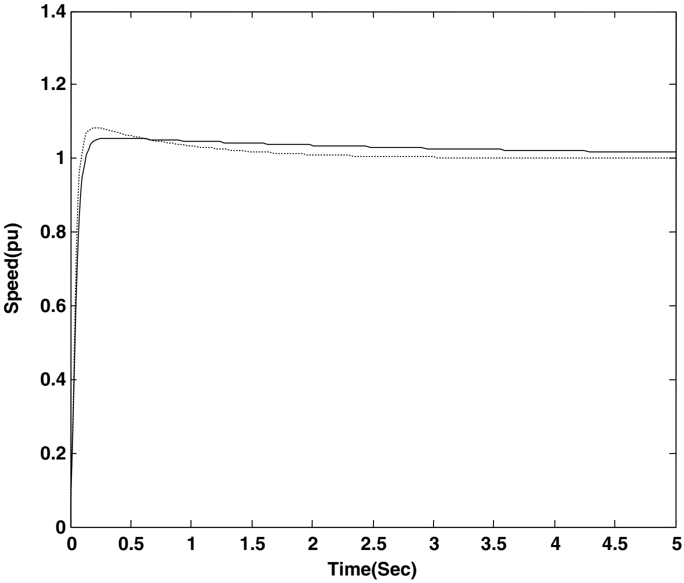


Fig. 9 Output speed system with step disturbance (10%) for the operating point (1): solid (IWOA), dashed (WOA)

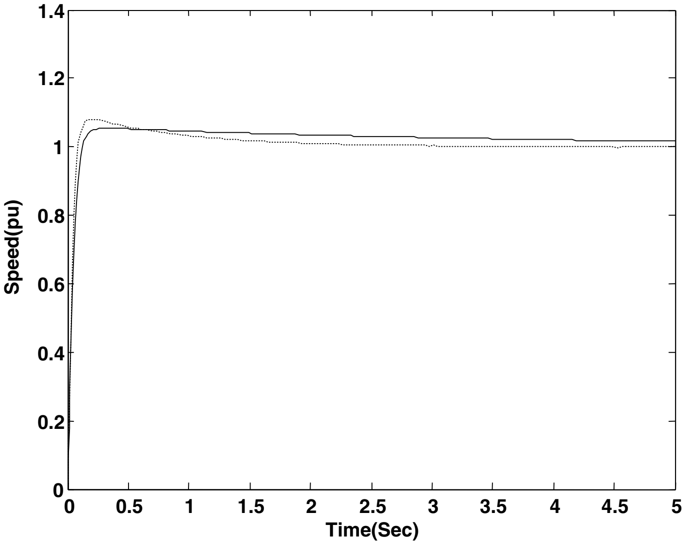


Fig. 10 Output speed system with step disturbance (10%) for the operating point (2): solid (IWOA), dashed (WOA)

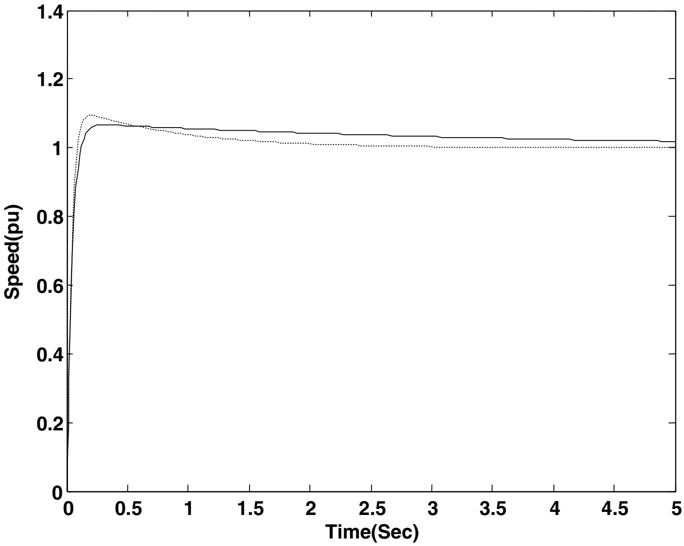


Fig. 11 Output speed system with step disturbance (10%) for the operating point (3): solid (IWOA), dashed (WOA)

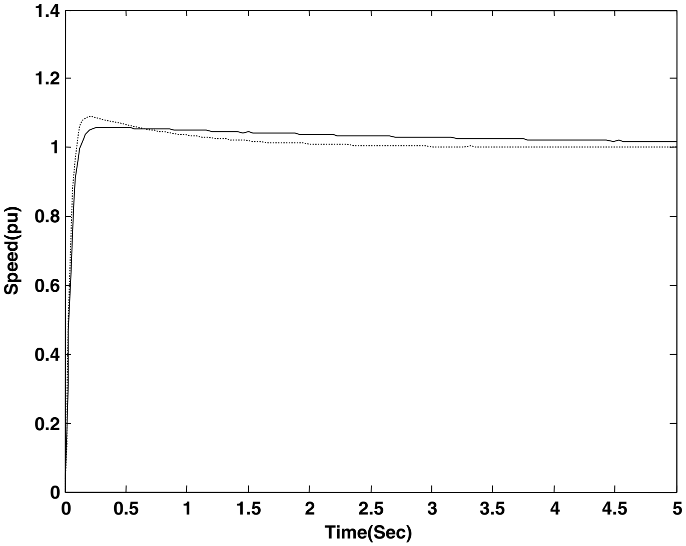


Fig. 12 Output speed system with step disturbance (10%) for the operating point (4): solid (IWOA), dashed (WOA)

Table 5 The results of analysis DC motor system without step disturbance for operating points

Case no	IAE		ISE		ITSE		ISTSE	
	IWOA	WOA	IWOA	WOA	IWOA	WOA	IWOA	WOA
1	0.0518	0.0642	0.028	0.0259	0.6168	0.6572	0.03048	1.9169
2	0.524	0.0654	0.0263	0.0238	0.5477	0.650	0.2817	2.5867
3	0.0583	0.0739	0.275	0.0253	0.6332	0.8325	1.1063	3.9347
4	0.0531	0.0718	0.0298	0.0278	0.6788	0.8155	0.3268	2.8663

Table 6 The characteristics of output response system without step disturbance for operating points

Case no	OS		ts		FD	
	IWOA	WOA	IWOA	WOA	IWOA	WOA
1	0.0041	00.0215	0.1824	0.4798	0.033	0.2393
2	0.0016	0.0281	0.1617	0.5631	0.0262	0.3281
3	0.0053	0.0358	0.1677	0.7731	0.0285	0.6156
4	0.0022	0.0311	0.1866	0.6435	0.0349	0.4277

Table 7 The results of analysis DC motor system with step disturbance (0.1) for operating points

Case no	IAE		ISE		ITSE		ISTSE	
	IWOA	WOA	IWOA	WOA	IWOA	WOA	IWOA	WOA
1	0.1255	0.1136	0.0293	0.0273	2.9	2.4	42	17
2	0.1202	0.1088	0.0274	0.0253	2.7	2.2	39	16
3	0.1395	0.1241	0.0296	0.0277	3.9	2.9	58	22
4	0.1321	0.1213	0.0312	0.0299	3.3	2.7	47	19

Table 8 The characteristics of output response system with step disturbance (0.1) for operating points

Case no	OS		ts		FD	
	IWOA	WOA	IWOA	WOA	IWOA	WOA
1	0.0398	0.0829	1.9531	1.3996	3.8370	2.0551
2	0.0386	0.0799	1.8881	1.4193	3.5860	2.1039
3	0.0476	0.0939	2.2478	1.5194	5.084	2.4323
4	0.0421	0.0892	2.0714	1.4585	4.3156	2.2388

7 Conclusions

In this study, the optimal control of the DC motor with different variations for the resistance and K is analyzed. The purpose is to provide a trade-off between optimal and robust solution for the system control.

Here, a new optimized method is proposed to determine the optimal PID controller parameters. For this purpose, a new version of the whale optimization algorithm is proposed for improving the algorithm convergence. The proposed whale optimization algorithm based PID is applied to the DC Motor drive for optimal control of the system with the minimum settling time and overshoot. Simulation results showed that the proposed technique can efficiently perform for achieving an optimal PID controller. By comparison with the standard WOA-PID controller, it shows that this method can develop the dynamic performance of the system in a better way.

References

1. Khalilpuor M, Razmjooy N, Hosseini H, Moallem P (2011) Optimal control of DC motor using invasive weed optimization (IWO) algorithm. In: Majlesi conference on electrical engineering. Majlesi town, Isfahan, Iran
2. Bandaghiri PS, Moradi N, Tehrani SS (2016) Optimal tuning of PID controller parameters for speed control of DC motor based on world cup optimization algorithm. Parameters 1:2
3. Shamseldin M, Ghany MA, Mohamed AG (2018) Performance study of enhanced non-linear PID control applied on brushless DC motor. Int J Power Electron Drive Syst 9:536

4. Hosseini H, Tousi B, Razmjooy N (2014) Application of fuzzy subtractive clustering for optimal transient performance of automatic generation control in restructured power system. J Intell Fuzzy Syst 26:1155–1166
5. Khanduja N, Bhushan B (2019) CSTR control using IMC-PID, PSO-PID, and hybrid BBO-FF-PID controller. In: Applications of artificial intelligence techniques in engineering. Springer, pp 519–526
6. Agrawal S, Agrawal J, Kaur S, Sharma S (2018) A comparative study of fuzzy PSO and fuzzy SVD-based RBF neural network for multi-label classification. Neural Comput Appl 29:245–256
7. Li T, Zhou J (2018) High-stability position-sensorless control method for brushless DC motors at low speed. IEEE Trans Power Electron
8. Blevins TL (2012) PID advances in industrial control. IFAC Proc 45:23–28
9. Valério D, Da Costa JS (2006) Tuning of fractional PID controllers with Ziegler–Nichols-type rules. Sig Process 86:2771–2784
10. Gamasu R, Jasti VRB (2014) Robust cohen-coon PID controller for flexibility of double link manipulator. Int J Control Autom 7:357–369
11. Åström KJ, Hägglund T (2004) Revisiting the Ziegler-Nichols step response method for PID control. J Process Control 14:635–650
12. Jin CY, Ryu KH, Sung SW, Lee J, Lee I-B (2014) PID auto-tuning using new model reduction method and explicit PID tuning rule for a fractional order plus time delay model. J Process Control 24:113–128
13. Tumari MZM, Abidin AFZ, Hussin MSF, Kadir AMA, Aras MSM, Ahmad MA (2019) PSO fine-tuned model-free PID controller with derivative filter for depth control of hovering autonomous underwater vehicle. In: Proceedings of the 10th national technical seminar on underwater system technology 2018, 2019, pp 3–13
14. Leva A (2018) PID-based controls in computing systems: a brief survey and some research directions. IFAC-PapersOnLine 51:805–810
15. Kaur G, Arora S (2018) Chaotic whale optimization algorithm. J Comput Des Eng 5:275–284
16. Clapham PJ (2000) The humpback whale. Cetacean societies, field studies of dolphins and whales. The University of Chicago, Chicago, pp 173–196
17. Kaveh A, Ghazaan MI (2017) Enhanced whale optimization algorithm for sizing optimization of skeletal structures. Mech Based Des Struct Mach 45:345–362
18. Abdel-Basset M, El-Shahat D, El-henawy I (2020) A modified hybrid whale optimization algorithm for the scheduling problem in multimedia data objects. Concurr Comput Pract Exp, p e5137
19. Mafarja MM, Mirjalili S (2017) Hybrid Whale Optimization algorithm with simulated annealing for feature selection. Neurocomputing 260:302–312
20. Oliva D, El Aziz MA, Hassanien AE (2017) Parameter estimation of photovoltaic cells using an improved chaotic whale optimization algorithm. Appl Energy 200:141–154
21. Wang J, Du P, Niu T, Yang W (2017) A novel hybrid system based on a new proposed algorithm—multi-objective whale optimization algorithm for wind speed forecasting. Appl Energy 208:344–360
22. Yang D, Li G, Cheng G (2007) On the efficiency of chaos optimization algorithms for global optimization. Chaos Solitons Fractals 34:1366–1375
23. Rim C, Piao S, Li G, Pak U (2018) A niching chaos optimization algorithm for multimodal optimization. Soft Comput 22:621–633
24. Yin Z, Razmjooy N (2020) PEMFC identification using deep learning developed by improved deer hunting optimization algorithm. Int J Power Energy Syst 40(2)
25. Cao Y, et al. (2019) Multi-objective optimization of a PEMFC based CCHP system by meta-heuristics. Energy Rep 5:1551–1559
26. Yu D, et al. (2019) System identification of PEM fuel cells using an improved Elman neural network and a new hybrid optimization algorithm. Energy Rep 5:1365–1374

Skin Color Segmentation Based on Artificial Neural Network Improved by a Modified Grasshopper Optimization Algorithm

Navid Razmjooy, Saeid Razmjooy, Zahra Vahedi, Vania V. Estrela, and Gabriel Gomes de Oliveira

Abstract One of the applications of image processing and computer vision is to detect the skin regions for a wide range of human–computer interaction and content based utilizations. Detecting nude parts in the movies, face detection, tracking of human body parts, and people recognizing in multimedia databases are a small part of its applications. Therefore, designing an efficient technique for skin area detection can help a lot to the determined applications. The grasshopper optimization algorithm is a new optimization algorithm evolutionary algorithm which is recently introduced to solve optimization problems. The main purpose of this paper is to propose a newly developed version of this algorithm to optimize the weights of the backpropagation neural network to design a good segmentation tool for skin area segmentation. The method has been compared with the traditional multi-layer perception and the ICA-MLP to declare the proposed method's efficiency.

Keywords Skin segmentation · Artificial neural network · Grasshopper optimization algorithm · Modified · Mathematical morphology

---

N. Razmjooy (✉)
Department of Electrical and Control Engineering, Tafresh University, 39518 79611 Tafresh, Iran
e-mail: navid.razmjooy@hotmail.com

S. Razmjooy
Department of Biosystems Engineering, Faculty of Agriculture Technology and Natural Resources, University of Mohaghegh Ardabili, Ardabil, Iran
e-mail: saeidrazmjooy@gmail.com

Z. Vahedi
Department of Electrical Engineering, Ardabil Branch, Islamic Azad University, Ardabil, Iran
e-mail: zahra23vahedi@gmail.com

V. V. Estrela
Department of Telecommunications, Fluminense Federal University (UFF), Rio de Janeiro, Brazil
e-mail: vania.estrela.phd@ieee.org

G. G. de Oliveira
State University of Campinas, UNICAMP, São Paulo, Brazil
e-mail: oliveiragomesgabriel@ieee.org

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_9

1 Introduction

The science of image processing is one of the most useful sciences in engineering applications and has been extensively studied and researched in the field for many years. The speed of these advances has been so rapid, that now, after a short period of time, the impact of image processing can be clearly seen in many sciences and industries. While some of these applications are dependent on image processing, they cannot be used without it. As the science of image processing is integrated and specialized in today's world, it is becoming more and more important. Recently, human skin area detection has found a wide range of applications in image processing. This field has several utilizations in the human–computer interaction domains [1]. Applications such as face detection [2], detecting and tracking of human body parts [3], naked people detection [4], and people retrieval in multimedia databases [5], all benefit from skin detection [6]. Also, color-based skin detection gains attention in contributing to blocking objectionable images or video content on the Internet automatically [7].

Using the color-based skin detection has an advantage toward grayscale-based one due to the extra dimensions of color such that it may happen a situation in which two objects have similar gray tones but different color space textures. This shows that human skin has a feature for easily recognizing the humans [2]. The color-space features are pixel-based characteristics that need no spatial context which improves its orientation and size invariant and the processing period.

In the recent decade, the applications of artificial neural networks have been increasing exponentially due to its potency in modeling complicated systems [6–8]. Using an artificial neural network (ANN) classifiers for skin-like pixels segmentation is a principal step of skin detection that requires a suitable insulator between the skin and the environment. Among different types of ANNs, a multi-layer perceptron (MLP) network have a wide application due to its simplicity and good efficiency. The connection weights and biases of the MLP networks have been trained normally by a backpropagation learning algorithm [9].

The backpropagation learning algorithm is a classic learning algorithm which is so complicated for big networks. It also has a big problem with trapping in the local minimum to find the minimum error. Several approaches have been used in solving this problem. One of the useful approaches is to use meta-heuristic algorithms. There are different types of meta-heuristics such as Genetic Algorithm (GA) [10], world cup optimization algorithm (WCO) [11], variance reduction of Gaussian distribution (VRGD) algorithm [12], humpback whale optimization (HWO) algorithm [13], Emperor penguin optimizer (EPO) [14]. Meta-heuristic algorithms can improve the ANN efficiency by applying to the different levels such as weight training, learning rules, architecture adaptation (for determining the number of hidden layers, and a number of node transfer functions and hidden neurons).

This study focuses on the weight optimization of ANN to propose an optimized algorithm for better classification of the skin pixels detection. To do so, a new improved meta-heuristic technique called modified grasshopper optimization algorithm has been utilized for optimizing the weights of Multilayer Perceptron (MLP).

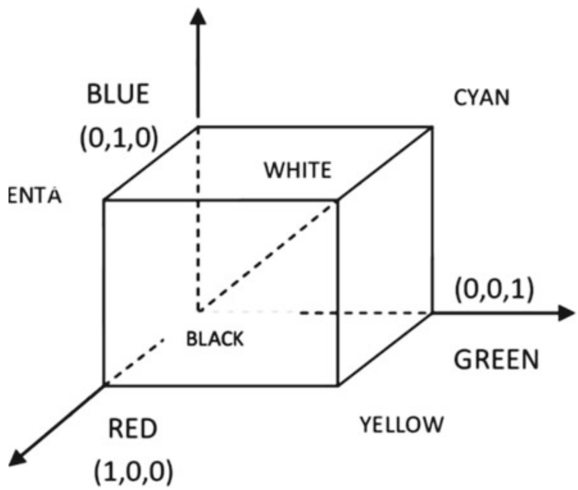
2 RGB Color Space

The purpose of selecting a color model is to facilitate the identification of colors in a standard. In essence, the color model defines a three-dimensional and sub-spatial coordinate system within that system in which each color system is expressed by a single point. Most color models now in use tend toward hardware such as monitors and color printers or applications that aim to work with color, such as producing color graphics for animation. RGB model (blue, red, green) is one of the most common hardware-oriented models for color screens. This model is based on the Cartesian coordinate system. Because of the cube's favorite color space is the image below. In the RGB model, the gray range from black to white is along the line and origin of these two points, and the other colors are points on or inside the cube defined by vectors passing through the origin. To simplify the model, it is assumed that all color values are aligned such that the image cube is below the unit cube, meaning that all values of G, R and B are in the range of [0, 255].

Each image in the RGB color model has three separate pages, each page being the primary color. When these three pages are given to the RGB display, they are combined on the phosphor screen to produce a color image.

So when the images themselves are naturally expressed in terms of three color pages, it makes sense to use the RGB model for image processing. Also, most color cameras use this format for digital imaging, which makes the RGB model an important model in image processing. As can be seen in Fig. 1, the RGB color space is a

Fig. 1 The RGB color cube



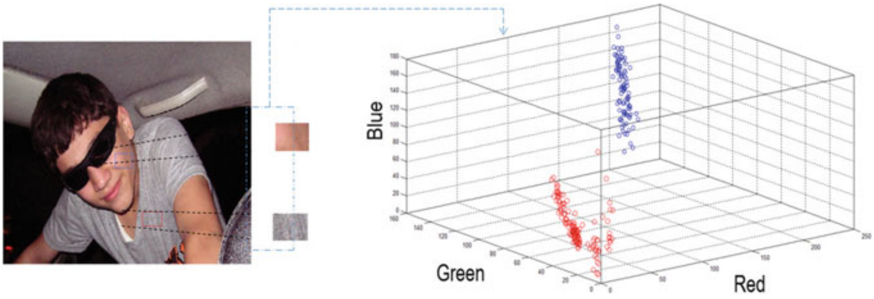


Fig. 2 Analysis of skin colors in the RGB color-space: 3D-plot of skin (blue), 3D-plot of non-skin (red); the skin colors approximately distributed in a linear fashion in the RGB color space

single cube and each axis represents one of the primary colors. The coordinate origin is where the cube lacks three primary colors and this dot represents black, while the opposite vertex is the combination of three primary colors, and this dot in the cube represents white.

The other vertices represent the cyan, purple and yellow secondary colors, each of which comes from a combination of two primary colors. In this model, all other colors within the cube are specified by three components, each point specifying the number of primary colors needed to create a particular color. Each of these components is usually specified by one byte. For example, light red has the value [255, 0, 0] and light yellow the value [255, 255, 0]. Since red, blue and green can be independently assigned, 2563 colors can be produced with three main colors. In the image file, three pages, or rather three matrices, each page is the same color, is used to store each image. Since each pixel is indicated by three bytes, the image depth is 24 bits, and the size of each color image is three times the size of a gray-level image. RGB color images are also known as true color or 24-bit images. Figure 1 shows the RGB Color Cube.

A sample of the 3D plot for skin and non-skin pixels in this space is shown in Fig. 2.

3 Skin Color Feature Extraction

The main application of machine learning algorithms is to extract or learn from existing data. Like a child who learns by observing people and parental recommendations. For example, he or she can figure out whether the heater is hot or should not go down the street alone. In fact, machine learning algorithms work to find a summarize of data or to actually create a model of data in any way. As you know, a model always contains a summary of the data. The World Map, for example, is a model of the world that models countries and roads without much detail (for example, what shop is on the street).

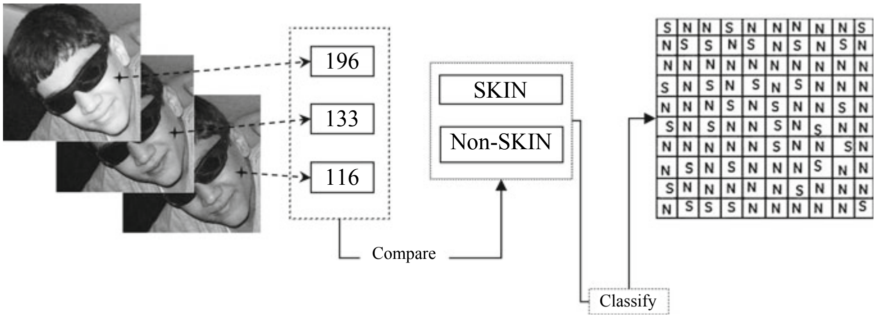


Fig. 3 Steps in supervised classification

Supervised classification is a type of machine learning in which input and output are specified and there is a so-called supervisor that provides information to the learner, thereby trying to learn a function from input to output. After extracting the RGB color information from the input image for skin region detection, it should be used as input data for the classifier to the classification. Image classification is a principal tool in image processing with different applications. This tool can be also utilized for image segmentation which is our purpose in this study. The classifier system divides the input data into two parts of the training and testing categories. In this study, 60% of data has been utilized for training and the 40% remained data have been selected as the testing set. Figure 3 shows the steps of classification.

4 The Improved Grasshopper Optimization Algorithm

4.1 The Standard Grasshopper Optimization Algorithm

The grasshopper optimization algorithm (GOA) is a new swarm-based optimization algorithm that is derived from grasshopper insect behavior [15].

In this algorithm, the population includes a collective of grasshoppers which is called a swarm. Each member of the swarm is a probable solution to the problem. The first step is started by generating a random swarm as the initial solution to the problem [16]. Then, the cost of each grasshopper is determined by obtaining the value of the cost function.

The process is continuous by absorbing the swarm via considered grasshoppers into their location to attract the grasshoppers to move into the considered grasshopper. Here, two main behaviors of the grasshoppers are considered: the tiny and slow movement of the larval grasshoppers in order to the long-range and no sequence movement drive adults and food searching process which is divided into two separate parts of exploitation and exploration.

The development of the *ith* grasshopper next to the target grasshopper is defined by P_i and is obtained as below:

$$P_i = R_1 SA_i + R_2 GF_i + R_3 WA_i \tag{1}$$

where, GF_i describes the gravity force on the *ith* grasshopper, WA_i is the wind advection, and SA_i describes the social interaction, and R_1, R_2, and R_3 are random constants in the interval [0, 1].

The social interaction of the *ith* grasshopper (SA_i) related to the social forces between two grasshoppers and is a repulsion force to stop collisions and an attraction force purpose over a small length scale.

$$SA_k = \sum_{\substack{l=1 \\ l \neq i}}^{N} SA(\mathrm{D}_{lk})\hat{D}_{lk} \tag{2}$$

$$D_{lk} = X_k - X_l \tag{3}$$

$$\hat{D}_{lk} = \frac{X_k - X_l}{D_{lk}} \tag{4}$$

where, D_{lk} describes the length of the Euclidian of the *kth* with the *lth* position grasshopper, and $\hat{D}_{lk}$ represents the present unit vector between the *lth* grasshopper and the *kth* grasshopper.

The intensity attraction strength is modeled by the following formula:

$$X_k^d = c\left(\sum_{\substack{l=1 \\ l \neq 1}}^{N} c\frac{U_B^d - L_B^d}{2} SF\left(\left|X_l^d - X_k^d\right|\right)\frac{X_l - X_k}{D_{lk}}\right) + \hat{T}_d \tag{5}$$

where, *SF* determines the strong point of the social forces that are evaluated by the following:

$$SF(R) = f_i e^{-R/L} - e^{-R} \tag{6}$$

where, L describes the length scale of attraction, and f_i is the intensity force of attraction.

GF is another parameter of the algorithm that is achieved as follows:

$$GF_i = -GF\hat{e}_g \tag{7}$$

where, G_i describes a constant for the gravity force, and $\hat{e}_g$ is a direction for unity vector alongside the wind.

The wind advection model (WA_i) is finally obtained as follows:

$$WA_i = U\hat{e}_g \tag{8}$$

where, U describes the drift constant.

The gravity assumptions and wind effects can be formulated as follows:

$$x_l^d = c\left(\sum_{\substack{l=1 \\ l\neq 1}}^{N} c\frac{U_B^d - L_B^d}{2} SF\left(\left|X_k^d - X_l^d\right|\right)\frac{X_l - X_k}{D_{lk}}\right) + \hat{T}_d \tag{9}$$

$$c = c_{\max} - l\frac{c_{\max} - c_{\min}}{L} \tag{10}$$

where, N is the grasshoppers quantity, LB_d and UB_d are the lower and the upper limitations in the dth dimension, respectively, $\hat{T}_d$ acts objective magnitude of dth dimension by the objective grasshopper, c describes the reducing factor to the proper area of the repulsion region and attraction region, $c_{\max}$ and $c_{\min}$ are the highest value and the lowest value of the factor c, respectively, and finally, l and L are the current iteration and the total number of iteration, respectively.

4.2 Improved Grasshopper Optimization Algorithm Based on Chaos Theory

In this subsection, a technique is proposed for modifying the essential parameters of the Grasshopper optimization algorithm to improve its convergence speed. The keys parameters of Grasshopper optimization algorithm convergence are R_1, R_2, R_3, and U. the modification is applied based on chaos theory.

Chaos theory is the study of unpredictable and random processes. The main idea of the chaos theory is to study the highly sensitive dynamic systems that can be affected by any small variations.

By the explanations above, a large diversity can be made for the swarm generation in GOA to improve the diversity of the algorithm.

This part improves the GOA algorithm capability from the point of the convergence speed and also for escaping from falling into the local optimal point [17, 18]. A general form for the chaos theory is formulated below:

$$\begin{aligned} &CM_{i+1}^j = f(CM_i^j) \\ &j = 1, 2, \ldots, k \end{aligned} \tag{11}$$

where, k describes the map dimension, and $f(CM_i^j)$ is the chaotic model generator function.

In the presented GOA, the parameters R_1, R_2, R_3, and U are modeled based on the Singer mechanism chaotic map as follows:

$$\begin{aligned} &R_{(j,i+1)} = 1.07 \times \left(7.9R_{(j,i)} - 23.3R^2_{(j,i)} + 28.7R^3_{(j,i)} - 13.3R^4_{(j,i)}\right) \\ &U_{(i+1)} = 1.07 \times \left(7.9R_i - 23.3R_i^2 + 28.7R_i^3 - 13.3R_i^4\right) \\ &R_{(j,0)} = rand(),\ j = 1, 2, 3, \\ &U_0 = rand() \end{aligned} \tag{12}$$

where, k is the number of iteration.

Figure 4 shows the flowchart diagram of the presented CGOA.

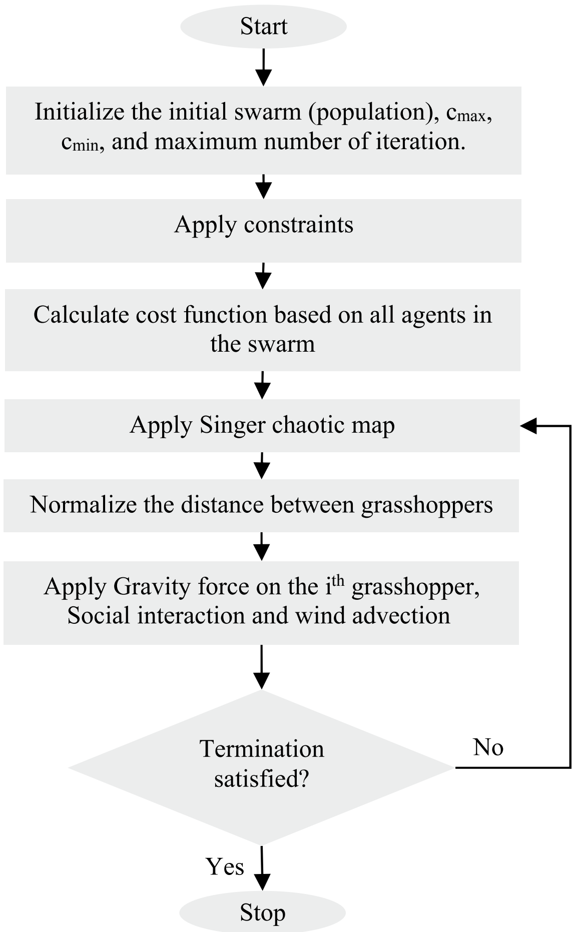


Fig. 4 The diagram flowchart of the CGOA

4.3 Validation of the CGOA

For performance analysis of the presented method, 4 standard test functions have been studied. Experimental simulation s of the presented CGOA are compared with some different popular and newly introduced algorithms including genetic algorithm (GA) [10], shark smell optimization (SSO) algorithm [19], particle swarm optimization algorithm (PSO) [20], world cup optimization algorithm (WCO) [11], and the original grasshopper optimization algorithm (GOA) [16].

The simulations are established based on MATLAB R2017b platform on a laptop computer with processor Intel® CoreTM i7-4720 HQ CPU@2.60 GHz with 16 GB RAM. Table 1 demonstrates the results of the simulation.

The results of the mean deviation (MD) and the standard deviation (SD) values of the methods for the analyzed benchmarks are illustrated in Table 2.

Table 1 The utilized benchmarks for efficiency analysis

Benchmark	Formula	Constraints	Dimension
Rastrigin	$f_1(x) = 10D + \sum_{i=1}^{D} \left(x_i^2 - 10\cos(2\pi x_i)\right)$	[−512, 512]	30–50
Rosenbrock	$f_2(x) = \sum_{i=1}^{D-1} \left(100\left(x_i^2 - x_{i+1}\right) + (x_i - 1)^2\right)$	[−2.045, 2.045]	30–50
Ackley	$f_3(x) = -20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i)\right) + 20 + e$	[−10, 10]	30–50
Sphere	$f_4(x) = \sum_{i=1}^{D} x_i^2$	[−512, 512]	30–50

Table 2 The results of the efficiency analysis by considering 30-dimensions

Benchmark		CGOA	GA [10]	PSO [20]	WCO [11]	GOA
f_1	MD	0.00	70.61	74.24	2.19	3.53
	SD	0.00	1.66	8.96	4.35	3.47
f_2	MD	9.53	35.41	200.1	13.16	9.71
	SD	2.87	27.15	59.00	4.62	3.86
f_3	MD	0.00	3.19e−2	8.26	3.14e−3	5.19e−17
	SD	0.00	2.14e−2	1.19	1.12e−3	0.00
f_4	MD	0.00	1.15e−4	8.27e−4	6.19e−9	0.00
	SD	0.00	3.14e−5	5.12e−4	3.28e−9	0.00

The results from Table 2 show that in all four benchmarks, the proposed CFSO algorithm gives satisfying results toward the other methods, especially the original FSO algorithm.

5 Artificial Neural Network

The very high speed and processing power of the human brain go back to the very large interconnections that exist between the brain-forming cells, and basically, without these communication links, the human brain would be reduced to a normal system and certainly not have the current capabilities. After all, the brain's excellent performance in solving a variety of problems and its high performance has made brain simulation and its capabilities the most important goal of hardware and software architects. In fact, if there is a day (but apparently not too far away) that we can build a computer with the same capabilities of the human brain, there will surely be a major revolution in science, industry, and of course, human life. Since the decades that computers have enabled computational algorithms to simulate human computational behavior, much research has been undertaken by computer scientists, engineers, and mathematicians, whose results are in the field of artificial intelligence and it is classified under the subcategory of computational intelligence as the topic of "Artificial Neural Networks". In the field of artificial neural networks, numerous mathematical and software models have been proposed to inspire the human brain which is applied to solve a wide range of scientific, engineering and practical problems in various fields.

Several types of computational models have been introduced as generic artificial neural networks, each of which can be used for a variety of applications, each inspired by a particular aspect of the capabilities and properties of the human brain. In all of these models, a mathematical structure is assumed to be graphically presentable and has a set of adjustable parameters. This general structure is adjusted and optimized by a training algorithm so that it can exhibit good behavior. A look at the learning process in the human brain also shows that we actually experience a similar process in the brain and all of our skills, knowledge, and memories are shaped by the weakening or strengthening of the communication between neurons. This reinforcement and weakening in mathematical language model and describe itself as a parameter (called Weight). One of the most basic neural models available is the Multi-Layer Perceptron (MLP) model that simulates the transient function of the human brain. In this type of neural network, most of the network behavior of the human brain and its signal propagation has been taken into account, and hence are sometimes referred to as Feedforward Networks. Each of the neurons in the human brain, called neurons, processes it after receiving input (from one neuronal or non-neuronal cell) and transmits the result to another cell (neuronal or non-neuronal). This behavior continues until a definite result is reached, which may eventually lead to a decision, process, thought, or move. The most popular method in the feed-forward networks is backpropagation (BP). This method evaluates the error on all of the training pairs and adjusts the weights to fit the desired output. To do so, several epochs have been considered and the process continues until achieving the minimum value for the total error on the training set or until the termination criteria are reached.

This method adopts supervised learning to train the network using the data for which inputs, as well as target outputs, are noticed. After training, the weights of the networks are fixed and can be employed for evaluating the output values of the new given samples.

BP is a method based on gradient descent algorithm on the error space that sometimes gets trapped into the local minimum, succeeded it entirely dependent on initial (weight) settings. This drawback can be covered by an exploration searching capability of the evolutionary algorithms.

6 ANN Weights Evolution Using CGOA

Due to modeling by artificial neural networks, they can obtain the desired precision to determine the optimal network architecture by receiving feedback from the network and the test process and time consuming, but this increases the hidden layers and the complexity of the neural network architecture. CGOA can be used as a method to find optimal values of different neural network parameters. This algorithm starts the search with a primitive set of random solutions called the initial population and can perform multilaterally and user searches on a population of variables that increase the likelihood of finding the global optimal point. This algorithm can then simultaneously determine the optimal network structure and weights using a function that is a measure of the optimal performance quality of the response produced. The method of optimal parameter selection of the weights based on the proposed grasshopper optimization algorithm can be formulated as follows:

1. Start algorithm
2. Generate N number of initial random grasshoppers as the weight of the network
3. Calculate the cost of the network based on each grasshopper
4. Apply the CGOA operators to each of the grasshoppers to generate the next generation
5. If the termination condition is reached, go (5), else go (2)
6. End of algorithm.

Consider a simple multi-layer perceptron network with its weights (w) and biases (b):

$$\sum_{i=1}^{H} w_i f\left(\sum_{j=1}^{d} w_j x_j + b\right) \tag{13}$$

where, H is the number of neurons in the hidden layer, w describes the weights of the network, b denotes the bias value and f is the activation function of each neuron which in this case is considered as sigmoid.

The network can be optimized by optimal selection of the weights and biases based on minimizing the mean squared error of the network (MSE) as follows:

$$MSE = \frac{1}{2}\sum_{k=1}^{g}\sum_{j=1}^{m}\left(Y_j(k) - T_j(k)\right)^2 \quad (14)$$

where, m describes the output nodes number, g represents the number of training samples, $Y_j(k)$ is the desired output, and $T_j(k)$ describes the real output.

7 Dataset Description

We used the Bao dataset to show our proposed method; this database includes 370 face images from various races, with 221 images with multiple people and other 149 images of one person mostly from Asia, with a wide range of size, lighting and background. In this paper, Bao Face Database has been utilized. This database is introduced by Frischholz that includes 370 face images from different races, mostly from Asia, with a wide range of size, lighting and background [21]. The database has no using or distribution policies or license.

8 Simulation Results

This study considers two categories including skin and non-skin classes. The method is based on pixels information that uses this information for classifying the skin-like and non-skin-like pixels one by one. This method reduces the image information into objects of interest, like faces or hands during skin pixels detection.

The input of the proposed classifier is a matrix of 3 × n pixel coefficients from the images either skin or non-skin image which n determines the number of neurons in the hidden layer. Since the employed transfer function is sigmoid, the output image gives different values in the range 0 and 255 (uint8 mode).

Since we need two labels in the final output (skin or non-skin), the output of the neural classifier should be thresholded so that it is either 0 or 1. In this study, Otsu thresholding has been used for this purpose, such that the output values of the neural network have been classified into two categories of 0 and 1 [22].

After thresholding the output, mathematical morphology has been applied to remove the extra parts. In this study, three operators have been utilized including opening, closing and filing holes [23]. Figure 5 shows the impact of the mathematical morphology on the images.

For validating the efficiency in the proposed system, three performance indices have been employed. The first metric is the false acceptance rate **(FAR)** that defines the ratio of the identification moments in which false acceptance happens. The second index is the false rejection rate **(FRR)** which is the percentage of the identification moments in which false rejection happens and the final index is the correct detection rate **(CDR)** which shows the correctly classified pixels to the total pixels.

$$CDR = \frac{No.of\ Pixels\ Correctly\ Classified}{Total\ Pixels\ in\ the\ Test\ Dataset} \tag{15}$$

$$FAR = \frac{No.of\ non-skin\ Pixels\ Classified\ as\ skin\ Pixels\ Classified}{Total\ Pixels\ in\ the\ Test\ Dataset} \tag{16}$$

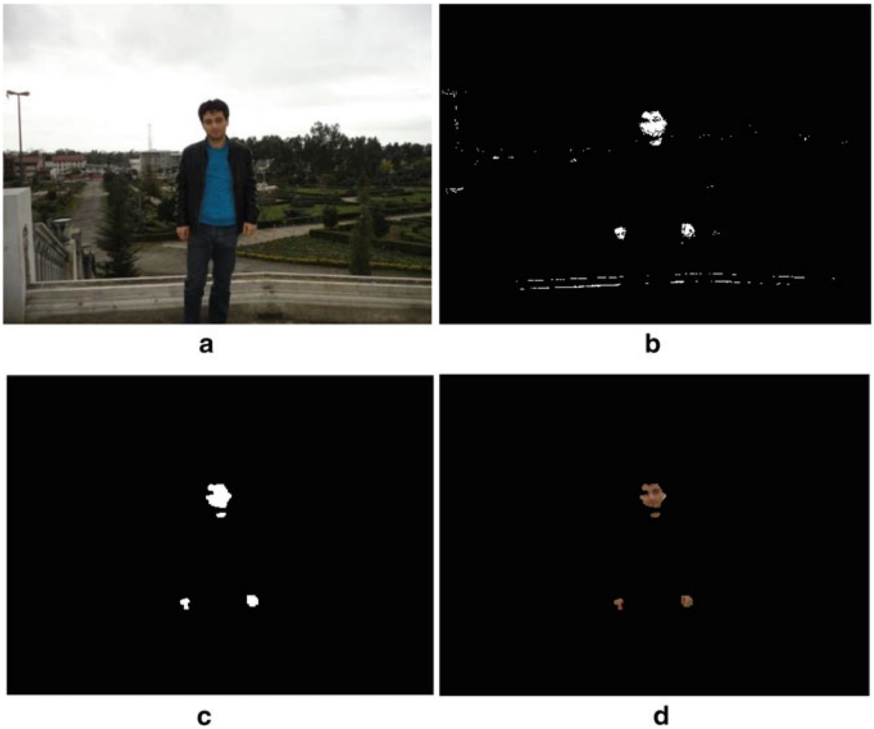


Fig. 5 **a** Original image, **b** skin region segmentation and thresholding, **c** applying mathematical morphology, **d** skin area detection

Fig. 6 Test set: **A** skin sets, **B** non-skin sets

Table 3 The performance of the skin region detection for the methods

Algorithm	CDR	FRR	FAR
CGOA-ANN	82.15	7.45	10.4
HNN-ICA	70.84	4.16	25
MLP	68.42	5.2	26.38

$$FRR = \frac{No.\,of\ skin\ Pixels\ Classified\ as\ non - skin\ Pixels\ Classified}{Total\ Pixels\ in\ the\ Test\ Dataset} \tag{17}$$

Figure 6 shows the utilized test set from the skin and the non-skin to measure the metric efficiency.

Here, the least mean square error minimization has been utilized for improving the network weights between the desired and the real outputs of the network. Comparison results of the proposed method with basic MLP and HNN-ICA [6] are tabulated in Table 3. As can be observed, the proposed optimized network performs better and gives a higher correct detection rate.

Figures 7 and 8 show the extent of the match between the measured and predicted skin rate of the train set and validation phases by CGOA-ANN in terms of a scatter diagram.

For more clarification, some sample outputs have been shown in Fig. 9. As can be observed, using the proposed CGOA-ANN classifier gives satisfying results for skin regions detection including face, hands, and neck.

9 Conclusion

This paper proposed a new optimized method for skin area classification. A new improved version of the grasshopper optimization algorithm was adopted to decrease the mean square error of the neural network to escape from the local minimum. Here, RGB color space was utilized for decreasing the computational cost. The proposed algorithm reduces the number of false detection ratio toward a gradient descent algorithm. The skin region detection method based on CGOA-ANN is compared with some methods to validate and analyze the efficiency.

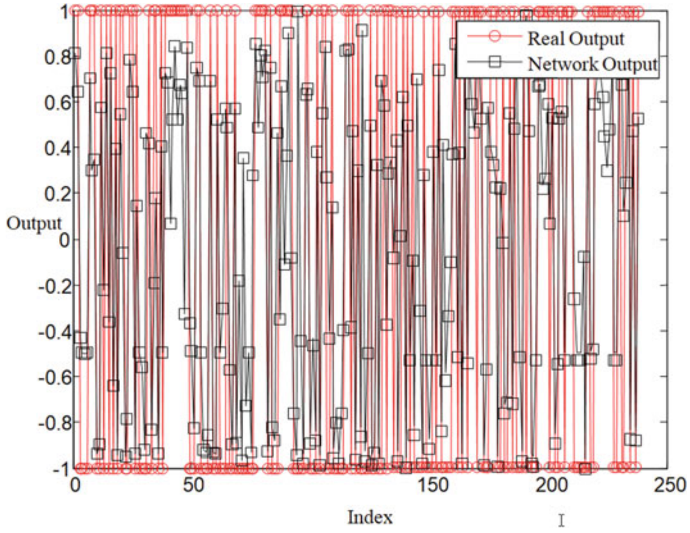


Fig. 7 Real output and CGOA-ANN network output for the training data

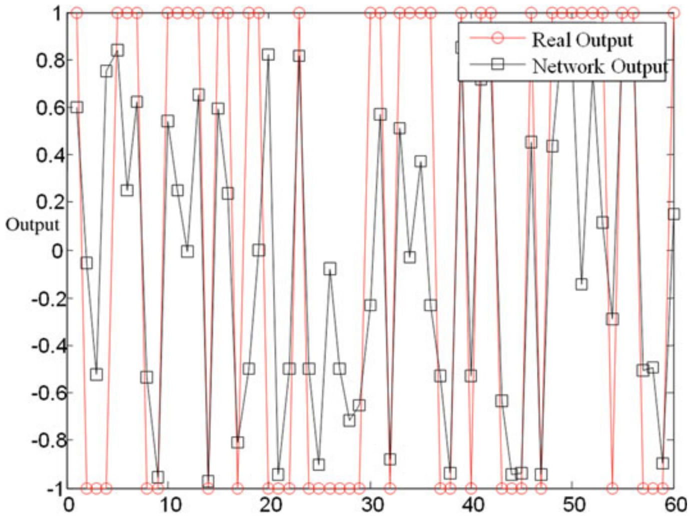


Fig. 8 Real output and CGOA-ANN network output for the test data

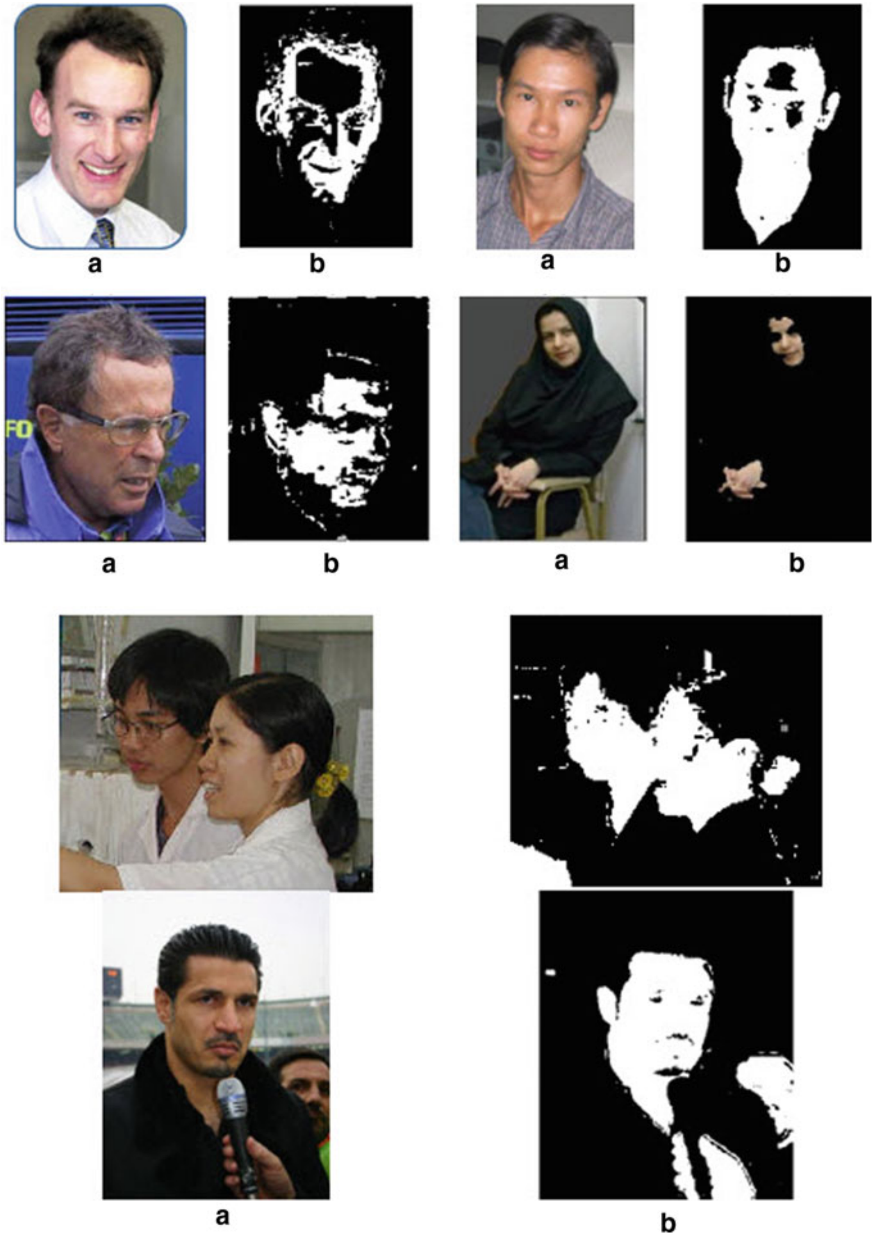


Fig. 9 Experimental results for skin region detection: **a** original image, **b** skin segmented result

References

1. Rashid Sheykhahmad F, Razmjooy N, Ramezani M (2015) A novel method for skin lesion segmentation. Int J Inf Secur Syst Manag 4:458–466
2. Sun X, Wu P, Hoi SC (2018) Face detection using deep learning: an improved faster RCNN approach. Neurocomputing 299:42–50
3. Jalal A, Kamal S, Azurdia-Meza CA (2019) Depth maps-based human segmentation and action recognition using full-body plus body color cues via recognizer engine. J Electr Eng Technol 14:455–461
4. Tian C, Zhang X, Wei W, Gao X (2018) Color pornographic image detection based on color-saliency preserved mixture deformable part model. Multimed Tools Appl 77:6629–6645
5. Aygun R, Benesova W (2018) Multimedia retrieval that works. In: 2018 IEEE conference on multimedia information processing and retrieval (MIPR), pp 63–68
6. Razmjooy N, Mousavi BS, Soleymani F (2013) A hybrid neural network Imperialist Competitive Algorithm for skin color segmentation. Math Comput Model 57:848–856
7. Razmjooy N, Ramezani M (2016) Training wavelet neural networks using hybrid particle swarm optimization and gravitational search algorithm for system identification. Int J Mechatron Electr Comput Technol 6:2987–2997
8. Razmjooy N, Sheykhahmad FR, Ghadimi N (2018) A hybrid neural network–world cup optimization algorithm for melanoma detection. Open Med 13:9–16
9. Zhang Z (2018) Artificial neural network. In: Multivariate time series analysis in climate and environmental research. Springer, pp 1–35
10. Holland JH (1992) Genetic algorithms. Sci Am 267:66–73
11. Razmjooy N, Khalilpour M, Ramezani M (2016) A new meta-heuristic optimization algorithm inspired by FIFA world cup competitions: theory and its application in PID designing for AVR system. J Control Autom Electr Syst 27:419–440
12. Namadchian A, Ramezani M, Razmjooy N (2016) A new meta-heuristic algorithm for optimization based on variance reduction of Gaussian distribution. Majlesi J Electr Eng 10:49
13. Clapham PJ (2000) The humpback whale. In: Cetacean societies, field studies of dolphins and whales. The University of Chicago, Chicago, pp 173–196
14. Dhiman G, Kumar V (2018) Emperor penguin optimizer: a bio-inspired algorithm for engineering problems. Knowl-Based Syst 159:20–50
15. Mirjalili S, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51–67
16. Mirjalili SZ, Mirjalili S, Saremi S, Faris H, Aljarah I (2018) Grasshopper optimization algorithm for multi-objective optimization problems. Appl Intell 48:805–820
17. Yang D, Li G, Cheng G (2007) On the efficiency of chaos optimization algorithms for global optimization. Chaos Solitons Fract 34:1366–1375
18. Rim C, Piao S, Li G, Pak U (2018) A niching chaos optimization algorithm for multimodal optimization. Soft Comput 22:621–633
19. Abedinia O, Amjady N, Ghadimi N (2018) Solar energy forecasting based on hybrid neural network and improved metaheuristic algorithm. Comput Intell 34:241–260
20. Bansal JC (2019) Particle swarm optimization. In: Evolutionary and swarm intelligence algorithms. Springer, pp 11–23
21. Frischholz R (2012) Bao face database at the face detection homepage
22. Otsu N (1979) A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern 9:62–66
23. Razmjooy N, Mousavi BS, Soleymani F (2012) A real-time mathematical computer method for potato inspection using machine vision. Comput Math Appl 63:268–279

A New Design for Robust Control of Power System Stabilizer Based on Moth Search Algorithm

Navid Razmjooy, Saeid Razmjooy, Zahra Vahedi, Vania V. Estrela, and Gabriel Gomes de Oliveira

Abstract This paper presents a new optimal design for the stability and control of the synchronous machine connected to an infinite bus. The model of the synchronous machine is 4th order linear Philips-Heffron synchronous machine. In this study, a PID controller is utilized for stability and its parameters have been achieved optimally by minimizing a fitness function to removes the unstable Eigen-values to the left-hand side of the imaginary axis. The considered parameters of the PID controller are optimized based on a new nature-inspired, called moth search algorithm. The proposed system is then compared with the particle swarm optimization as a high-performance and popular algorithm for different operating points. Final results show that using a moth search algorithm gives better efficiency toward the compared particle swarm optimization.

Keywords Power system stabilizer · Philips-Heffron model · Single machine · Infinite bus · Moth search algorithm

N. Razmjooy (✉)
Department of Electrical and Control Engineering, Tafresh University, 39518 79611 Tafresh, Iran
e-mail: navid.razmjooy@hotmail.com

S. Razmjooy
Department of Biosystems Engineering, Faculty of Agriculture Technology and Natural Resources, University of Mohaghegh Ardabili, Ardabil, Iran
e-mail: saeidrazmjooy@gmail.com

Z. Vahedi
Department of Electrical Engineering, Ardabil Branch, Islamic Azad University, Ardabil, Iran
e-mail: zahra23vahedi@gmail.com

V. V. Estrela
Department of Telecommunications, Fluminense Federal University (UFF), Niterói, Brazil
e-mail: vania.estrela.phd@ieee.org

G. G. de Oliveira
State University of Campinas, UNICAMP, São Paulo, Brazil
e-mail: oliveiragomesgabriel@ieee.org

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_10

1 Introduction

In recent decades, the stability of the synchronous machine has great attention and it will receive additive attention in the future [1, 2]. Once an error occurs, it is depended on the synchronous machine dealing after implementing. The stability problem has been analyzed from two directions: steady-state stability and transient state stability [3]. Steady-state stability analysis is to study the power system and its generators in strictly steady-state conditions and try to find the maximum possible generator load that can be transmitted without losing the synchronism of any other generator [4]. Transient stability is the ability of the power system to guarantee synchronism when subjected to a sudden and large disturbance in limited time such as a fault on transmission facilities, loss of a large load or loss of generation [5].

In this study, a new nature-inspired, called moth search algorithm (MSO) is presented and adopted for optimal selection of the PID controller parameters in the power system stabilizer. MSO algorithm is a new optimization algorithm that is introduced for solving the optimization problems [6]. This algorithm, like other nature-inspired algorithms, doesn't require the gradient of the function in its optimization process. MSO is a mathematical inspiration and the computer modeling of the behavior of moths. MSO algorithm starts with a set of the random population as solutions agents (moths). These moths have moved around in the search space based on the algorithm formulas.

In this study, for optimal selection of the PID parameters, the fitness function is designed such that it shifts Eigenvalues into the left side of the real axis. Eigenvalue analysis is applied to evaluate the impact of the optimized PSSs to damp the electromechanical modes of oscillations and improve the system dynamic stability. The efficiency of the MSO algorithm is compared with the particle swarm optimization is a widely used algorithm to show the proposed method's advantages and disadvantages.

2 Single Machine Infinite Bus (SMIB) Model

In this study, a single machine connected to an infinite bus system is studied through a transmission line including resistance r_e and inductance x_e. The system structure has been shown in Fig. 1.

Synchronous machines are major sources of electricity in power systems. The problem with power system stability is mainly to maintain a synchronized state between interconnected synchronous machines. Therefore, characteristic understanding and accurate modeling of their dynamic response are of great importance in studying the stability of power systems. Since frequency stability and voltage are important factors in determining the quality of the power source, real power control and reactive power in the satisfactory performance of power systems. A general

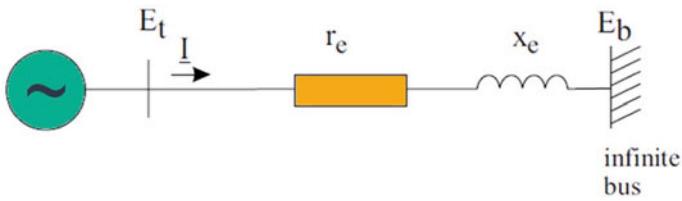


Fig. 1 Single machine infinite bus model

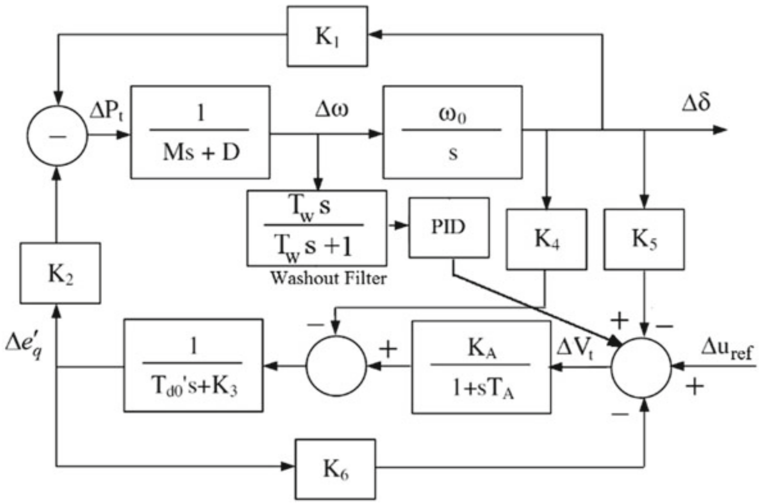


Fig. 2 Heffron- Philips model of synchronous machine

model of a synchronous machine connected to the infinite bus through the transmission network can be modeled by Thevenin's equivalent. In this study, a 4-ordered linear Heffron-Philips synchronous machine has been studied (Fig. 2).

In this model, there are six constants (K_1–K_6) for a synchronous generator which can represent the interaction between the voltage and the speed control equations of the machine. Except for K_3, the other five constants are dependent on the active and the reactive loading of the machine. The formulation of this model is given in the following:

$$\Delta\dot{\delta} = \omega_0 \Delta\omega \tag{1}$$

$$\Delta\dot{\omega} = \frac{1}{M}\left(-K_1\Delta\delta - D\Delta\omega - K_2\Delta e'_q\right) \tag{2}$$

$$\Delta\dot{e}'_q = \frac{1}{T'_{do}}\left(-K_4\Delta\delta - \frac{\Delta e'_q}{K_3} + \Delta e_{fd}\right) \tag{3}$$

$$\Delta \dot{e}_{fd} = \frac{1}{T_e}\left(-K_e K_5 \Delta\delta - K_e K_6 \Delta e'_q - \Delta e_{fd} + K_e u\right) \tag{4}$$

where, ω_0 represents the synchronous speed, ω is the rotor speed per unit (p.u.), δ describes the angular position if the in the synchronous generator to a reference axis, M is the inertia In the rotor, D describes the damping coefficient of the rotor motion, T'_{do} represents d-axis open circuit field time constant, $\Delta e'_q$ is the change of q-axis internal voltage, e_{fd} describes the field voltage, u describes the control vector, K_e and T_e are exciter gain and time constants, respectively.

Equations (1)–(4) can be represented in the state space representation by the following:

$$\dot{x} = Ax + Bu \tag{5}$$

where, A describes the system coefficient matrix and B is the input matrix. The general state-space model of the system without any controller is described below:

$$\begin{bmatrix} \Delta\dot{\delta} \\ \Delta\dot{\omega} \\ \Delta\dot{e}_q \\ \Delta\dot{e}_{fd} \end{bmatrix} = \begin{bmatrix} 0 & \omega_0 & 0 & 0 \\ -\frac{k_1}{M} & 0 & -\frac{k_2}{M} & 0 \\ -\frac{k_4}{T_{do}} & 0 & -\frac{1}{k_3 T_{do}} & \frac{1}{T_{do}} \\ -\frac{k_e k_5}{T_e} & 0 & -\frac{k_e k_6}{T_e} & -\frac{1}{T_e} \end{bmatrix} \begin{bmatrix} \Delta\delta \\ \Delta\omega \\ \Delta e_q \\ \Delta e_{fd} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{1}{M} & 0 \\ 0 & 0 \\ 0 & \frac{k_e}{T_e} \end{bmatrix} \begin{bmatrix} \Delta P_m \\ \Delta V_{ref} \end{bmatrix} \tag{6}$$

where, V_{ref} describes the reference input voltage and P_m is the mechanical power input to the machine.

3 Particle Swarm Optimization Algorithm

Among different kinds of nature-inspired algorithms, particle swarm optimization (PSO) algorithm is one of the widely used and popular optimization algorithms which is introduced by Kennedy and Eberhart. PSO inspired by the social behavior of birds swarm and fish swarm [7, 8]. Indeed, PSO is the mathematical model of the bird's behavior searching for food, for escaping from the hunters, and formate searching. In recent years, the PSO algorithm has been employed in several varieties of problems classical mathematical programming problems to scientific optimization problems and highly specialized engineering [9, 10].

The particle swarm optimization algorithm starts with an initial population (called swarm) of the search agents (called particles). These particles move around the search-space due to some mathematical models and formulas. The particle movements are pursued by their best-achieved position in the search-space as well as the swarms' best-known position [11].

After obtaining the best positions by particles, they come to guide the swarm movements. The process is repeated and by doing so it is hoped, but not guaranteed, that a promising solution will eventually be detected. The particles updating formula is as follows:

$$v_i^{t+1} = w.v_i^t + c_1 r_1 (p_i^t - x_i^t) + c_2 r_2 (g_i^t - x_i^t), \tag{7}$$

$$\begin{aligned} x_i^{t+1} &= x_i^t + v_i^{t+1} \\ i &= 1,\ 2,\ \ldots, n \end{aligned} \tag{8}$$

where, n describes the number of particles, t represents the iteration number, p_i describes the best position of the ith particle, g_i is the best particle among the group members, w represents the weighted inertia, C_1 and C_2 are the positive constants, and r_1 and r_2 are two random numbers distributed in the interval [0, 1], respectively.

By considering Eq. (5) for updating the velocity of the particle according to their previous velocity and distances to their current position from the best historical position of themselves and the best positions of the neighbors in every iteration step, and then they fly towards a new position specified by Eq. (6). The pseudo-code of a PSO Algorithm is given below.

Step 1: initializing the maximum iteration (I_{max}) and algorithm parameters.

Step 2: Initialize particles in the population.

Step 3: Evaluate the fitness value of the particles.

Step 4: Update the position of the particle-based on fitness value with p_i *and* g_i.

Step 5: If $I = I_{max}$, *go to step 7, otherwise, go to the next step.*

Step 6: Update velocity and position by Eqs. (5) and (6)

Step 7: Display the global best solution

4 Moth Search Algorithm

Nowadays, the tendency to use nature-inspired algorithms based on animal intelligence is very much appreciated, especially in the optimization techniques of these algorithms. nature-inspired algorithms such as artificial bee algorithm [12–14], firefly algorithm [15, 16], world cup optimization algorithm [17], quantum invasive weed optimization algorithm [18, 19], Variance Reduction of Gaussian Distribution algorithm give us more advantages than conventional algorithms. Optimization is the highest goal of many applications such as scientific, engineering, and industrial design. The purpose of optimization is to minimize the complexity and the costs and to maximize profit, output, performance, the efficiency of the problem.

Many nature-inspired techniques have been introduced in the last two decades. Nature-inspired algorithms have been the focus of researchers because of their simplicity, flexibility, no need to derive and escape local optimization. But there is no proof of nature-inspired techniques to solve all the optimization problems. In other words, a particular nature-inspired may show promising results for solving a series of problems, but the same algorithm may show poor performance for a number of other problems.

Based on the Phototaxis behavior, moths are attracted to fly into the light source. This phenomenon is still a mystery. There are different hypotheses for explaining this phenomenon. A reasonable model considers that the angle between the light source and moth holds on changing, but it mostly escapes from observation due to the long distance. In addition, if a moth uses a close light source for navigation, the variations in angle will be obvious. The moth flights continuously to select the best orientation so as to move towards the light source. This phenomenon leads to a spiral path for flight to get closer to the light source [20].

The second behavior in moth is the Lévy flight phenomenon. It is a widely used Non-Gaussian, heavy-tailed statistic to model the random movements. Lévy flight describes a class of random walks with Lévy distribution steps. By considering a smaller distance for the moths from the best one, they will fly around it in the form of Lévy flights, i.e. the location will be updated based on the Lévy flights process. This can be formulated as below:

$$x_i^{t+1} = x_i^t + \beta L(s) \tag{9}$$

where, i describes the number of moths, t represents the present generation, x_i^{t+1} and x_i^t are the updated and the original position of generation t, respectively, $L(s)$ describes Lévy flights step down, and β is the scale factor and is achieved as follows:

$$\beta = S_{max}/t^2 \tag{10}$$

where, S_{max} represents the max walk step.

The Lévy flights step drawn is achieved by the following equation:

$$L(s) = \frac{(\alpha - 1)\Gamma(\alpha - 1)\sin\left(\frac{\pi(\alpha-1)}{2}\right)}{\pi s^{\alpha}} \tag{11}$$

where, $s > 0$, $\alpha = 1.5$, and $\Gamma(i)$ describes the gamma function.

In some cases, moths which have a long distance from the source of light fly towards it. This motion is formulated below:

$$x_i^{t+1} = \gamma\left(x_i^t + \rho\left(x_{best}^t - x_i^t\right)\right) \tag{12}$$

where, ρ describes acceleration factor, x_{best}^t represents the best moth of the current generation (t), γ is a scale factor. Sometimes it is possible for the moth to fly towards the final position which far from the source of light. This can be formulated by the following equation:

$$x_i^{t+1} = \gamma\left(x_i^t + \frac{1}{\rho}\left(x_{best}^t - x_i^t\right)\right) \tag{13}$$

The process of the above two updating models is shown in Fig. 3.

In Fig. 4 x_i (start point), $_{best}$ (endpoint), and $x_{i,\ new}$ (light source) are the original, the best, and the updated position for moth i, respectively.

The population size and the maximum number of the iteration are set 120.

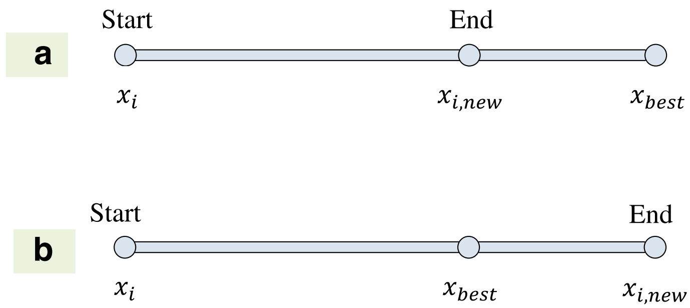


Fig. 3 A simple model of the rectilinear flight with **a** x_{best} (right) and **b** x_{best} (middle)

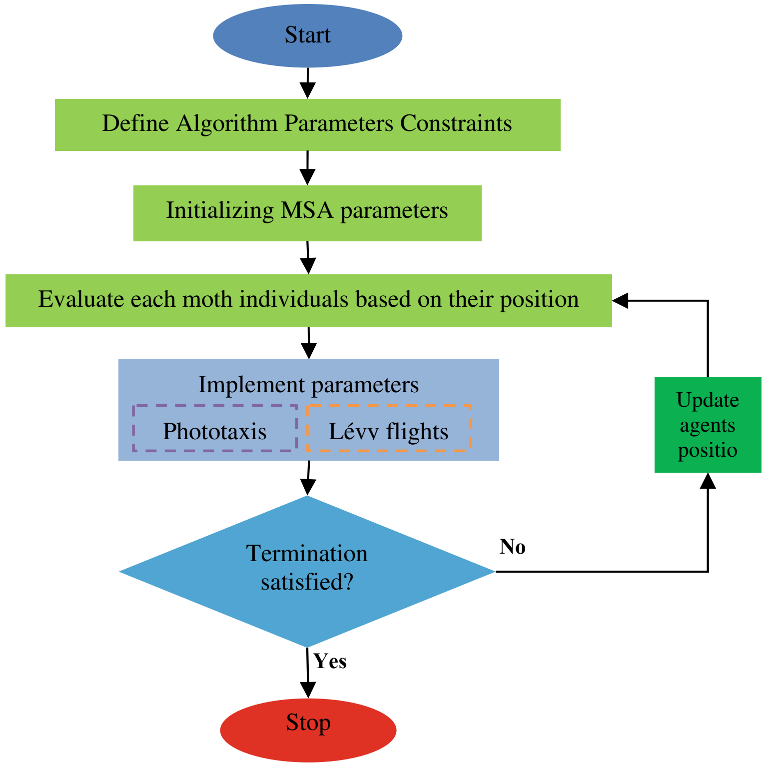


Fig. 4 The flowchart diagram of the proposed MSA

5 PID Based Power System Stabilizer

One of the widely used systems to control the low-frequency oscillations is to use power system stabilizers (PSS). This system is employed for improving the synchronous machine damping [13, 21].

Several control methods have been developed for PSS design. One of the popular methods for PSS is PID based stabilizers, because of their simple structure, high flexibility, and easy implementation. The structure of a PSS is such that its gain regulation is usually kept constant under certain conditions of power system performance. It is clear that the power system operating point changes with load changes and thus the designed system loses its sustainability. That is why classical stabilizers do not perform well in practice.

After years of research on PSSs, stabilizers were introduced, which adjusted their parameters simultaneously with changing power system operating conditions, which were known as PSSs with adaptive controllers [22–24].

However, the use of these types of controllers due to high computation time to implement the required PSS parameters as well as the difficulty of training them and sometimes requiring prior knowledge of system performance can affect the speed and accuracy of the controller performance [25].

In this study, PID parameters (k_p, k_i, k_d) will be tuned by using the MSA (and also PSO for comparison). The considered stabilizing signal for the co-ordinated controller can be obtained by the following equation:

$$U_{PSS} = [k_p + \frac{k_i}{s} + k_d s]\,\Delta\omega \,+\, \Delta u \tag{20}$$

The design starts with connecting the PID controller with the washout filter in the system matrix discussed in Eq. (5) to form the augmented A matrix as follows:

$$A = \begin{bmatrix} 0 & \omega_0 & 0 & 0 & 0 \\ \frac{-k_1}{M} & 0 & \frac{-k_2}{M} & 0 & 0 \\ \frac{-k_4}{Td0'} & 0 & \frac{-1}{k_3\,Td0'} & \frac{1}{Td0'} & 0 \\ \frac{Mk_e\left(-k_5 + {}^{Mk_i}/_{\omega_0} - k_d k_1\right)}{MT_e} & \frac{k_e k_p}{T_e} & \frac{-Mk_6 k_e - k_2 k_e k_d}{MT_e} & \frac{-1}{T_e} & \frac{k_e}{T_e} \\ \frac{{}^{Mk_i}/_{\omega_0} - k_d k_1}{MT_w} & \frac{k_p}{T_w} & \frac{-k_2 k_d}{MT_w} & 0 & \frac{-1}{T_w} \end{bmatrix} \tag{21}$$

The system data is given in the appendix.

In this study, a robust mechanism is required to develop the system damping over a wide range of operating conditions and configuration of the power system. This reason makes us provide an Eigen value-based cost function for PSS. The block diagram of the power system stabilizer is shown in the following.

The cost function of the system is given in the following [1]:

$$J = \max \ \mathrm{Re}(\lambda_i) \tag{22}$$

where, $Re\ (\lambda_i)$ is the real part of the ith electromechanical mode Eigenvalue and K_p, K_i and K_i describe the proportional, integrator and derivative coefficients of the PID controller.

Since considered Eigenvalue is extracted from $\lambda_i = \sigma_i \pm j\,\omega_i$, the final cost function and the parameter constraints are illustrated as follows:

$$\begin{aligned} & J = \max \sigma_i \\ & 0 \le K_p, K_i \le 50 \\ & 0 \le K_d \le 10 \end{aligned} \tag{23}$$

The main idea of the optimization of J is to transfer the poorly damped Eigenvalue into the D-shape in s-plane. Optimization constraints are the limitations of the optimized parameters which contain PID coefficients.

As aforementioned, in the presented study, a PID controller is adopted. In this method, the rotor speed changes are selected as the input signal of the PSS. Washout filter is also used for eliminating the controller effects in the steady-state. Indeed, the system oscillations are damped due to PSS. It generates a control signal to the AVR loop after the disturbance occurs.

6 Simulations and Results

In this study, the changes for the active power (P) and reactive power (Q) have been considered such that P = (0.1, 0.2…1) and Q = (−0.3, −0.2…1). To analyze the system reliability toward different conditions, the following operating points from Table 1 have been considered.

The optimized values for the parameters of the PID controller based on the described algorithms are given in Table 2. It is clear from Figs. 6, 7, 8, 9, 10 and 11 that the promising value for MSA has higher efficiency than the PSO.

It can be considered that the designed synchronous generator is connected to the infinite bus along with the power system stabilizer. In the presented system, there are

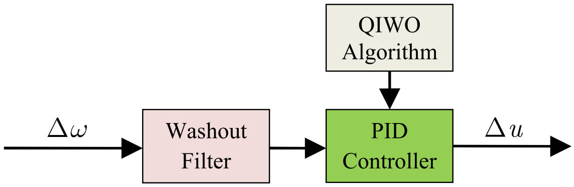
Table 1 Operating points

Case no.	P	Q
1	1	0.5
2	0.5	0.2
3	0.7	− 0.1
4	0.8	0
5	0.8	0.4
6	1	1

Table 2 Controller coefficients for the PID

Algorithm	k_P	k_I	k_D
MSA	49.13	44.53	8.12
PSO	47.26	3.83	9.32

Fig. 5 The block diagram of the power system stabilizer



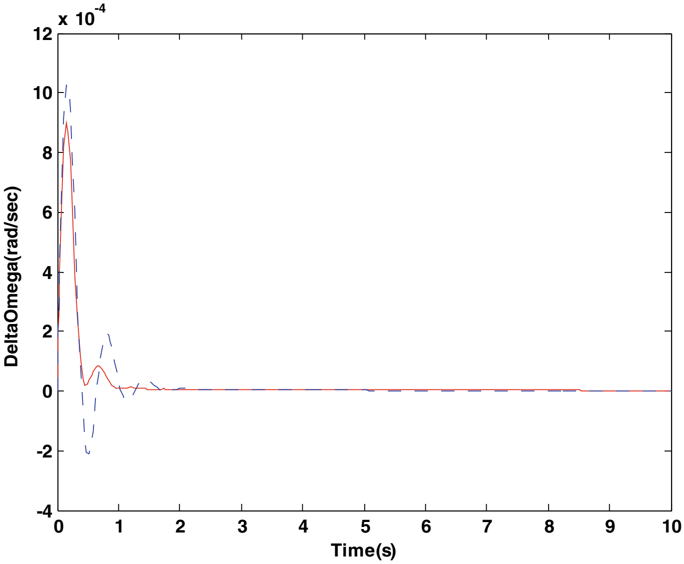


Fig. 6 Operation points: P = 1, Q = 0.5 (Solid (MSA), Dashed (PSO))

some important parameters such as electrical active power (P) and electrical reactive power (Q) for the system (operating points for synchronous generator) which can be used as the operating points to design the PID controller.

The control mechanism is applied with MSA to control the system oscillations. The output speed changes for the operating points of the system are shown in Figs. 6, 7, 8, 9, 10 and 11 (Tables 3 and 4).

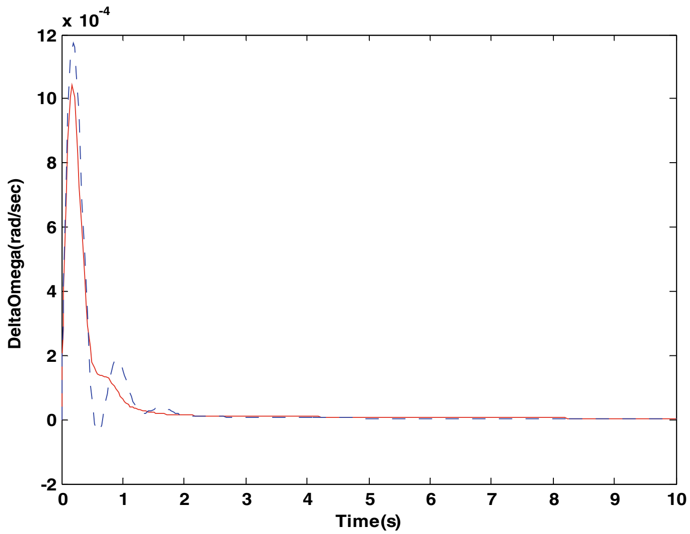


Fig. 7 Operation points: P = 0.5, Q = 0.2 (Solid (MSA), Dashed (PSO))

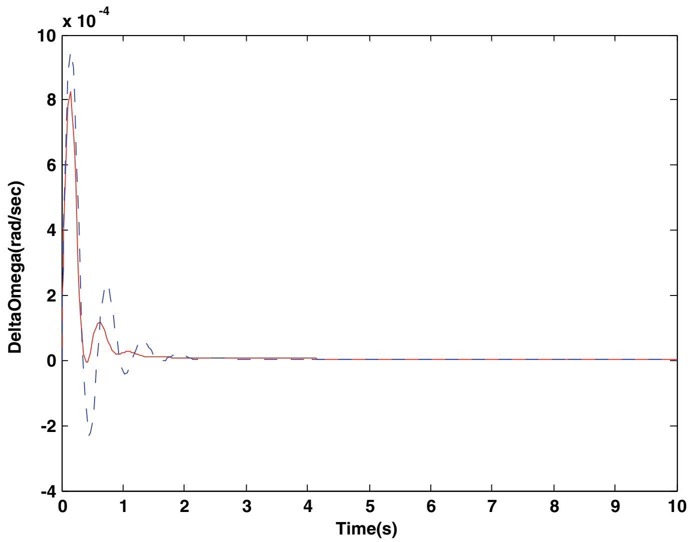


Fig. 8 Operation points: P = 0.7, Q = −0.1 (Solid (MSA), Dashed (PSO))

7 Conclusion

In this study, a new optimization algorithm called moth search algorithm (MSA) is presented to select optimal parameters of the PID controller in a power stabilizer

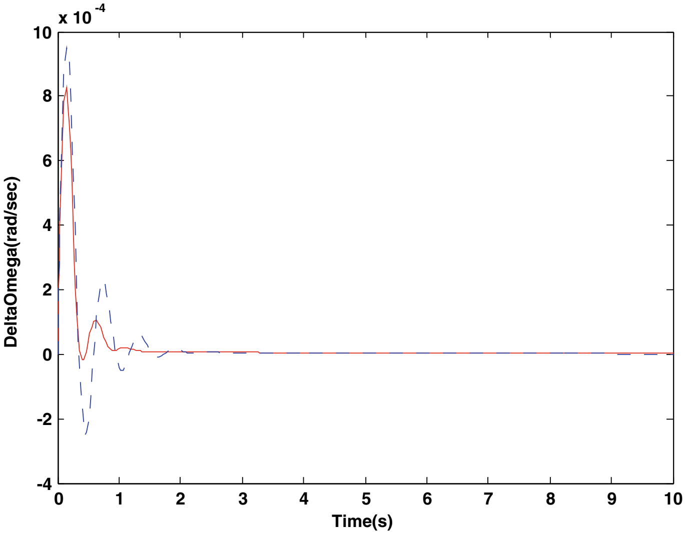


Fig. 9 Operation points: P = 0.8, Q = 0 (Solid (MSA), Dashed (PSO))

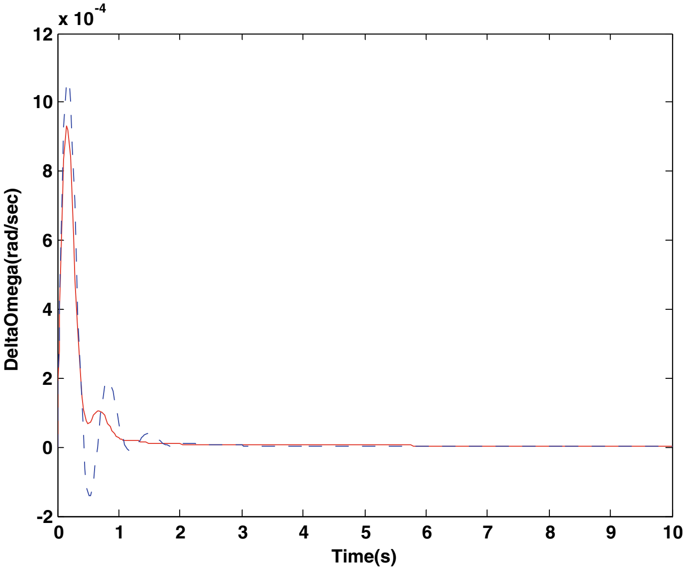


Fig. 10 Operation points: P = 0.8, Q = 0.4 (Solid (MSA), Dashed (PSO))

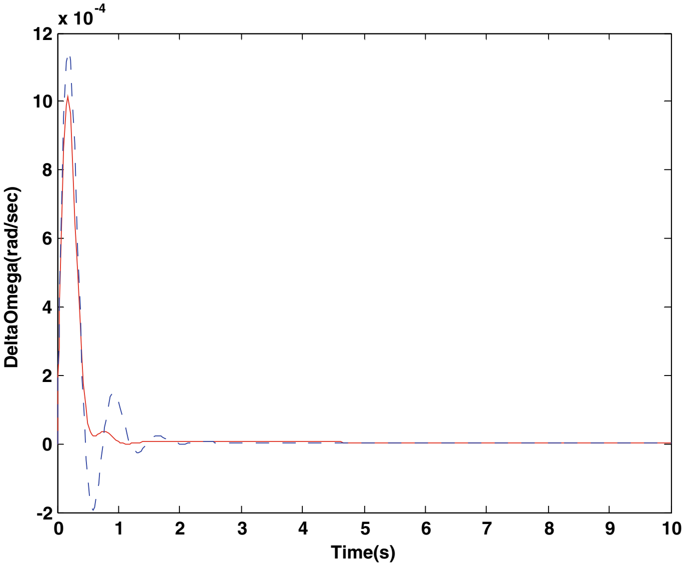


Fig. 11 Operation points: P = 1, Q = 1 (Solid (MSA), Dashed (PSO))

Table 3 Results of the response analysis in the considered operating points

Case no.	MSA		PSO	
	Over shoot	Under shoot	Over shoot	Under shoot
1	0.000861	0	0.000926	−0.0000124
2	0.001	0	0.0015	0
3	0.000815	−0.000041	0.000864	−0.0000397
4	0.0007953	−0.000035	0.0008349	−0.0000473
5	0.0009057	0	0.0009627	0

Table 4 Achieved results of the algorithms for the PSS

MSA			PSO		
Number of seeds	P_{max}	Iteration	Pop size	C1, C2	Iteration
20	15	50	50	2	20

system. The system stabilization is considered as a principal application of the algorithm. To develop the system stability and to provide good damping characteristics to electromechanical modes oscillations, a cost function based on Eigenvalue has been designed. The method efficiency is validated on a single machine infinite bus power system for stability improvement. The system efficiency of the presented dynamic MSA-PSS design is compared with the MSA-PSS design as a popular technique. The simulation results showed that the proposed MSA based method gives better solutions compared with PSO based method. It has also high convergence with less iteration. The simulation results have been analyzed on a single-machine power system and it has been found that the proposed method improved the system dynamic stability.

Appendixes

System data:

Machine (pu):

$x_d = 1.6$, $x_d{}' = 0.32$, $x_q = 1.55$.

$v_{t0} = 1.05$, $w_0 = 314$(rad/s), $Td0' = 6.0$ (s).

$D = 0$; $M = 10$.

P, Q = Electrical active and reactive power of output machine (pu).

Transmission Line (Pu):

$r_e = 0$; $x_e = 0.4$

Exciter:

$K_e = 50$; $T_e = 0.05$ (s).

Washout Filter:

$T_w = 5$ (s).

The function of k-parameters and other data are presented below:

$i_{q0} = (P * v_{t0})/\mathrm{sqrt}((P * x_q)^2 + (v_{t0}{}^2 + Q * x_q)^2)$;

$v_{d0} = i_{q0} * x_q$;

$v_{q0} = ((v_{t0}{}^2) - (v_{d0}{}^2))^{0.5}$;

$i_{d0} = (Q + xq*(i_{q0}{}^2))/v_{q0}$;

$E_{q0} = v_{q0} + i_{d0}*x_q$;

$E_0 = \mathrm{sqrt}((v_{d0} + i_{q0} * x_e)^2 + (v_{q0}\text{-}i_{d0} * x_e)^2)$;

$\mathrm{delta} = \tan^{-1}((v_{d0} + i_{q0}*x_e)/(v_{q0} - i_{d0}*x_e))$;

$K_1 = (((xq - x_d{}')/(xe + x_d{}'))*(i_{q0} * E_0 * \sin(\mathrm{delta}))) + ((E_{q0} * E_0 * \cos(\mathrm{delta}))/(x_e + x_q))$;

$K_2 = (E_0 * \sin(\mathrm{delta}))/(x_e + x_d)$;

$K_3 = (x_e + x_d{}')/(x_e + x_d)$;

$K_4 = ((x_d - x_d{}')/(x_e + x_d{}')) * (E_0 * \sin(\mathrm{delta}))$;

$K_5 = ((x_q * v_{d0} * E_0 * \cos(\mathrm{delta}))/((x_e + x_q) * v_{t0})) - ((x_d{}' * v_{d0} * E_0 * \sin(\mathrm{delta}))/((x_e + x_d{}') * v_{t0}))$;

$K_6 = (x_e * v_{q0})/((x_e + x_d{}') * v_{t0})$;

References

1. Hosseini H, Tusi B, Razmjooy N, Khalilpoor M (2011) Optimum design of PSS and SVC controller for damping low frequency oscillation (LFO). In: 2011 2nd International Conference on Control, Instrumentation and Automation (ICCIA), pp 62–67

2. Sahu PR, Hota PK, Panda S (2018) Comparison of grasshopper and whale optimization algorithm for design of FACTS controller with power system stabilizer. In: 2018 fifth international conference on parallel, distributed and grid computing (PDGC), pp 424–429
3. Razmjooy N, Khalilpour M (2015) A robust controller for power system stabilizer by using artificial bee colony algorithm. Tech J Engin App Sci 5:106–113
4. Razmjooy N, Madadi A, Ramezani M (2017) Robust control of power system stabilizer using world cup optimization algorithm. Int J Inf Secur Syst Manag 5:7
5. Chitara D, Niazi KR, Swarnkar A, Gupta N (2018) Cuckoo search optimization algorithm for designing of a multimachine power system stabilizer. IEEE Trans Ind Appl 54:3056–3065
6. Wang G-G (2018) Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems. Memetic Comput 10:151–164
7. Moallem P, Razmjooy N (2012) Optimal threshold computing in automatic image thresholding using adaptive particle swarm optimization. J Appl Res Technol 10:703–712
8. Bansal JC (2019) Particle swarm optimization. In: Evolutionary and swarm intelligence algorithms. Springer, Berlin, pp 11–23
9. Butt AA, Khan ZA, Javaid N, Chand A, Fatima A, Islam MT (2019) Optimization of response and processing time for smart societies using particle swarm optimization and Levy Walk. In: International conference on advanced information networking and applications, pp 14–25
10. de Jesus MA, Estrela VV, Saotome O, Stutz D (2018) Super-resolution via particle swarm optimization variants. In: Biologically rationalized computing techniques for image processing applications. Springer, Berlin, pp 317–337
11. AminShokravi A, Eskandar H, Derakhsh AM, Rad HN, Ghanadi A (2018) The potential application of particle swarm optimization algorithm for forecasting the air-overpressure induced by mine blasting. Eng Comput 34:277–285
12. Banharnsakun A (2018) Artificial bee colony algorithm for enhancing image edge detection. Evolv Syst 1–9
13. Razmjooy N, Khalilpour M (2015) A robust controller for power system stabilizer by using Artificial Bee Colony Algorithm
14. Sharifi S, Sedaghat M, Farhadi P, Ghadimi N, Taheri B (2017) Environmental economic dispatch using improved artificial bee colony algorithm. Evolv Syst 8:233–242
15. Moazenzadeh R, Mohammadi B, Shamshirband S, Chau K-W (2018) Coupling a firefly algorithm with support vector regression to predict evaporation in northern Iran. Eng Appl Comput Fluid Mech 12:584–597
16. Zamani MKM, Musirin I, Hassan H, Shaaya SA, Sulaiman SI, Ghani NAM et al (2018) Active and reactive power scheduling optimization using firefly algorithm to improve voltage stability under load demand variation. Indonesian J Electr Eng Comput Sci 9:365–372
17. Razmjooy N, Khalilpour M, Ramezani M (2016) A new meta-heuristic optimization algorithm inspired by FIFA world cup competitions: theory and its application in PID designing for AVR system. J Control Automat Electr Syst 27:419–440
18. Razmjooy N, Ramezani M (2014) An improved quantum evolutionary algorithm based on invasive weed optimization. Indian J Sci Res 4:413–422
19. Namadchian A, Ramezani M, Razmjooy N (2016) A New Meta-heuristic algorithm for optimization based on variance reduction of Guassian distribution. Majlesi J Electr Eng 10:49
20. Callahan PS (1977) Moth and candle: the candle flame as a sexual mimic of the coded infrared wavelengths from a moth sex scent (pheromone). Appl Opt 16:3089–3097
21. Mohammadi RS, Mehdizadeh A, Kalantari NT (2017) Applying TID-PSS to Enhance Dynamic Stability of Multi-Machine Power Systems. Trans Electr Electron Mater (TEEM) 18:287–297
22. Hemmati R (2018) Power system stabilizer design based on optimal model reference adaptive system. Ain Shams Eng J 9:311–318
23. Farhad Z, Ibrahim E, Tezcan SS, Safi SJ (2018) A robust PID power system stabilizer design of single machine infinite bus system using firefly algorithm. Gazi Univ J Sci 31:155–172
24. Moutis P, Amini H, Khan IA, He G, Mohammadi J, Kar S et al (2019) A survey of recent developments and requirements for modern power system control. In: Pathways to a smarter power system. Elsevier, pp 289–315

25. Hossain SJ, Bhattarai R, Yousefian R, Kamalasadan S (2018) Adaptive wide area damping controller for distributed energy resources integrated power grid. In: 2018 IEEE Power & Energy Society General Meeting (PESGM), pp 1–5

Design of Self-adaptive Fuzzy Logic Droop Controller for Hybrid Units in Islanded Microgrids

V. S. Vakula and B. Damodar Rao

Abstract In the present paper, a Multi-Segment P/f Droop Control Strategy is developed to achieve localized control, coordination and power management of different sources in an islanded microgrid which consists of two PV/Battery storage Hybrid units and a droop unit. Separate control loops are designed, each for PV power production control, Battery charging and discharging level control, and Droop unit operation control based on multi-segment P/f Droop characteristics. The conventional PI controllers can be used in the control loop design as they are very simple for implementation and give a better dynamic response. But their performance deteriorates when the complexity in the system increases due to disturbances like load variations and intermittent sources. Fuzzy Logic Control (FLC) technique is model-independent and a flexible tool. It can deal with complex systems such as microgrids with different types of imprecise inputs and variables particularly if the power is supplied by intermittent sources and is also consumed by varying and unpredictable loads during sudden disturbances. The Membership functions (MF) of a conventional FLC are determined by trial and error method. This method is time-consuming and does not guarantee an optimal controller. In the present paper, a Self-Adaptive Fuzzy Logic Droop Controller (SAFLDC) is proposed, whose input–output MFs are made adaptive by obtaining the cluster centers using Self-Organizing Maps (SOM) Algorithm. The paper aims to follow a systematic approach to propose an effective SAFLDC with optimal MFs whose parameters are tuned using Hybrid GA-PSO. The developed strategy has been validated using detailed switching models in MATLAB/SIMULINK tool and the behavior of key variables in the strategy is illustrated.

Keywords Microgrid · Droop control · PV · Battery storage · Hybrid unit · Droop unit · PI controller · FLC · SOM · SAFLDC · Hybrid GA-PSO

V. S. Vakula (✉) · B. D. Rao
Department of Electrical and Electronics Engineering, JNTUK UCEV, Vizianagaram, Andhra Pradesh, India
e-mail: dr.vakulavs.jntu@gmail.com

B. D. Rao
e-mail: bendidamodar27@gmail.com

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_11

1 Introduction

Solar energy is clean and green energy and it can be converted to electrical energy using Photovoltaic (PV) cells. PV power production is of non-continuous nature because it depends on solar irradiance and temperature at the installed location. PV arrays need the support of battery storage to increase system reliability and effectivity by regulating load voltage [1]. PV unit in addition to battery storage is collectively referred to as a Hybrid unit. The combination of hybrid units, dispatchable droop units and local loads constitute a Microgrid. Droop units can ensure continuity of supply, voltage regulation and frequency regulation. Therefore, comprehensive coordination between all the hybrid units, and droop units is required for the effective and reliable operation of a microgrid. Thus, for proper coordination, the following three issues (i) PV array and battery operation during the failure of droop units, (ii) battery operation only at peak loads and (iii) load sharing nature of all units, has to be resolved. This coordination problem can be solved with the application of centralized control strategies that need effective communications between each hybrid unit, droop unit and a centralized energy management system (EMS) [2, 3]. Any communication failure affects microgrid operation because some units, at which the failure occurs, may become invisible to the EMS [4]. To avoid dependence on communications and centralized management of units, decentralized power management for islanded microgrids can be used [5–12]. The control strategies used in [5–7] are limited to microgrids where the frequency is strictly regulated by a single battery unit. The strategy in [8] has focused on the external power flow between the PV/battery unit and the microgrid without considering dedicated controllers for the DC-DC converters and the PV curtailment.

In [9], separate controllers were developed for the PV source and battery, instead of including into a single hybrid unit. Moreover, the control strategies presented in [5–9], have not resolved coordination in the presence of droop controlled units. But it is more important to consider the droop controlled units to improve the performance of PV and battery systems in widely adopted droop controlled microgrids. Du et al. [10] proposed a strategy without management of battery storage and [11–13] employed adaptive P/f characteristics at each unit but the application of the proposed strategies is specifically limited to a single PV unit with a single battery unit [11].

The utilization of adaptive slope droop control for PV units was proposed in [12] to provide fully autonomous power management of PV units in islanded microgrids. However, the strategy in [12] may result in unnecessary discharging of the battery at low levels of PV power production.

Hybrid PV/battery systems provide more flexibility in power management as the local control system has access to the battery state of charge (SOC) estimate, as well as the PV power measurements. Hybrid units [13–15] in a combination of PV, battery, and other energy sources require a central control system [13] to coordinate the hybrid unit operation with a diesel generator to maintain the power balance in the microgrid under certain operating scenarios. On the other hand, only a single hybrid unit which is used in [14, 15], made the system controlled standalone power supply.

These drawbacks are eliminated in [16] by designing localized control units at each Hybrid unit based on P/f droop control strategy. The performance of the control unit with the use of conventional PI controllers deteriorates when the complexity of the system increases due to wide disturbances like load variations. It does not give a smooth transition between step changes of load and there are considerable fluctuations in power and voltage for every step change of load.

Fuzzy Logic Control (FLC) is a flexible tool that can deal with complex systems such as microgrids with different types of imprecise inputs, variables particularly if the power is supplied by intermittent sources and is also consumed by varying and unpredictable load during sudden disturbances. FLC [17] proposed individual controls for battery unit only to keep the SOC and power within its maximum and minimum limits regardless of variation in load and intermittent nature of PV sources in an islanded microgrid. Further, the PV source and battery are controlled separately and are not taken as a single hybrid unit. Generally, the membership functions (MF) of a conventional FLC are determined by trial and error method. This method is time-consuming and does not guarantee an optimal controller.

In the present paper, an islanded microgrid consisting of PV/battery hybrid systems and droop units along with a control system for voltage source converter (VSC) is adopted from [16]. The PI controller used in the reference generator of PWM in the VSC controller [16] is replaced with a Fuzzy Logic Controller (FLC). The major advantage of the fuzzy logic controller is its model independency. In the present paper, a Self-Adaptive Fuzzy Logic Droop Controller (SAFLDC) is proposed, whose input–output MFs are made adaptive by obtaining the cluster centers using Self-organizing maps Algorithm. The inputs to proposed SAFLDC are error, E $(V_{ref} - f_{ref})$, change in error, CE $\left(\Delta\left(V_{ref} - f_{ref}\right)\right)$ to obtain the required output (V_{dq_ref}).

The evolutionary computing techniques have exposed a significant interest in optimization for many years. Where exact and analytical methods cannot be applied, global optimization algorithms are the main concern for finding optimum solutions through evolutionary programming (EP) [18] and genetic algorithms (GA) [19]. Recently, particle swarm optimization (PSO) [20] and Differential Evolution Algorithm (DEA) [21] have been introduced and PSO has received increased interest among them.

In the present work, the Hybrid GA-PSO is used to tune the parameters of the PI controller [22]. This algorithm is a combination of GA and PSO and utilizes features of both the algorithms. This was developed to overcome the limitations of GA and PSO. Thus the present work concentrates on designing a Hybrid SAFLDC with self-organized membership functions in which its parameters are tuned by using hybrid GA-PSO.

This paper is organized into five sections as follows. System Overview and Droop Control Strategy is presented in Sect. 2. The proposed control strategy is presented in Sect. 3. Simulation results that validate the efficacy of the proposed control strategy are presented in Sect. 4, followed by conclusions from the presented work in Sect. 5.

2 System Overview and Droop Control Strategy

The islanded microgrid system with two PV/Battery hybrid units and one droop unit considered in this paper is as shown in Fig. 1. Figure 2 shows the PV/battery hybrid system with the proposed control strategy. PV unit consists of a unidirectional DC/DC converter that controls the PV array output voltage to achieve Maximum Power Point Tracking (MPPT) in all instants. Bidirectional DC/DC converter of battery regulates the DC link voltage. The Droop unit used here is a constant DC source. A DC/AC inverter, i.e., Voltage source converter (VSC) is provided at the droop unit to control the power generation of it according to the requirements of the microgrid bus (Fig. 3).

The control system of the VSC is operated as a voltage source that follows the multi-segment P/f droop characteristic curve, Fig. 4 [16]. The P/f characteristics are divided into three segments in the frequency range of nominal frequency (f_o) and minimum frequency (f_{min}) of the microgrid. The segments are PV Segment (f_{pv}), Droop segment (f_D), and Battery segment (f_B), which are defined over the ranges of [f_o, f_D], [f_D, f_B], and [f_B, f_{min}], respectively. The proposed P/f characteristics are chosen such that the main objective of the hybrid unit is to charge the battery whenever possible, and to discharge when all units reach their generation limits, or

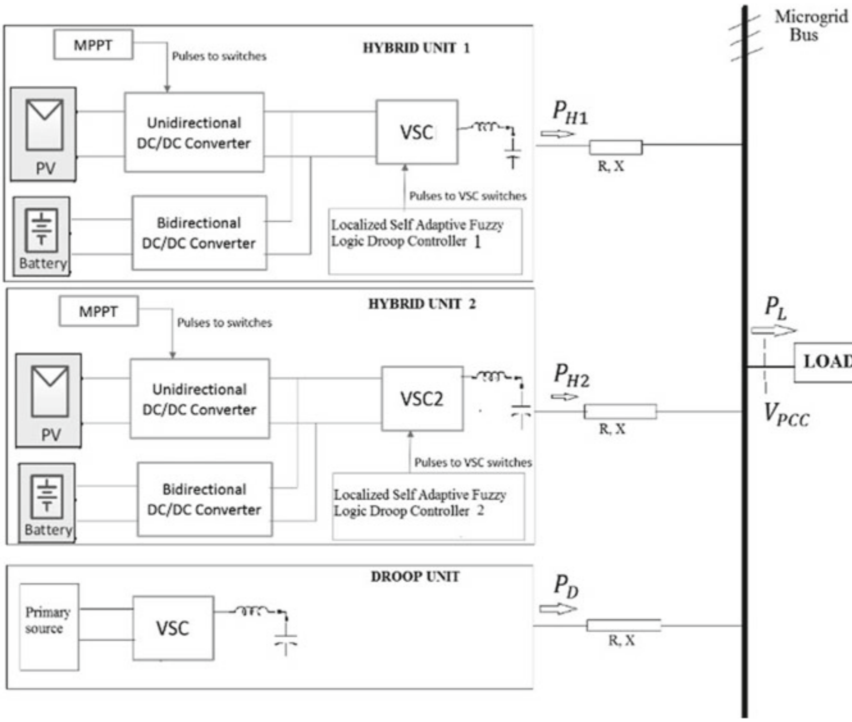


Fig. 1 Microgrid structure

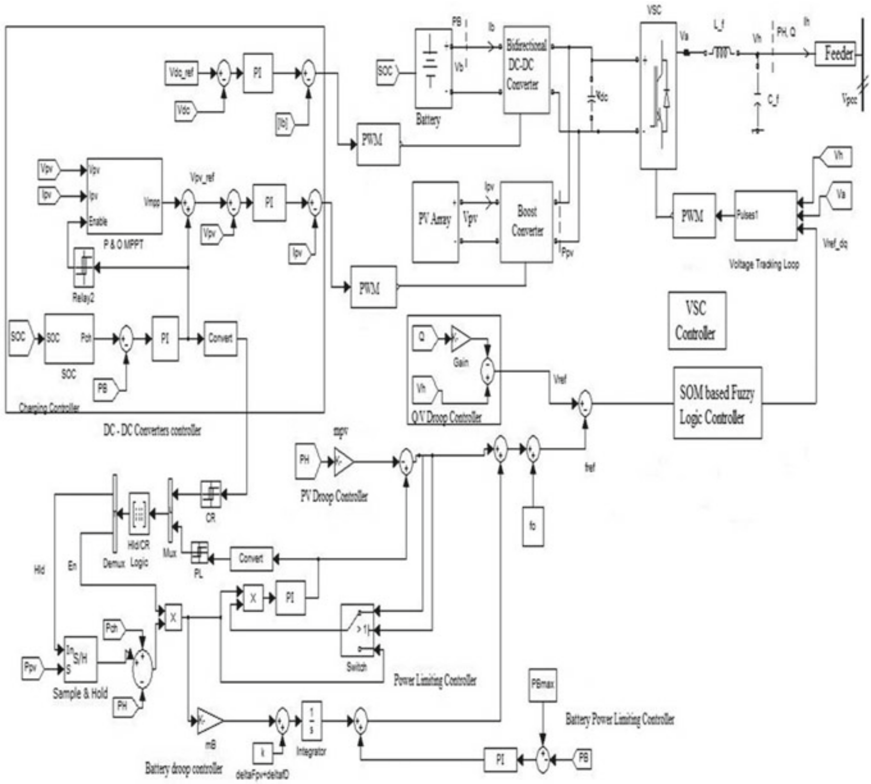


Fig. 2 The architecture of the proposed control system for PV/battery hybrid unit

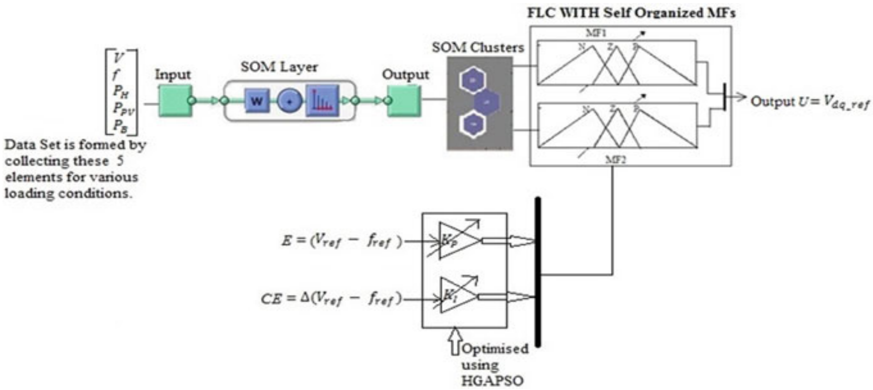


Fig. 3 The proposed structure of hybrid SOM based fuzzy logic controller

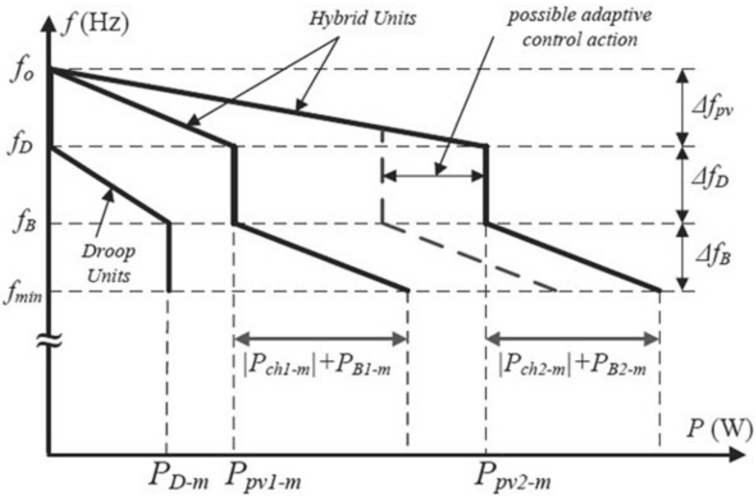


Fig. 4 General structure of the proposed multi-segment P/f characteristics [16]

during low PV production and peak load periods [16]. Also, the priority of charging should be given first to that battery which has lower SOC.

The adaptive change in the P/f characteristics, and the transition of the P/f operating point along with these characteristics, is determined by control loops based on three variables: power supplied by the hybrid unit to the microgrid (P_H), charging power reference (P_{ch}) of the batteries, and power produced by the PV array (P_{PV}) [16].

P_{PV} is equal to the maximum available power only when the PV array operates at the maximum power point (MPP). The reference P_{ch} is generated by the Charging controller, as a function of the battery SOC. A charging curve similar to that proposed in [16] and shown in Fig. 5 is adopted here.

3 Design of Self-adaptive Fuzzy Logic Droop Controller

The parameters of the controllers used in the proposed control strategy are to be tuned such that effective autonomous coordination is achieved between multiple PV/Battery hybrid units and droop units in an islanded microgrid. In this paper, a Self-Adaptive Fuzzy Logic Droop Controller (SAFLDC) is proposed. The input–output MFs of SAFLDC are made adaptive by obtaining the cluster centers using the Self-organizing maps algorithm. The inputs to proposed SAFLDC are defined as an error ($\text{E} = V_{ref} - f_{ref}$), change in error $\left(\text{CE} = \Delta\left(V_{ref} - f_{ref}\right)\right)$ to obtain the required output ($\text{U} = V_{dq\_ref}$).

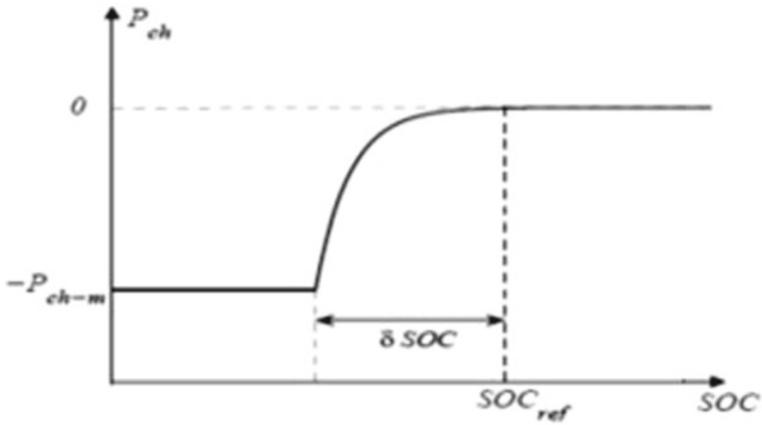


Fig. 5 Battery charging curve [16]

3.1 Fuzzy Logic Controller

The fuzzy logic control technique has been a good replacement for conventional control techniques, which require highly complicated models. Fuzzy logic controllers are first implemented by King and Mamdani [23] on the basis of the fuzzy logic system which is generalized from the fuzzy set theory. The theory of fuzzy sets was introduced by Zadeh [24]. It has appeared to offer a feasible solution to various control problems. The fuzzy sets have been applied to different areas of science. Its use in engineering disciplines has been widely spread such that commercial and industrial fuzzy systems have been successfully developed in the last few years. The main attraction undoubtedly lies in the unique characteristics of fuzzy systems. They are capable of handling complex, nonlinear, and sometimes mathematically intangible dynamic systems using simple solutions [25].

The use of the concept of fuzzy logic control in the real-time application was started by Mamdani and Assilian in 1975. These controllers have the potential for robust control in the face of a system parameter and load uncertainties. In general, Fuzzy Logic Controllers (FLC) are suitable for plants that cannot be described precisely by a mathematical formulation. It is realized that fuzzy logic control can perform very effectively when the operating conditions change rapidly and also when the system nonlinearities are significant. These features make up very attractive for power system applications since the power system is a highly non-linear and chaotic system [26]. Conventional controllers designed for one operating condition are not effective for strong non-linear systems and over a wide range of operating conditions.

3.1.1 Design of Fuzzy Logic Controller (FLC)

Figure 6 shows the basic structure of the Fuzzy Logic Controller. The steps involved in the design of a fuzzy logic control system using Fuzzy Inference System are as follows [27].

1. Identify the inputs and outputs using linguistic variables (Fuzzification).
2. Assign membership functions to the selected variables (Inference mechanism).
3. Frame the rule base (Knowledgebase).
4. Calculate the crispy control action (Defuzzification).

The linguistic variables, membership functions, and rule base can be suitably developed from the experience. A large rule base gives smooth control; however, it increases the control complexity. It is, therefore required to keep the rule base reasonably sized.

1. **Fuzzification**
   Different input and output variables, their numerical range and linguistic variables for membership functions are to be selected. The first operation to be performed is fuzzification. It involves transferring of the range of the inputs and output of the FLC into their corresponding Universe of Discourse (UOD). The second operation is to divide the respective inputs into suitable linguistic variables. The parameters of the fuzzification module depend on the shape of the membership functions (MF). Triangular and trapezoidal MFs are selected for simplicity [27].
2. **Fuzzy Inference Mechanism**
   The interface mechanism plays a vital role in designing FLC. The membership values, obtained in the Fuzzification step, are combined to obtain the firing strength of each rule. Each rule characterizes the control goal and control policy of the domain experts by means of a set of linguistic control rules. Then depending on firing strength, the consequent part of each qualified rule is generated. The most commonly used interface mechanisms are 1. Mamdani type and 2. Takagi–Sugeno type.
3. **Knowledgebase**

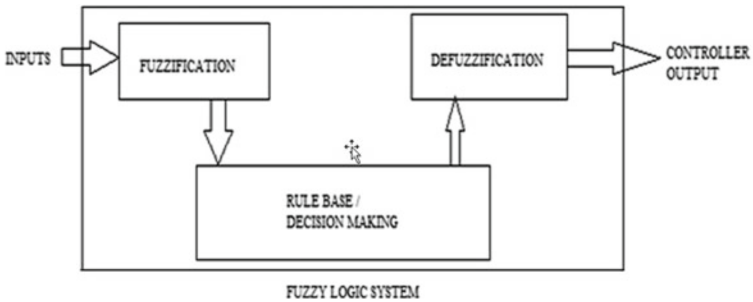


Fig. 6 Structure of a fuzzy logic controller [27]

The knowledge base of an FLC consists of a database, whose basic function is to provide, the necessary information for the proper functioning of the fuzzification module, the inference engine, and the de-fuzzification module.
The necessary information includes:

- Fuzzy sets (membership) representing the meaning of the linguistic values of the process state and the control output variables.
- Physical domains and their normalized counter-parts together with the normalization (scaling) factors.

4. **Defuzzification**
The following are the functions of the Defuzzification:

- It converts the set of modified control output values into a non-fuzzy control action.
- It performs an output de-normalization which maps the range of values of fuzzy sets to the physical domain.

The commonly used strategies for defuzzification [27] are

(a) Min-Max method: OR operator is used on the output of each rule to obtain the final output as shown below:

$$\mu_{out} = \mu_{MF1}(.)\ \mathrm{OR}\ \mu_{MF2}(.)\ \mathrm{OR}\ \mu_{MF3}(.)$$

(b) The center of areas: Strategy generates the center of gravity of the possibility distribution of control action. In the case of a discrete domain, this method defers. Using the centroid method defuzzified output is found using Eq. (1) [28].

$$U = \frac{\sum \left\{ \begin{array}{c} Membership\ value\ of\ input * output\ corresponding\ to\ the \\ membership\ value\ of\ input \end{array} \right\}}{\sum \{membership\ value\ of\ input\}}$$

$$U = \frac{\sum V(A_i, B_i)}{\sum \mu(A_i, B_i)} \tag{1}$$

(c) Height or Mean of maximum method: It is the simplified form of a center of gravity method, where the only highest point at the center of each MF is considered for finding the crispy output.

FLC Fuzzification Module

Two variables error (e = $E \cdot K_p$) and change in error (ce = $CE \cdot K_i$) are used as input signals. The output signal is U. K_p and K_i transform the scaled real values to the desired value in the decision limit and are tuned using Hybrid GAPSO.

The three similar fuzzy sets defining the two inputs of the FLC are given by Eq. (2). The MF of these is defined by $\mu_{\rm p}(.)$, $\mu_{\rm N}(.)$ and $\mu_{\rm Z}(.)$ or $\mu_1(.)$ $\mu_{-1}(.)$ and $\mu_0(.)$ [21].

$$error\ (e) = \{N\ (Negative),\ Z\ (Zero),\ P\ (Positive)\} \tag{2}$$

Similarly for change in error (*ce*).

Let the number of fuzzy sets of the inputs and their MF be identical. If the members of the input fuzzy set N, Z and P, are X_{-1} (.), X_0 (.), X_1 (.) respectively, then the output function is derived using the control rules, where i, j and k can take any value from −1, 0, +1.

The fuzzy control rule design involves writing rules that relate the input variables to the output model properties. These rules are expressed in an English like language with the syntax [22] such as

IF {error 'e' is x_i and change in error 'ce' is x_j} THEN {control output is $U_{-(i+j+k)}$}.

The above fuzzy rule is called a linear control rule because the linear function is employed to relate the indices of the input fuzzy sets to the index of the output fuzzy set. Based on this concept the rules are framed. Such a group of rules forms a fuzzy control rule base.

FLC Rule Module

The number of membership functions of the input fuzzy is taken as 3, and are defined as {N (Negative), Z (Zero), P (Positive)}. The number of membership functions of the output fuzzy is taken as 5, and are defined as U = {NL (negative large), NS (negative small), Z (Zero), PS (positive small), PL (positive large)}.

These membership functions are considered and partitioned within the UOD (Universe of Discourse) in the desired range for the outputs. The decisions in a fuzzy logic-based approach are made by forming a series of rules which relate the inputs to outputs by IF-THEN statements.

FLC Defuzzification Module

To achieve crisp numerical values, the input variables are related to the output variable, the fuzzy results are defuzzified using the Centroid method which is called Defuzzification Process. Using the Centroid method the defuzzified output is found using Eq. (3) [28].

Table 1 Rules for two input—three membership functions

Rule	E (i/p)	CE (i/p 2)	o/p
1	P	P	PL
2	P	N	NS
3	P	Z	PS
4	N	P	NS
5	N	N	NL
6	N	Z	NS
7	Z	P	PS
8	Z	N	NS
9	Z	Z	Z

$$U = \frac{\sum_{i,\,j,\,k=1}^{n} v(i,\,j,\,k)}{\sum_{i,\,j,\,k=1}^{n} \mu(i,\,j,\,k)} \tag{3}$$

where i, j, k are the ith, jth, kth membership functions of inputs respectively. The output corresponding to the membership value of a particular input is found in Eq. (4), if the input does not lie in the overlapping region of two membership functions. Otherwise, it is found as in Eq. (5) [28],

$$\mu_U = Min\left[\mu_i(e),\, \mu_j(ce)\right] \tag{4}$$

$$\mu_U = Max[\mu_{U1},\, \mu_{U2}] \tag{5}$$

where, μ_{U1}, μ_{U2} are the output membership values of input in region 1 and region 2 respectively and $v(i,\,j,\,k)$ in Eq. (3) is the incremental control output contributed by any fuzzy control rule in Table 1. Zadeh fuzzy logic 'and' is used to execute the IF side of the fuzzy control rule that is

$$\mu(i,\,j) = Min\left[\mu_i(e),\, \mu_j(ce)\right] \tag{6}$$

The Fuzzy Logic Toolbox provides tools for the user to create and edit fuzzy inference systems within the framework of MATLAB. Users can also integrate fuzzy systems into simulations with Simulink models.

3.1.2 Characteristics and Limitations of FLC

Characteristics

The advantages of using fuzzy control usually fall into one of the following categories

- Robust nonlinear control—The most attractive feature of a fuzzy logic controller is that it generally has a non-linear transfer function.
- For the conventional Proportional Integral (PI) controllers, a substantial parameter changes for a major external disturbance. In the presence of such a disturbance, the conventional PI controller fails. In this case, the fuzzy controller offers to implement simple but robust solutions that convert a wide range of system parameters and can cope with major disturbances.
- Fuzzy control offers a method of implementing an expert's Knowledge.

Limitations

Although FLC introduces a good tool to deal with complicated, nonlinear and ill-defined systems, it suffers from the following drawbacks.

- The black box approach of FLC makes the design procedure of FLC complicated.
- The proper decision rules cannot easily be derived by human expertise for the too complex control system.
- Rule base formation for large nonlinear systems requires knowledge and is time-consuming.

3.2 Design of SOM Based FLC

In recent years, the fuzzy modeling technique has become an active research area due to its successful application to the complex system model, where classical methods such as mathematical and model-free methods are difficult to apply because of lack of sufficient knowledge. The Fuzzy Logic model is empirically based and relying on an operator's experience rather than their technical understanding of the system.

For a little more complex system, but for which significant data exist, model-free methods such as neural networks provide a powerful and robust means to reduce some uncertainty through learning, based on patterns in the available data. For the most complex system where few numerical data exist and only ambiguous or imprecise information may be available, fuzzy reasoning provides a way to understand system behavior by allowing us to interpolate approximately between observed input and output situation [28]. The imprecision in fuzzy models is therefore, generally quite high. Fuzzy systems can implement crisp input and output, and produce a non-linear functional mapping. Depending on the system, it may not be necessary to evaluate every possible input combination since some may rarely or never occur. This will increase the number of fuzzy rules and complexity but may also increase the quality of the control. Obtaining an optimal set of membership functions and rules for FLC is not an easy task. In most fuzzy control systems, the membership functions and fuzzy rules are derived and tuned by human experts. It requires time, experience, and skills of the operator for the tedious fuzzy design and tuning exercise, and the obtained FLCs may not be optimal [25].

Recently, there has been an extensive amount concentrated on the membership function development and the rule construction in FLC design. Clustering algorithms are used in the design of an FLC by optimizing the fuzzy sets and the fuzzy control rules from the operating data of the controlled plant or system. Clustering algorithms permit the classification of the data in distinct groups using distance and/or similarity functions. These groups can later be used directly in selecting appropriate fuzzy set boundaries. Also, the algorithms can automatically combine similar objects (data entries) in order to reduce the global size of the data. Finally, the clustering algorithms let us easily detect potential outliers (clusters containing one or very few data entries). This feature is taken into consideration to design a fuzzy controller.

In the design process of the FLC, fuzzy rules are defined for two input variables, the error (e) and the change in error (ce), and a single output variable. With the PI controller operate the system in steps over its full range of operation and at each step, record the values of the variables of the system control units [29].

3.2.1 Self-organizing Maps (SOM)

The combination of the theory of neural networks and fuzzy sets are termed as Neuro-fuzzy systems [30]. Kohonen's Self-Organizing Map is one of the best-known neural network models. It is also known as Kohonen Neural Network. SOM could be applied to a number of approaches [31].

In today's data-driven world, it has become increasingly important to analyze huge amounts of data for extracting information from it. There are two types of such data analysis namely exploratory and confirmatory [32].

One of the dominant techniques of exploratory data analysis is Cluster analysis. It aims to group a collection of objects into clusters. In the process of grouping, those samples which are more closely related to one another are assigned to different clusters. Clustering methods are classified as hierarchical and partitioning clustering. Partitioning clustering method is again classified as K-Means clustering and Self-Organizing Maps where the former is an iterative descent clustering algorithm and the latter is a constrained version of K-Means clustering. SOMs have the advantage that it is possible to easily displace the output as a 2-D grid of samples [32].

The basic idea of a SOM is to map the data patterns onto an n-dimensional grid of neurons or units. That grid forms what is known as the output space, as opposed to the input space where the data patterns are. This mapping tries to preserve topological relations, i.e., patterns that are close in the input space will be mapped to units that are close in the output space, and vice-versa. So as to allow an easy visualization, the output space is usually 1 or 2 dimensional [33].

SOM is less prone to local optima [31]. The search space is better explored by SOM. This is due to the effect of the neighborhood parameter which forces units to move according to each other in the early stages of the process [33]. This is a technique that reduces the dimensions of large data with the use of self-organizing neural networks. SOM training happens using unsupervised learning to produce a two-dimensional, discretized representation of the input space of the training samples. The trained samples are called maps. The way of representing the multi-dimensional data in one or two-dimensional spaces is called vector quantization. This is a process of reducing the dimensionality of vectors and a data compression technique [31].

In unsupervised learning, the training of the network is entirely data-driven and no target results are provided for the input data vector. The unsupervised learning type, such as SOM, is used for clustering the input data. SOMs operate in two modes: (i) training: builds the map using input examples, (ii) mapping: mapping automatically classifies a new input vector. The network is trained with the SOM batch algorithm [33].

SOM Algorithm for Clustering

Let the number of samples in the large data be ‘i’ and the number of variables in each sample is ‘j’ and clusters are termed as ‘a, b, c, etc.’

Step 1: Initialize cluster centroids by random selection from the samples.

Step 2: Finding of Winning Cluster.

Consider a sample ‘i’. Calculate the distance from each cluster ‘a’ to the sample ‘i’. The winning cluster is that which has a minimum Euclidean distance from the selected sample. The considered sample goes to the winning cluster.

$$d_{ai} = \left\{ \sum_{j=1}^{n} \left(x_{aj} - x_{ij} \right)^2 \right\}^{1/p} \tag{7}$$

where, p = 2 for Euclidean distance and x is the value of the variable in a sample.

Step 3: Updating of Centroid values.

Each variable in the cluster is updated using Eq. (8)

$$G_{aj}^{|} = \frac{x_{aj} + x_{ij}}{2} \tag{8}$$

Step 4: Repeat step 2 and step 3 for all the samples from i = 1 to m.

Step 5: Finally we will get all the updated values of cluster centroids.

3.2.2 SOM Based FLC

In the present work, self-tuned adaptive MFs are designed, using the SOM clustering technique [34]. Inputs to SOM based FLC are error (e = $E{\cdot}K_p$) and change in error (ce = $CE{\cdot}K_i$) where,

$$E = V_{ref} - f_{ref} \tag{9}$$

$$CE = \Delta\left(V_{ref} - f_{ref}\right) \tag{10}$$

To implement SOM clustering, the samples of Microgrid voltage, frequency, Hybrid unit power, Battery power and PV array power are collected at different loading conditions on the test system. These samples are clustered into an equal number of clusters and MFs for each input variable. From the clustered samples, each cluster results in a minimum value, maximum value and its cluster center. Self-tuned MFs are formulated such that minimum, center and maximum values give the lower, center and upper values of the triangular MF respectively. With these values, the triangular MFs are formulated. The MF's of the inputs are given in Eqs. (11) and (12).

$$e = \{N\,(Negative),\ Z\,(Zero),\ P\,(Positive)\} \tag{11}$$

$$ce = \{N\,(Negative),\ Z\,(Zero),\ P\,(Positive)\} \tag{12}$$

The relationship between the two inputs (*e*, *ce*) of fuzzy controller is derived, by dividing the two inputs into intervals defined by Cluster centers formed by the SOM algorithm. Each input has three triangular membership functions. The triangular membership functions (N, Z, P) as in Fig. 7a partitioned within the Universe of discourse in the range [−0.5, +0.5]. The number of membership functions of the output ($U = V_{dq\_ref}$) are taken as 5, and are defined as follows:

$$U = \{NL, NS, Z, PS, PL\} \tag{13}$$

The membership functions of output are considered as in Fig. 7b and are partitioned within the UOD in the range [−0.5, +0.5]. The decisions in a fuzzy logic-based approach are made by forming a series of rules, which relate the inputs to outputs by IF-THEN statements. In this case, the number of control rules to cover all the possible combinations of the three membership functions of each input variable are 3 × 3 (9). These rules are composed as in Table 1 for the proposed and designed SAFLDC.

Defuzzification Process is performed to achieve crisp numerical values for the formation of Fuzzy rules. Using the Centroid method the defuzzified output is found using Eq. (3). The output corresponding to the membership value of a particular input is found as in Eqs. (4) and (5). After Defuzzification for each combination of

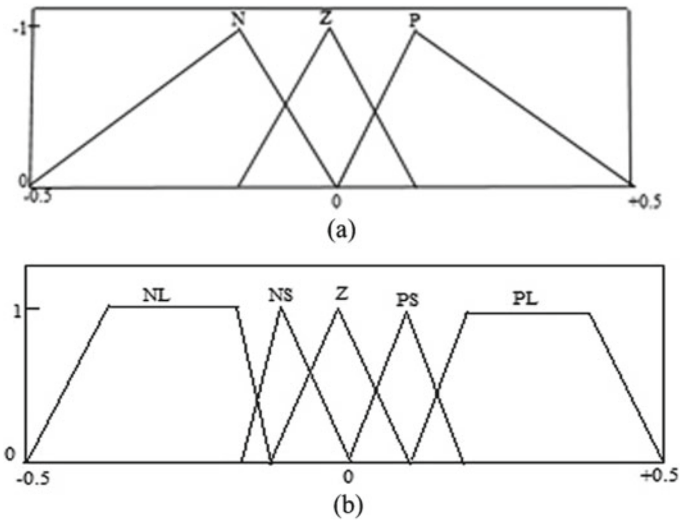


Fig. 7 **a** Triangular MFs for inputs—error (E) and change in error (CE), **b** output MFs

MFs of the two inputs, the corresponding MFs of the output are calculated and are shown in Table 1.

The complete step by step flow of operation of the proposed SAFLDC is given in Fig. 8.

3.2.3 Optimal Parameter Tuning for the Proposed SAFLDC

The evolutionary computing techniques have exposed a significant interest in optimization for many years. Where an exact and analytical methods cannot be applied, global optimization algorithms are the main concern for finding optimum solutions through Evolutionary Programming (EP) [18] and Genetic Algorithms (GA) [19]. Recently, Particle Swarm Optimization (PSO) [20] and Differential Evolution Algorithm (DEA) [21] have been introduced and PSO has received increased interest among them.

Genetic Algorithm

Genetic algorithm (GA) is an optimization method based on the mechanics of natural selection and natural genetics. Its fundamental principle is that the fittest member of the population has the highest probability for survival. GA is a parallel and global search technique. The GA works with objective function information in a search for the optimal parameter set [19].

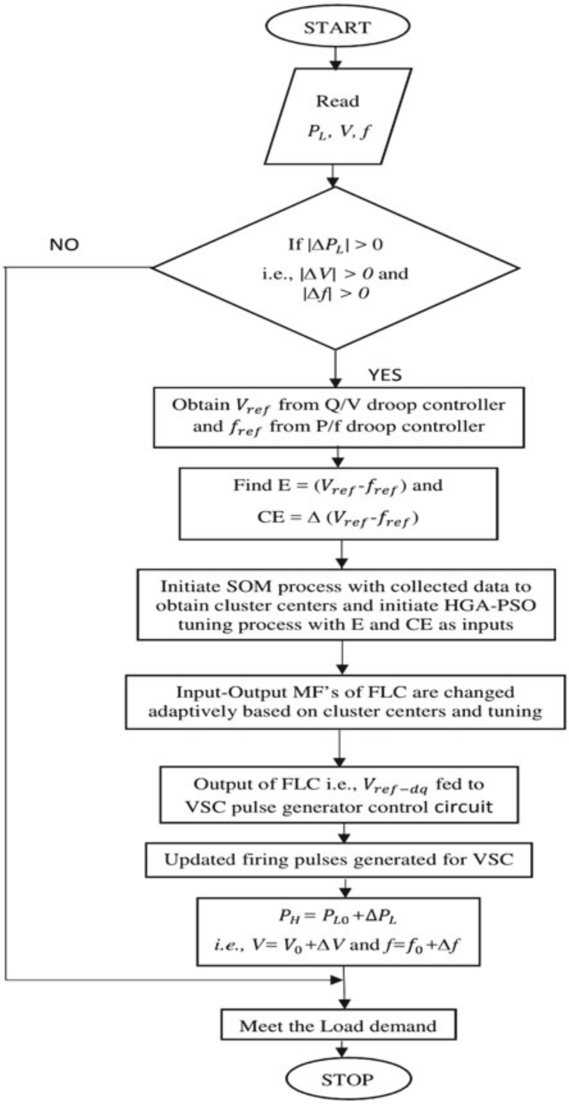


Fig. 8 Flowchart for the operation of SAFLDC

It begins with a set of solutions called population. Solutions from one population are taken and used to form a new population with hope, that the new population will be better than the old one. Solutions that are then selected to form new solutions are selected according to their fitness. This is repeated until some condition is satisfied.

Genetic Algorithms are playing an increasingly important role in studies of complex adaptive systems. Applications of GA include load flow, optimal power flow, economic dispatch, optimal reactive power dispatch, Optimal multistage planning of distribution networks, unit commitment, etc. GA is also used for optimal tuning of controller parameters while designing an optimal controller. It involves the minimization of the objective function (performance index) of the controllers. The control of a microgrid is a nonlinear complex problem such that a well efficient heuristic method like GA should be employed to solve it in reasonable computation time.

GA effectively explore many search space rather than a single region and hence it is less sensitive to a local minimum. But there are some deficiencies in GA performance like premature convergence which again degrades its efficiency and reduces the search capability.

Particle Swarm Optimization (PSO)

PSO is one of the newly developed optimization techniques with many attractive features. It has many applications in science and technology. It is a population-based approach. It is developed through simulation of bird flocking in multi-dimensional space [20].

It is similar to the genetic algorithm technique for optimization. In PSO, a population of individuals called the swarm is considered instead of concentrating on a single individual implementation. The algorithm then, rather than moving a single individual around, will move the population around looking for a potential solution. This is an example of a heuristic approach, in which there is no guarantee of an optimal solution.

In PSO, the coordinates of each particle represent a possible solution associated with two vectors, the position and velocity vectors. The particles of swarm fly through the feasible solution space to explore points where optimal solutions exist. During their search, particles interact with each other in a certain way to optimize their search experience. In each iteration, the particle with the best solution shares its position coordinates information i.e., global best with the rest of the swarm. Then, each particle updates its coordinates based on its own best search experience as well as the global best. It searches the optimal solution through continuous iteration, and it finally employs the size of the value of the objective function, or the function to be optimized, in order to evaluate the quality of the solution [20].

Unlike many other metaheuristic techniques, PSO has fewer parameters to tune and adjust. PSO algorithm is simple to comprehend, and easy to implement and to program since it utilizes simple mathematical and Boolean logic operations. Like GA, PSO is also less susceptible to getting trapped on local optimum [35]. Its applications include function optimization, artificial neural network training, proportional and integral fuzzy system control, and other near-optimal search and optimization areas where GA can be applied.

Hybrid GA PSO

In the present work, the hybrid GA-PSO is used to tune the parameters of the PI controller [22]. This algorithm is the combination of GA and PSO algorithm that utilizes the features of both algorithms and overcomes the limitations of GA and PSO. This algorithm is better in terms of convergence, robustness and precision. One of the disadvantages of PSO is that the swarm may converge to the point which is not guaranteed for a local optimum [35]. This point may be the line between particle best and global best. This problem may also be caused by the fast rate of information flow between particles, this increases the possibility of being trapped in local optima due to a loss in diversity. Another drawback of this type of stochastic approach is problem-dependent performance. This problem-dependent performance is caused by the parameter setting requirement of such type of algorithms can be addressed by combining advantages of different approaches through the hybrid mechanism. A hybrid algorithm with GA was proposed to overcome the limitations of PSO. Thus combined hybrid GA-PSO algorithm will have merits of PSO along with GA merits. One major advantage of PSO over GA is it can be easily applied to a wide range of problems and has the ability to control convergence.

One simple way to combine the GA and PSO techniques is an initial population of PSO is assigned by the solution of GA. The total number of iterations is equally shared by GA and PSO. The first half of the iterations are run by GA and the solutions are given as initial population of PSO and the remaining iterations are run by PSO. The steps followed in solving HGA-PSO is shown using a flow chart of Fig. 9.

In the present formulation, the model of the proposed SAFLDC has two variables e $(= E \cdot K_p)$ and ce $(= CE \cdot K_i)$ that are used as input signals and the output signal is U $(= V_{dq-ref})$. The overall tuning of SAFLDC is defined as an optimization problem to control VSC pulse signals in order to match the generation and load demand in the considered islanded microgrid. Tuning is done by considering the parameters of SAFLDC as linearly equivalent to conventional PI controller gains using Hybrid GA-PSO Algorithm. K_p and K_i transform the scaled real values to the desired value in the decision limit and are tuned using Hybrid GAPSO.

The controller gains are tuned using Hybrid GA-PSO by minimizing the Objective Function; Integral Squared Error in Eq. (14) while satisfying the equality and inequality constraints.

$$J = \int_0^T (P_G - P_L)^2 + (\Delta f)^2 + (\Delta V)^2 dt \tag{14}$$

The constraints include:

1. Power Balance Constraint:

$$\sum P_G = \sum P_L \tag{15}$$

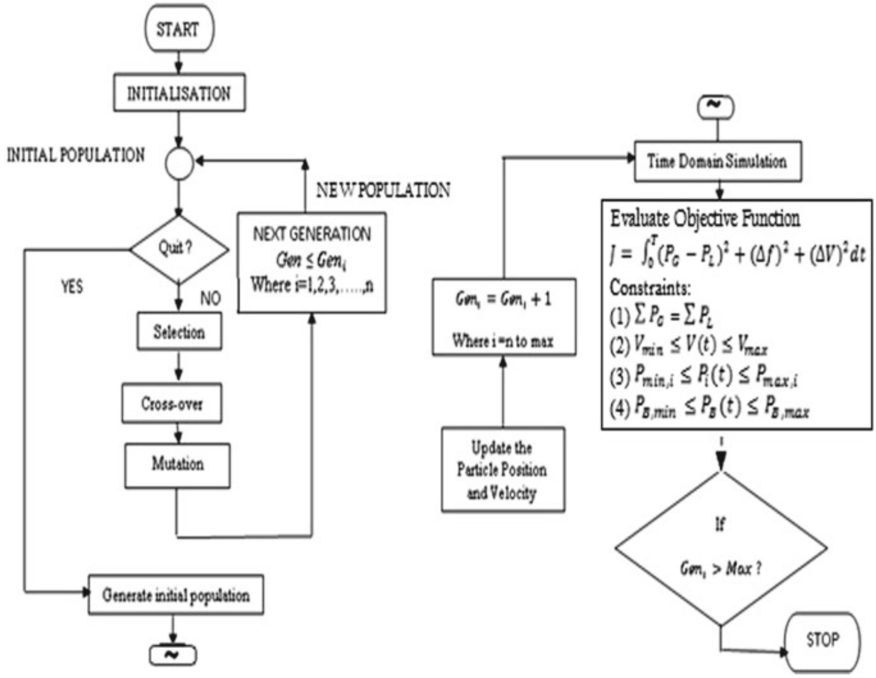


Fig. 9 Flow chart for the hybrid GA PSO algorithm

2. Voltage Constraint:

$$V_{\min} \leq V(t) \leq V_{\max} \tag{16}$$

3. Unit Power Generation Constraint:

$$P_{\min,i} \leq P_i(t) \leq P_{\max,i} \tag{17}$$

4. Battery Capacity Constraints:

$$P_{B,\min} \leq P_B(t) \leq P_{B,\max} \tag{18}$$

The proposed SAFLDC has K_p and K_i as the two scaling factors of the inputs. HGA-PSO technique is mainly utilized to determine their optimal values such that a controlled system obtains the better coordination and power management among all power sources in the islanded microgrid.

4 Results and Analysis

A detailed switching model of a microgrid test system with two hybrid units and one droop unit is simulated separately with the PI controller and the proposed control strategy, in MATLAB/Simulink. The microgrid specifications and controller parameters are listed in Appendix. The performance of microgrid in response to variations in load, SOC and irradiance is studied using the obtained waveforms. The microgrid response with both the controllers is also compared.

1. *Performance of microgrid in response to variations in the load*

The charging power is determined based on the respective battery SOC which is 85% in the hybrid unit 2, and 60% in the hybrid unit 1. The behavior of the key variables in the developed strategy is illustrated in Fig. 10. The operation of units in a microgrid is as follows.

Up to 0.1 s the hybrid units share the load equally and at the same time charge their batteries from their own PV power. The charging power is determined based on the respective battery SOC which is 85% in the hybrid unit 2, and 60% in the hybrid unit 1. Both units are initially giving less than maximum PV power by adjusting their PV array voltages above the maximum power voltage (v_{mppt}), as shown in Fig. 10.

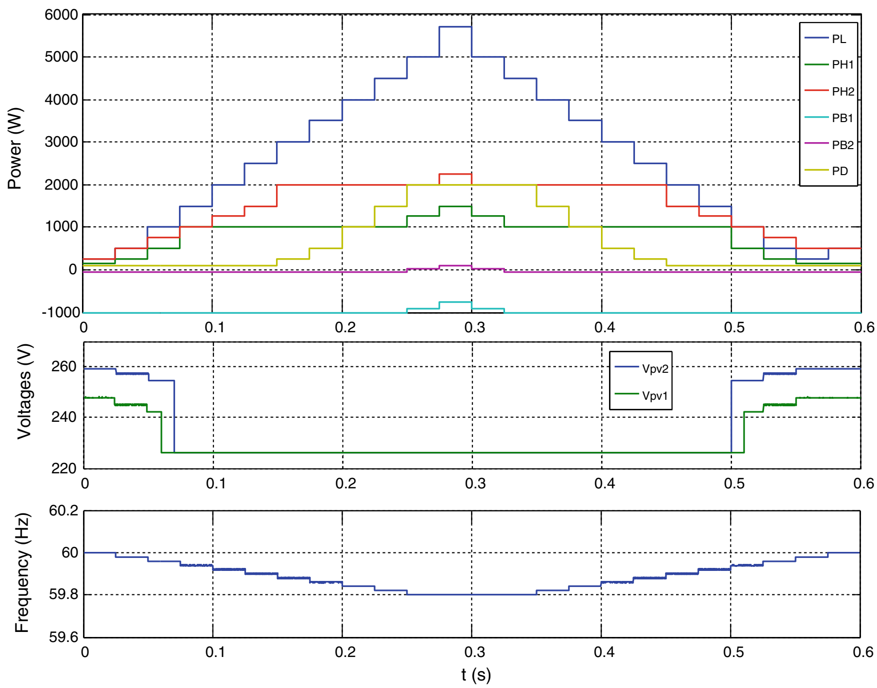


Fig. 10 Microgrid response to variations in the load

Up to 0.1 s, the entire microgrid is in the PV segment, so the droop unit is in a non-operating state.

At t = 0.1 s, Hybrid unit 1 reaches its limit $P_{pv1} - |P_{ch}|$, and starts operating at MPP. For an increase in the load, hybrid unit 1 keeps its output at its MPP, while hybrid Unit 2 supplies the increased load.

At t = 0.15 s, hybrid unit 2 reaches its power limit and starts operating MPP. Now the droop unit comes into action by feeding the extra load. The microgrid operates in the droop segment.

At t = 0.25 s, the droop unit reaches its maximum power limit at $f = f_B$, and the hybrid units start operating as voltage sources and the load frequency is constant. Between t = 0.25 s and t = 0.325 s, the hybrid units supply the peak load. As a hybrid unit has higher SOC, it starts discharging first and hybrid unit 1 continues charging at a reduced rate. After t = 0.325 s, the microgrid starts operating in the droop segment in response to the load decrease.

With SAFLDC, Fig. 11a–c prove that the variations in power at hybrid units, batteries, and droop units respectively are smooth compared to that of with PI controller. As these are used as references for other controllers and power converters, the operation of them can be smooth and effective. From Fig. 11b, it can be observed that there is no unwanted charging or discharging of batteries. Fluctuations or ripple in PV array voltages are minimized, see Fig. 11d, e.

2. *Microgrid response to variations in SOC and load*

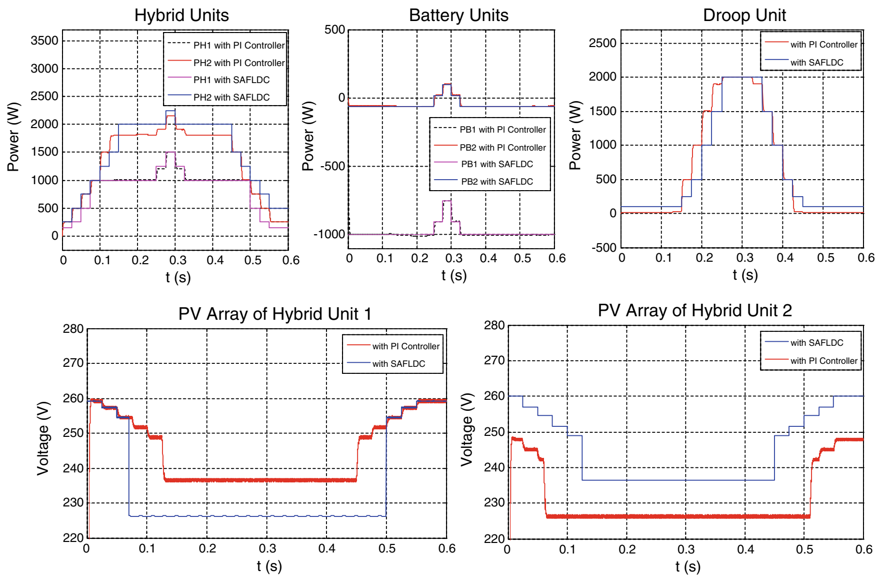


Fig. 11 Comparison of microgrid response for load variations for PI controller and SAFLDC

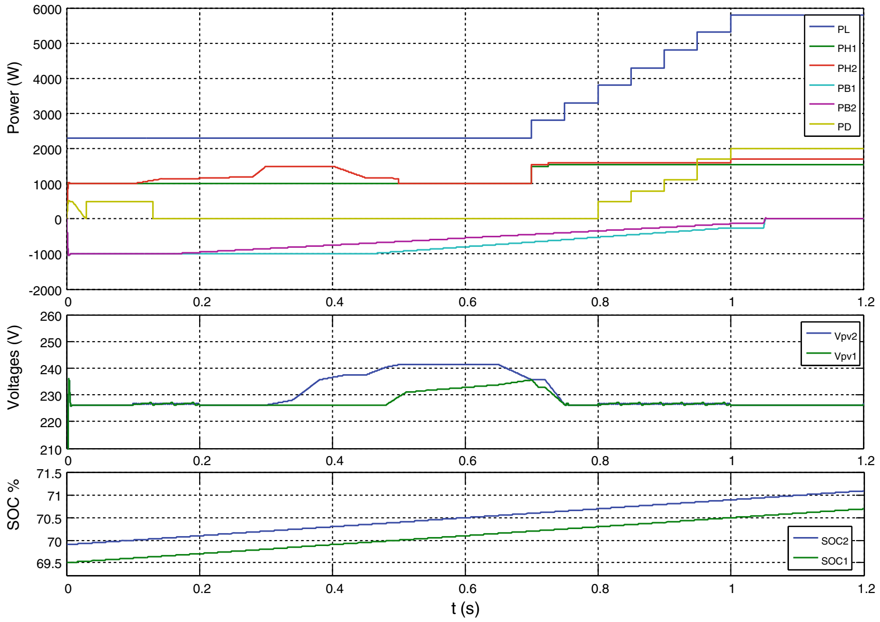


Fig. 12 Microgrid response to variations in the SOC and the load

SOC_{ref} is set to 71%. In this case, the charging reference $|P_{ch}|$ starts dropping significantly once SOC_2 exceeds 70% at t = 0.08 s. Accordingly, the Power limiting controller starts increasing the output power, while the droop unit output power decreases until it reaches zero at t = 0.3 s. At this point, the Power limiting controller reaches its operating range limit, and therefore, hybrid unit 2 starts operating as a droop unit in the PV segment. Accordingly, the Charging controller starts curtailing PV power production to follow the decreasing reference $|P_{ch}|$ (Fig. 12).

Similarly, when SOC_1 exceeds 70%, the Power limiting controller starts increasing the output power until t = 0.5 s, when both hybrid units start operating as droop units in the PV segment while sharing the load equally. Accordingly, hybrid unit 1 starts curtailing PV power to follow the decreasing reference $|P_{ch}|$.

At t = 0.7 s, the increased load is supplied by the hybrid units until both units reach their maximum available power again at t = 0.75 s. Beyond this point, the further increase in the load is supplied by the droop unit until it reaches its maximum limit at t = 1 s. Thereafter, any increase in the load is supplied by the hybrid units.

The operation of the droop unit with SAFLDC is instantaneous as shown in Fig. 13c when compared to that of with PI controller. As observed in Fig. 13d, batteries are charging smoothly. This eliminates the burden on them and improves their life. With PI controller, the ripples in PV array voltages are continuous for continuous step changes in load but with SAFLDC they are minimized (Fig. 13e, f).

3. *Microgrid response to solar irradiance and temperature variations*

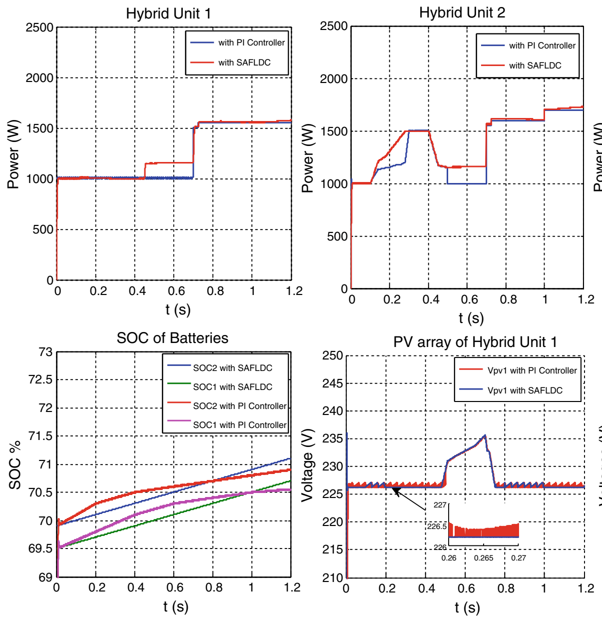


Fig. 13 Comparison of microgrid response for SOC and load variations for PI controller and SAFLDC

A hybrid unit 1, solar irradiance is dropped from 1000 W/m^2 to 750 W/m^2, and 500 W/m^2, at t = 0.45 s, and 0.55 s, respectively. At the same time, the temperature increases linearly from 25° at t = 0 s to 60° at t = 0.70 s. At t = 0.45 s, Hybrid Unit 1 enables the MPPT algorithm and starts tracking the new Maximum Power Point, drops the unit output power to the new limit. And again at t = 0.55 s, above step repeats and PV power production reduces, hybrid unit 1 starts importing power to support charging the battery and droop unit comes into action (Fig. 14).

It can be observed from Fig. 15, that the proposed SAFLDC controller is also responding quite effectively for temperature and irradiance variations and taking necessary control actions. Thus, it eliminates the disadvantages due to the intermittent nature of PV sources.

5 Conclusions

The proposed multi-segment P/f droop control strategy with SAFLDC for islanded microgrid has been simulated in MATLAB/Simulink environment. The proposed SAFLDC provides more effective localized VSC control for autonomous power balance maintenance in islanded microgrids. It overcomes the disadvantages that occurred with the use of conventional PI controllers. It uses optimum values

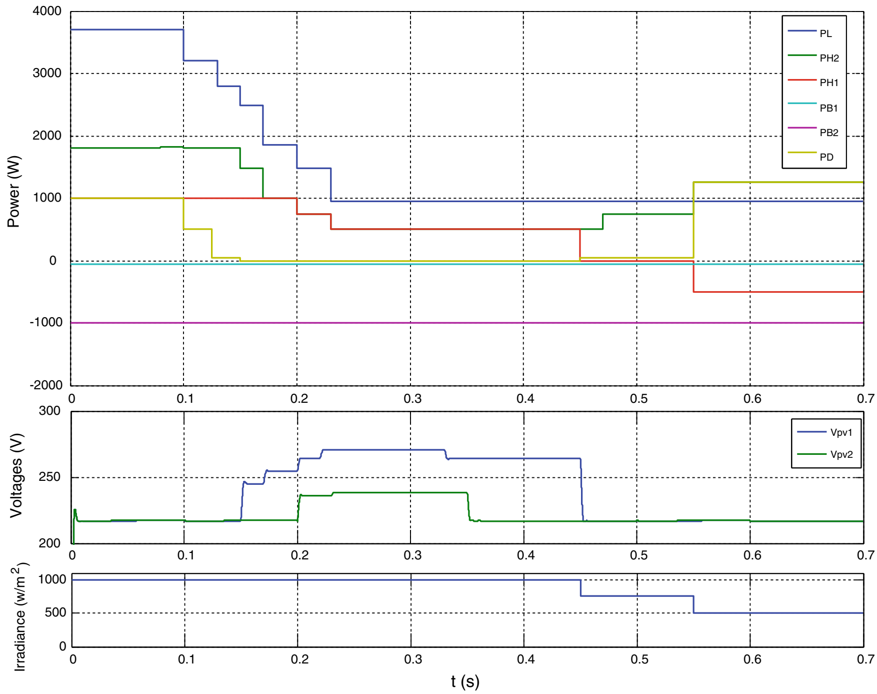


Fig. 14 Microgrid response to the variations in solar irradiance and temperature

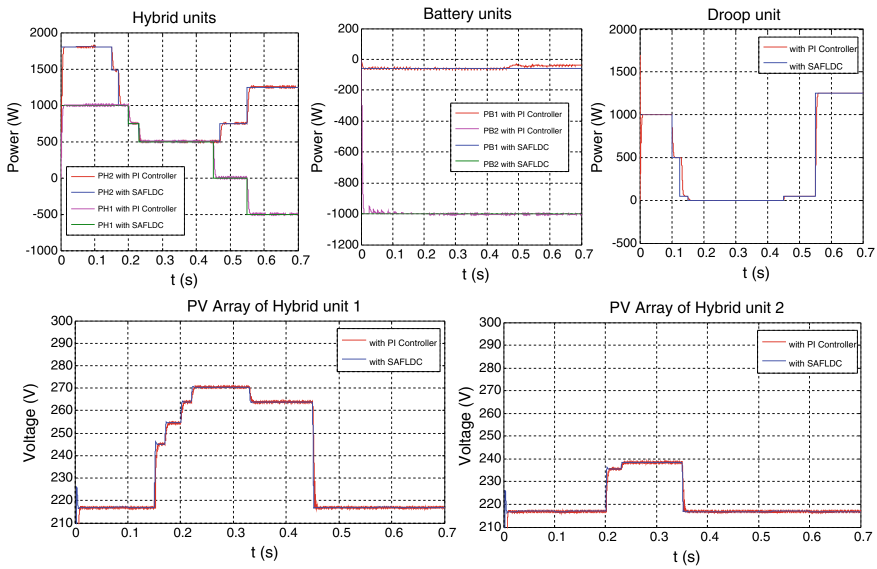


Fig. 15 Comparison of microgrid response for solar irradiance and temperature variations for PI controller and SAFLDC

of controller parameters by tuning using Hybrid GA-PSO and adaptive fuzzy membership function tuning and optimal fuzzy rule base using a SOM clustering algorithm.

In case 1 of load variations, curtailment of PV generation occurs for fewer loads. The operation of droop units is in action only when available PV power reaches its maximum. For high peak loads, when PV sources and droop units together could not meet the demand, batteries are supplying the power by discharging. Thus an autonomous power balance is being maintained in the islanded microgrid. All these control actions are local without the need for communications between units and also for a centralized energy management control unit. These local voltage controllers are acting based on the adaptive P/f droop characteristic curves. These curves are adjusted locally for various operating conditions.

From case 2, the results obtained show that a priority of charging is being given to the batteries which have lower SOC and the hybrid units reach an equal SOC finally. From case 3, the results show that the proposed control strategy also keep tracking the variations in environmental factors like irradiance and temperature effects, and taking the necessary control actions.

For the conventional PI controllers, there is a substantial change in parameters for a major external disturbance. In the presence of such a disturbance, the conventional PI controller fails. In this case, the fuzzy controller offers to implement simple but robust solutions that convert a wide range of system parameters and can handle major disturbances.

With the use of SAFLDC, there are smooth variations in power and voltages as can be seen from the obtained waveforms. There are fewer ripples in the voltages waveforms. Thus this control strategy acts more sensitively to the variations in load, SOC, irradiance and temperature. Thus it can be concluded that the proposed SAFLDC technique for the control of the Voltage source converter is considerably effective than the conventional PI controllers.

The main focus in this work is on real power management while reactive power-sharing can be performed using the conventional reactive power/voltage droop control technique. Here in this work, only two hybrid units and one droop unit are considered. The same control technique can be applied to any number of hybrid units and droop units effectively. Other sources of distributed generation like Fuel cells, Wind turbines, etc. can also be installed in the Hybrid unit. Other applications, such as market participation and economic dispatch can be done using communications among units and centralized algorithms.

Appendix

See Table 2.

Table 2 Microgrid specifications and controller parameters

Description	Parameter	Value
Nominal frequency	f_o	60 Hz
Minimum frequency	f_{min}	59.7 Hz
PV segment	Δf_{PV}	0.1 Hz
Droop segment	Δf_D	0.1 Hz
Battery segment	Δf_B	0.1 Hz
Battery capacity	C_{bat}	10 Ah
Battery converter rating	P_{B-max}	1000 W
PV array power rating	P_{pv-m}	2 kW
Droop unit power rating	P_{D-m}	2 kW
Power limiting controller		
Proportional gain	K_p	0.001 Hz/W
Integral gain	K_i	0.005 Hz/(W s)
Battery limiting controller		
Proportional gain	K_p	0.001 Hz/W
Integral gain	K_i	0.005 Hz/(W s)
Charging controller		
Proportional gain	K_p	0.05 V/W
Integral gain	K_i	0.625 V/(W s)

References

1. Pranith S, Kumar S, Singh B (2019) Multimode operation of PV-battery system with renewable intermittency smoothening and enhanced power quality. IET Renew Power Gener 13(6):887–897
2. Pourbabak H, Chen T, Zhang B, Su W (2017) Control and energy management system in microgrids. In: IET GTD, clean energy microgrids. https://doi.org/10.1049/PBPO090E_ch3
3. Anvari-Moghaddam A, Guerrero JM, Vasquez JC et al (2017) Efficient energy management for a grid-tied residential microgrid. IET GTD 11(11):2752–3276
4. Lasseter RH, Eto JH, Schenkman B, Stevens J, Vollkommer H, Klapp D, Linton E, Hurtado H, Roy J (2011) CERTS microgrid laboratory test bed. IEEE Trans Power Deliv 26(1):325–332
5. Serban E, Serban H (2010) A control strategy for a distributed power generation microgrid application with voltage- and current controlled source converter. IEEE Trans Power Electron 25(12):2981–2992
6. Guo Z, Sha D, Liao X (2014) Energy management by using point of common coupling frequency as an agent for islanded microgrids. IET Power Electron 7(8):2111–2122
7. Wu D, Tang F, Dragicevic T, Vasquez JC, Guerrero JM (2014) Autonomous active power control for islanded ac microgrids with photovoltaic generation and energy storage system. IEEE Trans Energy Convers 29(4):882–892
8. Karimi Y, Oraee H, Golsorkhi M, Guerrero J (2017) Decentralized method for load sharing and power management in a PV/battery hybrid source islanded microgrid. IEEE Trans Power Electron 32(5):3525–3535
9. Loh PC, Chai YK, Li D, Blaabjerg F (2014) Autonomous operation of distributed storages in microgrids. IET Power Electron 7(1):23–30

10. Du W, Jiang Q, Erickson MJ, Lasseter RH (2014) Voltage-source control of PV inverter in a CERTS microgrid. IEEE Trans Power Deliv 29(4):1726–1734
11. Mahmood H, Michaelson D, Jiang J (2015) Strategies for independent deployment and autonomous control of PV and battery units in islanded microgrids. IEEE J Emerg Sel Top Power Electron 3(3):742–755
12. Mahmood H, Michaelson D, Jiang J (2014) A power management strategy for PV/battery hybrid systems in islanded microgrids. IEEE J Emerg Sel Top Power Electron 2(4):870–882
13. Adhikari S, Li F (2014) Coordinated V-f and P-Q control of solar photovoltaic generators with MPPT and battery storage in microgrids. IEEE Trans Smart Grid 5(3):1270–1281
14. Bae S, Kwasinski A (2014) Dynamic modeling and operation strategy for a microgrid with wind and photovoltaic resources. IEEE Trans Smart Grid 3(4):1867–1876
15. Valverde L, Rosa F, Bordons C (2013) Design, planning and management of a hydrogen-based microgrid. IEEE Trans Ind Inform 9(3):1398–1404
16. Mahmood H, Jiang J (2017) Autonomous coordination of multiple PV/battery hybrid units in islanded microgrids. IEEE Trans Smart Grid 9(6)
17. Al Badwawi R, Issa W, Mallick T, Abusara M (2016) Power management of AC islanded microgrids using fuzzy logic. In: 8th IET international conference on power electronics, machines and drives (PEMD 2016)
18. Fogel LJ, Owens AJ, Walsh MJ (1965) Artificial intelligence through a simulation of evolution. In: Maxfield M, Callahan A, Fogel LJ (eds) Biophysics and cybernetic systems: proceedings of the 2nd cybernetic sciences symposium. Spartan Books, pp 131–155
19. Holland JH (2011) Adaptation in natural and artificial systems. Aust J Basic Appl Sci 5(3):295–302
20. Kennedy J, Eberhart RC (1995) Particle swarm optimization. In: Proceedings of the 1995 IEEE international conference on neural networks, vol 4. IEEE Press, pp 1942–1948
21. Storn R, Price K (1995) Differential evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical report. International Computer Science Institute, Berkley
22. Dash P, Saikia LC, Sinha N (2015) Automatic generation of multi area thermal system using bat algorithm optimized PD-PID cascade controller. Electr Power Energy Syst 68:364–372
23. King PJ, Mamdani EH (1977) The application of fuzzy control systems to industrial processes. Automatica 13(3):235–242
24. Zadeh LA (1965) Fuzzy sets. Inf Control 8:338–353
25. Wichasilp C, Wiriyasuttiwong W, Kantapanit K (2003) Design of fuzzy logic controllers by fuzzy c-means clustering. Thammasat Int J Sci Technol 8(2)
26. Shayeghi H, Shayanfar HA, Malik OP (2007) Robust decentralized neural networks based LFC in a deregulated power system. Electr Power Syst Res 77(3):241–251
27. Jain S. A textbook on "Modelling and simulation using matlab-Simulink", 2nd edn. Wiley, pp 589–666
28. Vakula VS, Sudha KR (2012) Design of differential evolution algorithm-based robust fuzzy logic power system stabiliser using minimum rule base. IET Gener Transm Distrib 6(2):121–132. https://doi.org/10.1049/iet-gtd.2011.0195
29. Vuorimaa P (1993) Fuzzy self-organizing maps. Fuzzy Sets Syst 66:223–231
30. Lee SC, Lee ET (1975) Fuzzy neural networks. Math Biosci 23:151–177
31. Kohonen T (1990) The self-organizing map. Proc IEEE 78(9):1464–1480
32. Wang Y, Mathee K, Narasimhan G (2005) Clustering using adaptive self-organizing maps (ASOM) and applications. Bioinformatics Research Group (BioRG), School of Computer Science, Florida International University, Miami
33. Bação F, Lobo V, Painho M (2005) Self-organizing maps as substitutes for K-means clustering. In: ICCS 2005. LNCS, vol 3516. Springer-Verlag, Berlin Heidelberg, pp 476–483

34. Vakula VS, Sireesha N, Vamsi Krishna S (2016) Design of membership functions for fuzzy power system stabilizer using self organized mapping. In: 2016 international conference on advances in electrical, electronic and systems engineering (ICAEES)
35. Saikia LC, Nanda J, Mishra S (2011) Performance comparison of several classical controllers in AGC for multi-area inter connected thermal system. Electr Power Energy Syst 33:394–401

Skin Melanoma Segmentation Using Neural Networks Optimized by Quantum Invasive Weed Optimization Algorithm

Navid Razmjooy and Saeid Razmjooy

Abstract Early detection of skin cancer makes a high chance for the patient to escape from the malady and cure him/her at initial stages. In other words, by early detection of skin cancer, the quality of human life improves. In recent years, a wide range of dermatology clinics and hospitals employed systems based on image processing and computer vision for early detection of skin cancer. In this paper, a new method based on the optimized artificial neural network is presented to recognize the malignant lesion of skin cancer from benign lesions. To do this purpose, at first, a number of pre-processing operations are applied to the input image to filter noise and unwanted parts. Afterward, the proposed optimized neural network based on Quantum Invasive Weed Optimization Algorithm is applied to the filtered image for separating the skin lesion regions. To analysis the system performance, it has been applied to the DermIS Database and Dermquest Database. Experimental results show that the proposed method has a good efficiency for the skin lesion segmentation.

Keywords Medical image processing · Melanoma · Skin cancer · Artificial neural network · Quantum invasive weed optimization algorithm

1 Introduction

Skin cancer is the most common malignant cancer in the human body. It is estimated that more than one million people in the world every year suffer from skin cancer. Skin cancer has a variety of different types; the main difference among them is the type of cell that is cancerous [10].

Three main types of skin cancers are Basal Cell Carcinoma, Squamous Cell Carcinoma, and Malignant Melanoma.

N. Razmjooy (✉)
Department of Electrical and Control Engineering, Tafresh University, 39518 79611 Tafresh, Iran
e-mail: navid.razmjooy@hotmail.com

S. Razmjooy
Department of Biosystems Engineering, Faculty of Agriculture Technology and Natural Resources, University of Mohaghegh Ardabili, Ardabil, Iran
e-mail: saeidrazmjooy@gmail.com

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_12

Long exposure to sunlight is the main cause of skin cancer, especially if it causes sunburn and blistering.

Other factors that are less important are frequent medical or occupational exposure (in factories) with X-rays, scarring of diseases and burns, and occupational exposure to some chemicals such as coal and arsenic. Skin cancer may have a heritable background, so in some families, the incidence of skin cancer is higher [23].

Skin cancer, especially in the early stages, may not be very similar to cancerous lesions but is more like ordinary skin lesions.

For this reason, any chronic and unusual lesions should be taken into consideration. Particularly any change in size, color, shape or thickness of the mole should be monitored. Try to understand the symptoms of skin cancer so that they can be detected in a timely manner in the event of a suspicious change [17].

Malignant skin cancer first occurs with pre-cancerous lesions. Pre-cancerous lesions are skin lesions that are not cancerous but become cancerous over time. If anyone knows the signs of skin cancer warning, it can cure the disease at the very beginning stage, as skin cancer can be cured if it is diagnosed promptly.

Malignant melanoma is the deadliest type of skin cancer that originates from skin pigmentation cells (melanocytes).

In this disease, cancer cells continue to pigmentation, so in this type of cancer, shades of brown, brown or black are seen in different colors, of co [24].

Currently, the only definitive way to diagnose skin cancer is to do a biopsy, in which physicians perform the removal of suspected skin lesions, and then examine the tissues stained with cancer cells under a microscope. Detection and examination of melanoma can sometimes be white or red. Malignant melanoma has a high tendency for dissemination to other organs of the body, so diagnosis and treatment are very important in the early stages to sample skin lesions are inappropriate; however, for each discovery and diagnosis of skin cancer, about 25 negative biochemical samples are taken. That's estimated at $ 6 billion for the US healthcare system, according to estimates by researchers.

With the development of science in recent years, digital dermatoscopes have been replaced by conventional dermoscopy with the ability to capture and store dermatological images.

Therefore, it is possible to provide commercial software packages to help identify some skin lesions, create databases, and create resume therapy for each patient (Fig. 1).

Studies show that the most effective step in the processing of dermatoscopic images of melanocytic lesions is to marginalize the lesion. In fact, the extent of the lesion, the shape of the border, the amount of tzarysis within the gap between the lesion and the background are considered as key parameters in the diagnosis of cancer [13].

Since eye diagnosis is precisely the exact boundary of the lesion, especially in the early stages of the onset of the disease, it is very difficult and in some cases that is impossible, and on the other hand, the onset of the treatment process has a direct effect on reducing mortality from skin cancer, so the processing techniques

Fig. 1 Some varieties of melanoma

that improve the boundaries is always a matter of interest in the recent researches [6].

This skin imaging technique compares to traditional clinical methods, which makes the underlying structures of the skin more visible. This, in turn, reduces the coverage error and increases the resolution in some cases such as obscurantist complications.

In microscopy, a non-invasive diagnostic method for observing the inside of the body to examine skin lesions [14].

Masked images have a high potential for early diagnosis of malignant melanoma, but their interpretation is even time-consuming for dermatologists. So now, special attention has been paid to the advancement of computer-based diagnostic systems, which can be of great help in analyzing dermatologists.

To reduce the diagnostic error caused by human perceptual and visual factors, the development of computerized image analysis is of great importance.

Automatic border detection is often considered as the first step in the automatic analysis of images in microscopy. Usually, the standard method for automatic analysis of macroscopic images consists of three steps:

1. Image segmentation
2. Feature extraction
3. Cancer classification.

The segmentation stage is one of the most important parts, due to its high impact on the accuracy of the next steps.

Although segmentation is difficult due to the image's different shapes, sizes, and colors among different types of cancerous tissues. In addition, some cancers have an uncertain border and in some cases, there is a very mild area between skin and cancer. One of the other problems is the dependence of the black hair that covers the cancerous areas and the presence of reflection in the images.

To overcome this problem, different algorithms have been proposed. These methods are generally classified based on thresholding, edge-based methods, and district-based methods. One example of thresholding methods can be found in Erkol et al. [9]; in this method, a hybrid method including global thresholds, comparative thresholds and clustering are used.

When there is a good contrast between skin and cancer, thresholding methods yield good results, but these methods become problematic when the two areas overlap. Among the edge-based methods, there are a number of active contour techniques, such as gradient flow.

Edge-based methods work poorly in situations where borders are not well distinguished, in other words, when the boundary between cancer and skin cannot be distinguished. In these situations, the edges have a gap and the contour may be leaked between them.

Another problem is the presence of fake edges that do not relate to the main edge of cancer. This problem is due to the presence of artifacts such as hair, reflection, or even disorientation in the skin tissue that prevents the correct detection of the boundary of cancer.

Region-based methods are also used to diagnose cancer. One of these methods is the Morphological Flood Model [29] and the Accurate Markov Accidental Field Accuracy Algorithm [11]. When there is cancer or skin region of varying colors, region-based methods face a big problem that makes the segmentation inappropriate and excessive.

In [7], an unsupervised method was used based on a modified model of the JSEG algorithm for border detection. In [8], the use of algorithms for the integration of statistical areas based on the growth of the regions and their integration separates the problem from the background image. In [32], the GVF Snake method has been used to divide skin cancer images by using a decent filter to de-duplicate it and then segment the image using the GVF Snake multi-directional.

In 2013, another study was conducted that examined the diagnosis of skin cancer using both global and local methods [22]. The research has pursued two goals, determining the best detection method (national or local) and determining the best features (color or texture properties).

In the designed experiments, three different classifications were used, and a combination of different features was considered, with significant results obtained.

The results show that good results can be achieved using a simple Knn stratigraphy, and also the color properties have a better performance than texture properties. Also, a small number of features are sufficient for high-precision classification and ensures an increase in system generality (rather than the use of a large number of features).

The results also show that with both detection methods (national and local), good results can be achieved, with the difference that the local method needs more time.

In another study that was conducted in 2013, the role of shape features in classifying images in microscopy has been investigated [4].

In 2017, an automatic lesion segmentation based on using convolutional networks [21] with Jaccard Distance.

They presented an automatic method for lesion segmentation using 19-layer deep convolutional neural networks that are trained end-to-end and do not rely on prior knowledge of the data. Final results showed that their approach has a higher performance than the state-of-the-art algorithms [35].

Venkatesh et al. [33] proposed another automatic method for lesion region segmentation based on multi-scale residual connections based on UNet. They lost the information in the encoder stages due to the max-pooling layer at each level is preserved through the multi-scale residual connections. For performance analysis, they employed ISIC 2017 challenge dataset without using any external dermatologic image set [33].

In this paper, a new optimized method is proposed based on applying a newly introduced optimization algorithm, called Quantum Invasive Weed Optimization Algorithm on the multi-layer perceptron neural network. The proposed method is performed under MATLAB 2017 platform.

The rest of the paper is detailed as follows: in Sect. 2, pre-processing of the input images has been explained; in Sects. 3 and 4, ANN and QIWO are introduced as soft computing tools for medical image segmentation. In Sect. 5, the method for optimizing ANN based on QIWO is described. Section 6 explains the mathematical morphology in brief. Section 7 introduces the main dataset utilized in the research. Section 8 shows the final experimental results.

Finally, research findings are concluded in Sect. 8. The flowchart diagram of the suggested lesion diagnosis is shown in Fig. 2.

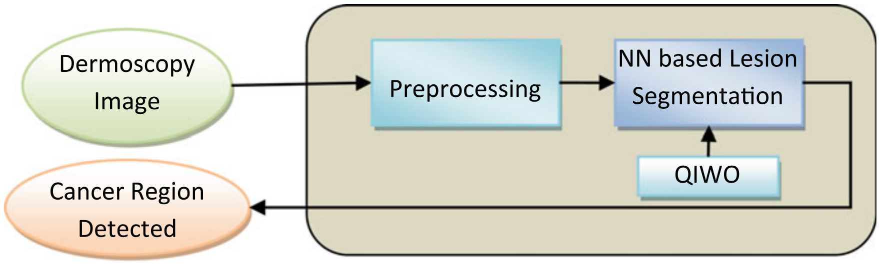


Fig. 2 Flowchart of the designed lesion detects system

2 Input Image Pre-processing

2.1 Histogram Equalization

Given that the histogram of an image is the probability density function of the gray levels of the image, it is possible to determine the brightness level in the image. If the image histogram is distributed around finite gray levels, then this function can be used with the help of information theory [12].

In such a way that the form of the distribution function (histogram) is uniform. Sometimes certain parts of the image are to be processed. For this purpose, a localization of a histogram with a Gaussian function is obtained which shows better results than the rest of the functions [16, 18–20]. As a result of this operation, image differentiation improves.

Histogram equalization is a method for adjusting the image intensity to get better results in contrast. Assume f as a given image described by a matrix with a size of $[m_r \times m_c]$ of integer pixel intensities which is spread in the interval [0, L − 1] [30].

L presents the number of possible intensity values and is usually considered 256. Let p describes the normalized histogram of f with a bin for all possible intensities,

$$P_n = \frac{number\ of\ pixels\ with\ intensity\ n}{total\ number\ of\ pixels}, \quad n = 0, 1, \ldots, L-1 \tag{1}$$

In this case, the adjusted image (g) can be presented as follows:

$$g_{i,j} = floor\left(L - 1\sum_{n=0}^{f_{i,j}} p_n\right) \tag{2}$$

where, *floor(.)* rounds the value down into the closest integer. In other words, pixel intensities transformation, k, of f based on the following formula.

$$T(k) = floor\left(L - 1\sum_{n=0}^{k} p_n\right) \tag{3}$$

Transformation here is utilized to consider the intensities of f and g as random and continuous variables X, Y between 0 and L − 1. Y can be described as the following formula:

$$Y = T(X) = (L-1)\int_{0}^{X} px(x)dx, \tag{4}$$

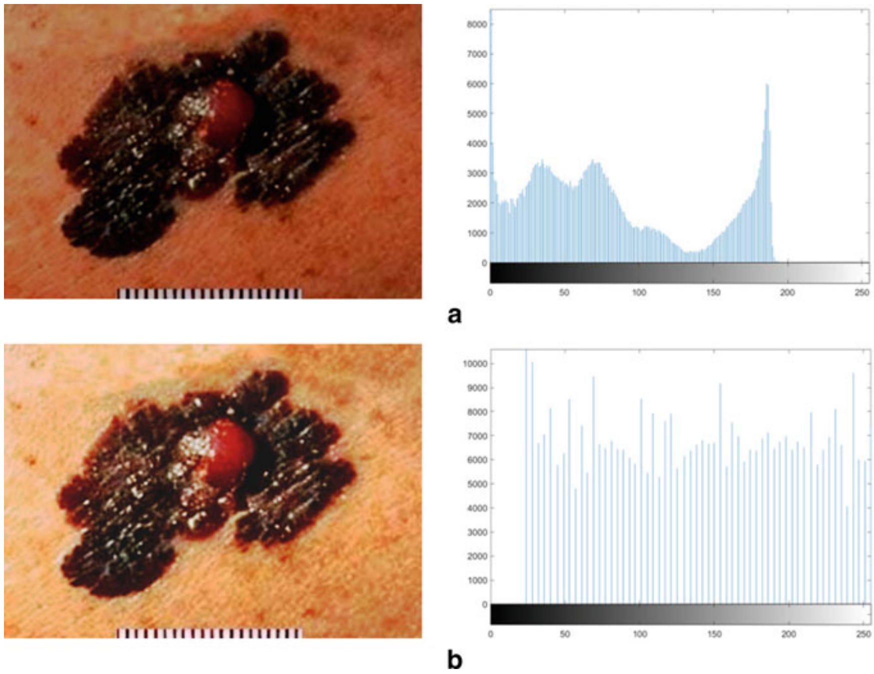


Fig. 3 Histogram equalization and it histogram before (**a**) and after (**b**) image after histogram equalization

where *px* is the histogram of f. T illustrates the cumulative distributive function of X multiplied by (L − 1). In this study, T is assumed invertible and differentiable. A simple example of the Histogram equalization and it histogram before and after image after histogram equalization is shown in Fig. 3.

2.2 Digital Noise Removal Using Median Filter

The median filter is a low pass filter, which requires more processing time than other filters. The median low pass filter uses a neighborhood of m × n to arrange all neighborhoods in ascending order and consider the mean value of them to replace in the central pixel [5].

$$y[m, n] = median\{x[i, j], (i, j) \in \omega\} \tag{5}$$

where, ω describes a neighborhood centered around position (m, n) of the image.

It should be noted that the low-pass filter for removing salty pepper can be used. In this project, the median filter is employed for removing the digital noise [34].

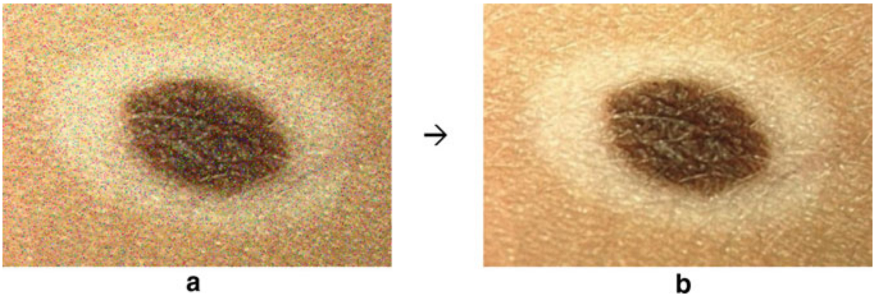


Fig. 4 Input image noise removal: **a** input image with pepper and salt noise (density = 0.08), **b** image after noise reduction

For this, a medium filter with 5 × 5 masks is applied to the image. The bigger the size of the mask, the better the noise reduction, of course with the loss of the edges.

In Fig. 4, a noisy image with its filtering by the median filter is shown.

3 Artificial Neural Network

An artificial neural network (ANN) is a machine learning system that simulates human brains processing for solving nonlinear and complicated problems. In ANN, based on good programming knowledge, a data configuration is designed to act like a human brain's neuron.

This is called a node. Then, each network can be trained by generating nodes and combining them based on a learning algorithm [1–3].

Each memory of the neural network has one of active (on or 1) or inactive (off or 0) mode. The modeled synapse for ANN as an edge connects the nodes to each other within a weighted value. Edges with positive weight activate the next inactive nodes, and edges with negative weighted make the next connected node inactive (when it is active).

Generally, ANN is a parallel computational system that generates a lot of simple connected elements to each other in a specific way to apply a specific task. ANN has a superpower computing capability that can generalize and learn from training data that simplifies the programming.

A neuron's network function $f(x)$ can be mathematically presented as a form of other functions $g_i(x)$, that can be described additionally as a composition of other functions.

One of the popular types of composition in ANNs is the nonlinear weighted sum which is described below:

$$f(x) = K\left(\sum_i \omega_i g_i(x)\right) \tag{6}$$

where, K describes a pre-defined function, such as hyperbolic tangent and $g = (g_1 \ldots g_n)$ is referred to a collection of functions g_i as simply as a vector.

The backpropagation (BP) algorithm is a popular method in feedforward networks. It calculates the error of all training pairs and then the weights are adjusted for fitting the desired output. This process is applied after some iterations until the total training set error becomes small enough or when the error stops to descend.

Afterward, the network is ready to evaluate the output values for the considered samples. Unfortunately, the BP algorithm based on its nature is a gradient-based method which makes it trapped into the local minimum. Since the success of it fully depends on the initial (weight) adjusting. In this paper for compensating for this problem, the QIWO algorithm is employed.

4 Quantum Invasive Weed Optimization Algorithm

Recently, meta-heuristic algorithms that are inspired by various phenomena turned into popular methods for solving complicated problems. The phenomena inspired by social, natural, etc. [15, 19, 27]. One of the new popular met heuristic methods is an Invasive weed optimization algorithm (IWO) [25, 31].

IWO was first introduced by Mehrabian and Lucas in 2006. This algorithm was motivated by the natural behavior of weeds in colonizing and discovering a suitable place for reproduction and growth. This method is inspired based on the colonization of invasive weeds. Weeds have illustrated a very adaptive and robust nature. This makes them inappropriate plants in agriculture.

In 2014, Razmjooy proposed an improved version of IWO based on quantum computing. The significant concept for the Quantum based IWO is that improves the exploration part in a quantum search space. In the following the concept of quantum, IWO is detailed.

4.1 Initialization

In quantum computing, the smallest unit is a quantum bit or Q-bit. A Q-bit, unlike traditional bits, maybe in the state “1”, state “0” or in any superposition between these two. In order to enhance the stochastic model for initializing the seeds (population), each seed is defined as one Q-bit that is called Q-seed (Fig. 5).

The position of Q-bits (Ψ) can be achieved as follows:

$$\Psi = \bigcup_{j=1}^{n} \left|\psi_j(t)\right\rangle = [\alpha_j \ \beta_j]^T$$
$$j = 1, 2, \ldots, n \tag{7}$$

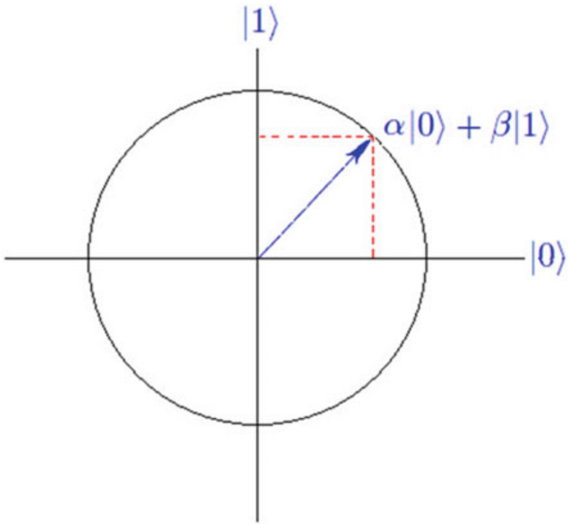


Fig. 5 Polar plot for q-seeds

where α and β include random integers that characterize the probability of the related states. $|\alpha|^2$ and $|\beta|^2$ are the probability that the Q-bit $|\psi\rangle$ can be reached in the "0" and "1" positions, respectively. Therefore, they satisfy the relation $|\alpha|^2 + |\beta|^2 = 1$.

Quantum Invasive Weed Optimization (Q-IWO), like any other optimization algorithm, depends on the illustration of the population, the fitness function, and the population dynamics.

In the initialization step, the quantum median value of the interval [0, 1] is selected as one of the initial population. So, the initializing step is made by a kind of pseudo-scholastic method.

$$\psi_1 = \frac{1}{\sqrt{2}}[1\ 1] \tag{8}$$

Next step us to normalize the randomly generated seeds in the interval [0, 1]:

$$|\psi\rangle_{[0,1]} = \frac{|\psi\rangle - \min(|\psi\rangle)}{Max(|\psi\rangle) - \min(|\psi\rangle)} \tag{9}$$

where, $Max(|\psi\rangle)$ and $\min(|\psi\rangle)$ describe the upper and the lower bounds respectively. Each individual can be obtained as follows:

$$\beta_j(t) = \sqrt{1 - \alpha_j(t)^2} \tag{10}$$

where, *rand* describes a random value between 0 and 1.

In the ordinary IWO, the possible positions for the seeds generate a vector space of dimensions; that is there is only one possible state. However, in a QIWO, the obtained state space has dimensions. It is this exponential growth of the state space with the number of population that puts in a proposal a possible exponential speed-up of calculation on quantum computers over classical computers [27].

4.2 Quantum Gates

Quantum gates are usually presented as matrices. The quantity for Q-bits in the input and the output of the gate should be equal. In this work for single Q-bits, the *Pauli-X gate* is employed which can be considered as the quantum equivalent of a *NOT* gate. It maps $|0\rangle$ to $|1\rangle$ and $|1\rangle$ to $|0\rangle$ as follows:

$$|\psi\rangle = \begin{bmatrix} \alpha_j \\ \beta_j \end{bmatrix}$$
$$\langle\psi| = I_{2\times 1} - \begin{bmatrix} \alpha_j \\ \beta_j \end{bmatrix} = \begin{bmatrix} \beta_j \\ \alpha_j \end{bmatrix} \tag{11}$$

where, $|\psi\rangle$ and $\langle\psi|$ are the initial and the explored Q-seeds.

After applying the quantum computation into the seeds, they should be transformed into the state to continue with the standard IWO. To transform the Q-seeds into standard seeds, we used quantum measurement technique as follows:

$$P(m) = \|\psi\|_2 \tag{12}$$

Afterward, the algorithm uses the following formula to put the result it in the considered range as [a, b]:

$$\Phi = \mathrm{a} + (\mathrm{b} - \mathrm{a}) \times \Psi \tag{13}$$

This flowchart of the QIWO is given in Fig. 6.

5 Optimizing ANN Weights Based on QIWO Algorithm

For the optimal selection of the neural network's weights, an exploration search problem can be modeled. The optimal quantity of the weights may be described as being a train with a particular length and the network is wholly encoded by approximation of the values of the weights in the network in the weed.

For achieving the optimized weights in neural networks, the following steps should be applied:

1. Procreate the initial weeds of *N* number of weights.
2. Calculate the cost for the EANN.

1) Start
2) Initializing the quantum weeds
3) Calculate the cost function for the weeds
4) Transformation to the quantum space
5) Apply fast non-dominated sorting
6) Procreate weeds based on tournament selection and IWO parameters
7) Calculate the cost values of the child weeds
8) Merge parent and child weeds
9) Apply fast non-dominated sorting
10) Generate next generation based on crowding distance and rank
11) Go to 7
12) Transformation to the normal space
13) End

Fig. 6 Flowchart of the QIWO

3. Due to the fitness function and adopting proper selection methods, regenerate some seeds for the weeds in the present generation.
4. Perform QIWO parameters to the generated seeds and achieve the next generation.
5. Check whether the network has obtained the needed error rate or the special numbers of generations have been obtained then go to step 3.
6. End.

Consider a two-layered network which is formulated as below:

$$\sum_{i=1}^{H} w_i f\left(\sum_{j=1}^{d} w_j x_j + b\right) \tag{14}$$

where, w and b are weights and bias of the network, H is the quantity of the neurons in the hidden layer, and f determines the activation function for the neurons which in this case is considered as sigmoid.

The optimization algorithm is adopted to select the optimal vale of weights to the nodes' interconnection and bias terms until the achieved values for the output layer neurons are as close as possible to the actual outputs. The mean squared error (MSE) for the network is formulated in the following.

$$MSE = \frac{1}{2}\sum_{k=1}^{g}\sum_{j=1}^{m}\left(Y_j(k) - T_j(k)\right)^2 \tag{15}$$

where, $Y_j(k)$ describes the desired output, $T_j(k)$ is the real output, and m determines the number of output nodes, g is the number of training samples.

6 Mathematical Morphology

The term morphology generally refers to a branch of science that speaks of the shape and structure of the body. Here the term is used with the content of mathematical morphology and as a tool for extraction of image components. This tool is very useful in presenting and describing the shape of areas and features such as borders, skeleton and body convex [26, 28].

One reason for the use of gray-level morphology operations is to make threshold level selection easier to a binary image. This eliminates the noise or dark artifacts in the image.

In this research, some different mathematical morphology such as region filling, area opening, and closing are utilized to eliminate the suspect defects. Here, the mathematical filling is made by combining some different operations including dilation, complementation, and intersections. The formulation for this operation is given below:

$$X_k = (X_{k-1} \oplus B) \cap A^c, \quad k = 1, 2, 3 \ldots \tag{16}$$

where, B and A describe the structuring element and the set of boundaries, respectively. The operation stops when $X_k = X_{k-1}$.

The opening of A by B is performed based on a dilation followed by erosion, i.e.

$$A \circ B = (A \Theta B) \oplus B \tag{17}$$

Area opening is utilized to remove small area blemishes which can be not considered by the cancer images.

The closing operator smoothes the counters, removes small holes, fuses narrow breaks, fills gaps in the contour, and long thin gulfs. The closing of set A by structuring element B described by $A \cdot B$ and mixed thin distances as below:

$$A \cdot B = (A \oplus B) \Theta B \tag{18}$$

Closing then might result in mixtures of disconnected components that generate new holes. Note that in this paper, the structural element is selected as a single 5×5 cube.

7 Input Datasets

To do this research, we have used images from two databases that are among the standard and valid information sources in this field:

1. DermIS Digital Database: An authoritative source for using by the medical research community in the image analysis. This database is the largest online

Fig. 7 The diversity of TLM and XLM images

information service available on the Internet which provides elaborate images with differential diagnosis and diagnosis, and more information on almost all skin conditions.

2. Dermquest Database: Online medical resources for dermatologists and dermatologist-based healthcare professionals. All images of this website are reviewed and approved by the famous international editorial board. Figure 7 shows some examples of different images for TLM and XLM.

8 Experimental Results

As it is presented, DermIS and Dermquest are employed as the popular skin cancer databases. The main idea in this study is to recognize the melanoma region in the skin from the input dermoscopy images. Figure 8 shows some examples of the simulated operation sequence and the final stage of identifying the cancer region for the input images. As indicated in Fig. 8, the proposed method performs the detection

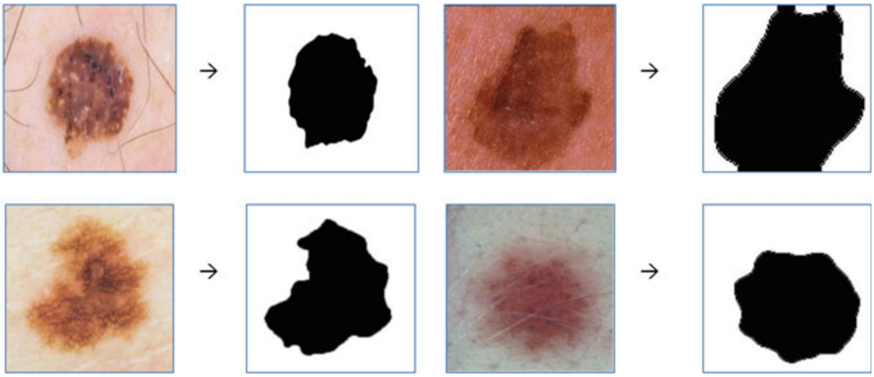


Fig. 8 Some of the results of the proposed algorithm

function satisfactorily. The presented method is applied to multimodal skin cancer images. From Fig. 8, we can see that the proposed system illustrates good results for most of the different databases. To check the accuracy of the proposed method, four parameters have been selected: The first parameter is the correct detection rate (CDR) whose formula is shown below:

$$CDR = \frac{Number\,of\,corrected\,classified\,pixels}{Total\,number\,of\,Pixels\,of\,the\,test\,dataset} \quad (19)$$

The second parameter is the FAR, which is the percentage of selected elements that have been mistakenly selected. In other words, in this project, images that are not cancerous but are mistakenly identified as cancerous. This rate is obtained by the following formula:

$$FAR = \frac{Number\,of\,non-cancer\,pixels\,classified\,as\,cancer}{Total\,number\,of\,Pixels\,of\,the\,test\,dataset} \quad (20)$$

The last measurement parameter is the FRR. This parameter basically represents the images that had a cancerous lesion, but the system did not recognize them. The formula for this parameter is as follows:

$$FRR = \frac{Number\,of\,cancer\,pixels\,classified\,as\,non-cancer}{Total\,number\,of\,Pixels\,of\,the\,test\,dataset} \quad (21)$$

For evaluating the presented algorithm performance, we compare the proposed QIWO based algorithm by the ordinary neural network and also PSO based neural network.

From Table 1, it is clear that final results are very acceptable and have a low error rate. The accuracy of the proposed model is suitable and therefore its reliability is high. It should be noted that the proposed model for early diagnosis of skin cancer is very useful for specialists and radiologists and can help them determine the location of the lesions. The performance accuracy of the proposed segmentation algorithm is illustrated in Table 1.

Table 1 shows the results from the comparison. In this comparison, the proposed algorithm has better performance than the ordinary ANN and the optimized ANN

Table 1 Performance comparison for the QIWO based, PSO based and ordinary neural networks

Metric	Performance evaluation based on QIWO based technique	Performance evaluation based on PSO based technique	Performance evaluation based on ordinary neural network
CDR (%)	87	85.3	80
FAR (%)	8.5	7.7	13.3
FRR (%)	4.5	7	6.3

by the popular PSO algorithm. The results also show that the algorithm has higher precision for complex images.

9 Conclusion

Skin cancer is one of the most commonly diagnosed diseases in the world. In this paper, based on the necessity of early and timely diagnosis of this disease, a new method based on an optimized neural network has been proposed which is used to diagnose and determine the exact location of the cancerous area. According to the proposed model, the masses are completely separated from other parts of the image and their quality and brightness have been increases so that the location and size of the mass in a given image are clearly and precisely determined. The final results showed acceptable accuracy for the presented method.

To improve the presented segmentation method at its best, neural network weights are optimized using the QIWO algorithm. The final results are compared with the ordinary neural network and the PSO algorithm optimized neural network as a widely used method to show the system efficiency.

References

1. Adam S, Likas A, Vrahatis M (2017) Interval analysis based neural network inversion: a means for evaluating generalization. Paper presented at the international conference on engineering applications of neural networks
2. Ahangarpour A, Farbod M, Ghanbarzadeh A, Moradi A, MirzakhaniNafchi A (2018) Optimization of continual production of CNTs by CVD method using radial basic function (RBF) neural network and the Bees algorithm. J Nanostruct 8(3):225–231
3. Ahmed IO, Ibraheem BA, Mustafa ZA (2018) Detection of eye melanoma using artificial neural network. J Clin Eng 43(1):22–28
4. Barata C, Ruela M, Francisco M, Mendonça T, Marques JS (2014) Two systems for the detection of melanomas in dermoscopy images using texture and color features. IEEE Syst J 8(3):965–979
5. Boyat AK, Joshi BK (2015) A review paper: noise models in digital image processing. arXiv preprint arXiv:1505.03489
6. Bozorgtabar B, Sedai S, Roy PK, Garnavi R (2017) Skin lesion segmentation using deep convolution networks guided by local unsupervised learning. IBM J Res Dev 61(4):6:1–6:8
7. Emre Celebi M, Alp Aslandogan Y, Stoecker WV, Iyatomi H, Oka H, Chen X (2007) Unsupervised border detection in dermoscopy images. Skin Res Technol 13(4):454–462
8. Emre Celebi M, Kingravi HA, Iyatomi H, Alp Aslandogan Y, Stoecker WV, Moss RH et al (2008) Border detection in dermoscopy images using statistical region merging. Skin Res Technol 14(3):347–353
9. Erkol B, Moss RH, Joe Stanley R, Stoecker WV, Hvatum E (2005) Automatic lesion boundary detection in dermoscopy images using gradient vector flow snakes. Skin Res Technol 11(1):17–26
10. Esteva A, Kuprel B, Novoa RA, Ko J, Swetter SM, Blau HM, Thrun S (2017) Dermatologist-level classification of skin cancer with deep neural networks. Nature 542(7639):115–118

11. Gao J, Zhang J, Fleming MG (2000) A novel multiresolution color image segmentation technique and its application to dermatoscopic image segmentation. Paper presented at the 2000 international conference on image processing, 2000. Proceedings
12. García-Lamont F, Cervantes J, López-Chau A, Ruiz S (2018) Contrast enhancement of RGB color images by histogram equalization of color vectors' intensities. Paper presented at the international conference on intelligent computing
13. Hardie RC, Ali R, De Silva MS, Kebede TM (2018) Skin lesion segmentation and classification for ISIC 2018 using traditional classifiers with hand-crafted features. arXiv preprint arXiv:1807.07001
14. Jahanifar M, Tajeddin NZ, Gooya A, Asl BM (2017) Segmentation of lesions in dermoscopy images using saliency map and contour propagation. arXiv
15. Kennedy J (2011) Particle swarm optimization encyclopedia of machine learning. Springer, pp 760–766
16. Khalilpuor M, Razmjooy N, Hosseini H, Moallem P (2011) Optimal control of DC motor using invasive weed optimization (IWO) algorithm. Paper presented at the Majlesi conference on electrical engineering, Majlesi town, Isfahan
17. Leung VKY, Dobbinson SJ, Goodman DJ, Kanellis J, Chong AH (2017) Skin cancer history, sun-related attitudes, behaviour and sunburn among renal transplant recipients versus general population. Australas J Dermatol
18. Moallem P, Razmjooy N, Mousavi B (2014) Robust potato color image segmentation using adaptive fuzzy inference system. Iran J Fuzzy Syst 11(6):47–65
19. Mousavi BS, Soleymani F (2014) Semantic image classification by genetic algorithm using optimised fuzzy system based on Zernike moments. SIViP 8(5):831–842
20. Mousavi BS, Soleymani F, Razmjooy N (2013) Color image segmentation using neuro-fuzzy system in a novel optimized color space. Neural Comput Appl 23(5):1513–1520
21. Pandey S, Solanki A (2019) Music instrument recognition using deep convolutional neural networks. Int J Inf Technol. https://doi.org/10.1007/s41870-019-00285-y
22. Parolin A, Herzer E, Jung CR (2010) Semi-automated diagnosis of melanoma through the analysis of dermatological images. Paper presented at the 2010 23rd SIBGRAPI-conference on graphics, patterns and images
23. Queen L (2017) Skin cancer: causes, prevention, and treatment
24. Rashid Sheykhahmad F, Razmjooy N, Ramezani M (2015) A novel method for skin lesion segmentation. Int J Inf Secur Syst Manag 4(2):458–466
25. Razmjooy N, Khalilpour M (2015) A new design for PID controller by considering the operating points changes in hydro-turbine connected to the equivalent network by using invasive weed optimization (IWO) algorithm. Int J Inf Secur Syst Manag 4(2):468–475
26. Razmjooy N, Mousavi BS, Soleymani F (2012) A real-time mathematical computer method for potato inspection using machine vision. Comput Math Appl 63(1):268–279
27. Razmjooy N, Ramezani M (2014) An improved quantum evolutionary algorithm based on invasive weed optimization. Indian J Sci Res 4(2):413–422
28. Razmjooy N, Ramezani M, Ghadimi N (2017) Imperialist competitive algorithm-based optimization of neuro-fuzzy system parameters for automatic red-eye removal. Int J Fuzzy Syst 19(4):1144–1156
29. Schmid P (1999) Lesion detection in dermatoscopic images using anisotropic diffusion and morphological flooding. Paper presented at the 1999 international conference on image processing, 1999. ICIP 99. Proceedings
30. Simon SS, Joseph XF (2017) Pre-processing of dental X-ray images using adaptive histogram equalization method
31. Suresh K, Kundu D, Ghosh S, Das S, Abraham A (2009) IWO with increased deviation and stochastic selection (IWO-ID-SS) for global optimization of noisy fitness functions. Paper presented at the world congress on nature & biologically inspired computing, 2009. NaBIC 2009
32. Tang J (2009) A multi-direction GVF snake for the segmentation of skin cancer images. Pattern Recogn 42(6):1172–1179

33. Venkatesh G, Naresh Y, Little S, O'Connor NE (2018) A deep residual architecture for skin lesion segmentation. In: OR 2.0 context-aware operating theaters, computer assisted robotic endoscopy, clinical image-based procedures, and skin image analysis. Springer, pp 277–284
34. Yahya A, Tan J, Su B, Liu K, Hadi A (2017) Image noise reduction based on applying adaptive thresholding onto PDEs methods. J Eng 1(1)
35. Yuan Y, Chao M, Lo Y-C (2017) Automatic skin lesion segmentation using deep fully convolutional networks with Jaccard distance. IEEE Trans Med Imaging 36(9):1876–1886

A Computational Intelligence Perspective on Multimodal Image Registration for Unmanned Aerial Vehicles (UAVs)

Vania V. Estrela, Navid Razmjooy, Ana Carolina Borges Monteiro, Reinaldo Padilha França, Maria A. de Jesus, and Yuzo Iano

Abstract Remote Sensing (RS) applications generally require robustness, stability, accuracy, promptness, and a high autonomy level to simplify the Big Data (BD) processing in real-time. Image Registration (ImR) is among the most employed RS tasks. ImR transforms different groups of images into a coordinate system that allows overlaying two or more images from the same scene acquired with various sensors and/or taken at different times and angles. The original imageries must be normalized and geometrically aligned to create an ample image containing information from all the separate images. ImR is a crucial step when one has several views and a myriad of sensors that must be fused. The BD aspect of Multimodal Image Registration (MIR) is related to the idea of multispectral and hyperspectral imaging, which involve a vast amount of frequency bands. BD from different sources assist the decision-making processes and create additional more massive datasets for the long-term tracking of various phenomena. This chapter focuses on the MIR from infrared and optical sensors relying on the Particle Swarm Optimization (PSO) class of algorithms. These computational intelligence procedures circumvent problems related

V. V. Estrela (✉) · M. A. de Jesus
Department of Telecommunications, Fluminense Federal University (UFF), Duque de Caxias, RJ CEP 25086-132, Brazil
e-mail: vania.estrela.phd@ieee.org

M. A. de Jesus
e-mail: majesus1977br@gmail.com

N. Razmjooy
Department of Electrical Engineering, University of Tafresh, Tafresh, Iran
e-mail: navid.razmjooy@hotmail.com

A. C. B. Monteiro · R. P. França · Y. Iano
Nuclear Instrumentation Laboratory, Federal University of Rio de Janeiro, Centro de Tecnologia (CT), Bloco I, sala I-133. Ilha do Fundão, Rio de Janeiro, RJ CEP 21941-972, Brazil
e-mail: monteiro@decom.fee.unicamp.br

R. P. França
e-mail: padilha@decom.fee.unicamp.br

Y. Iano
e-mail: yuzo@decom.fee.unicamp.br

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_13

to multiresolution methods and the high computational cost of hard optimization methods.

Keywords Multimodal image registration · Remote sensing · Intelligent agents · Hybrid algorithms · Unmanned aerial vehicle · Particle swarm optimization · Surveillance · Image fusion

1 Introduction

Ideally, Remote Sensing (RS) applications require accuracy, stability, robustness, speed, and a high level of autonomy to expedite the processing of enormous volumes of data in real-time [1–4].

Image Registration (ImR) is the procedure of transforming different image sets into a suitable and more general coordinate system. This strategy permits to overlay two or more images from various probing equipment or sensors at different times and angles, or the same scene to geometrically normalize and align them for analysis is a crucial step where information from multiple images must be combined everywhere. When the images and objects they represent have different scales and viewpoints, they are registered, after geometrical transformations for alignment and merging. This technique is widely used with the objective of automatically find the pixel transformation from the original data into the normalized reference images, to match them according to their features. Moreover, the interpolation of values between these feature points must refer to the same anatomical point (feature point correspondence).

In medical imaging, Multimodal Image Registration (MIR) allows the combination of data from many modalities, such as Computed Tomography (CT), Magnetic Resonance (MR), Single Photon Emission Computed Tomography (SPECT) or Positron-Emission Tomography (PET). Having information from different types of sensors helps to expand the patient's medical knowledge. A typical example consists of monitoring the growth of a tumor for treatment assessment, appraising improvements in interventions, or even comparing patient data with an anatomical atlas. Still considering its importance in this area since with the increasing number of different types of data acquisition devices and their increasing availability, the option of creating a disease database is viable.

ImR contributes to the combination of the more patient's health evidence from different modalities or platforms (e.g., CT and SPECT) covering anatomical, functional or molecular methodological aspects of imaging. Furthermore, multimodal imaging gives a better estimate of volumes, body functions, morphology, and anatomy, enabling the characterization of tissues, intra-cardiac masses, pericardial diseases, and other diseases.

Additionally, multimodality imaging plays an essential role in the exact identification of diseased and healthy tissues utilized for treatment planning with superior soft tissue definition (MRI) and is useful in the identification of disease at metabolic level (PET) even before being visible on MRI. The image registration proves to be valuable for treatment planning purposes.

Multimodality imaging joins different probing modalities with a better solution to circumvent the limitations of independent techniques by providing a wealth of material for each preclinical experiment. The need for combining morpho-functional information can be addressed by obtaining images at different times (asynchronous) or even simultaneous (synchronous) image acquisition, for automatic merging.

Hyperspectral imaging (HSI) is an increasingly popular imaging practice that employs analytical tools combined with the 2-D visualization of objects gotten by optical imaging [5]. HSI analyzes a broad spectrum of light rather than merely assigning the red, green, and blue primary colors to a pixel. An image pixel encloses spectral evidence, which can help to recuperate the third dimension (depth) of values to the 2-D spatial image, thereby generating a 3-D data cube, called a hypercube or image cube.

In HSI image acquisition, where each pixel has a set of information data within certain spectral bands. Where this set of images carries the pro-pixel information close to the collected one, and can then create multidimensional maps of total hemoglobin concentration or even hemoglobin oxygen saturation. It is imperative to note that an HSI image differs from a Multispectral (MSI) image by its high resolution and a more substantial amount of frequency bands, where the second gets images of particular parts of the electromagnetic spectrum, and the first collects object images within a series of spectral windows.

Therefore, this chapter aims to provide an updated computational intelligence perspective related to multimodal image registration present in drone technology, categorizing, and synthesizing the potential of technologies that involve thematic. Thus, Sect. 2 explains Image Registration. Multimodal Image Registration (MIR) is analyzed in Sect. 3. Section 4 talks over the MIR for Visible as well as Infrared Images in Unmanned Aerial Vehicles (UAVs). Section 5 examines the algorithms regarding computational intelligence in ImR. Lastly, Sect. 6 discusses research, and Sect. 7 outlines future trends in technology. Finally, Sect. 8 closes the chapter.

2 Image Registration

ImR aligns two or more images, among them or relative to another data source, like a map containing vector data for instance. A mathematical transformation standardizes geometric differences (normalization) in imageries to implement this alignment. To be registered, images should encompass overlapping views of matching ground features. In the underlying case, one image may need to be translated only, or translated with rotation for the sake of image alignment. ImR entails locating and matching equivalent regions from imageries to be registered. In manual ImR, an operator performs the assignments visually with interactive software. Unfortunately, in RS, computerized procedures are not dependable and correct all the time, which leads to manual registration quite often. The user extracts distinctive locations from both images (aka Control Points (CPs), reference points or tie-points).

First, the CPs from both images (and datasets) are matched interactively to accomplish correspondence. Next, corresponding CPs help to figure out the parameters of the necessary geometric transformation. Most existing commercial frameworks follow this ImR methodology. Manual CP selection represents, however, a repetitive, painstaking, and time-intensive task that comes to be prohibitive for massive amounts of data. Likewise, since the interactive selection of CPs in satellite images is occasionally tricky thanks to excessively few points, inexact points, or ill-distributed points. Manual ImR can augment registration errors.

The foremost ImR goal is to raise the accurateness, robustness, and efficiency of entirely automatic, algorithmic approaches to the imagery fusion problem. Usually, automatic ImR algorithms embrace three phases [6]:

(1) Extraction/identification of distinct sections, or characteristics, to be matched.
(2) Finding equivalent features (feature matching) that is the transformation responsible for the best imagery alignment.
(3) Resampling an imagery to obtain a new one in the normalized coordinate system grounded on the computed transformation.

Automatic ImR differs from manual ImR in several ways. One algorithm may mine simple features but perform this task via a complex matching scheme, while another may utilize rather intricate features, but then work with a relatively simple matching strategy. When several data sources are fused, the resampling step can be replaced or supplemented by the ImR process. Finally, automatic strategies may comprise two resampling stages. Employing a temporary stage during matching augments the similarity among image patches, but its outcomes can be thrown away while a second, more accurate code segment delivers the final image product.

Ideally, RS applications require accuracy, stability, robustness, speed, and a high level of autonomy that will ease the processing of outsized amounts of records in real-time. This chapter examines the specific concerns associated to InR for UAVs, and to describe the recommended methods to solve these issues [6].

2.1 Image Normalization and Orthorectification

Normalization aligns and warps MTI data in a generic anatomical model, where it is usually performed with knowledge from multiple subjects. It is a more painstaking process since evidence from various subjects must be aligned with or without substantial image warping and deformations (which often occur while trying to squeeze different profiles into a common model or reference space). To make an analogy, it is similar to folding clothes into a suitcase. Consequently, a model for each organ requires transformations in size, shape, and dimensions by normalization (or warping) to a template with standard dimensions in addition to coordinates.

Most researchers rely on these transformations when reporting their results, so these dimensions and coordinates can also be called standardized space.

In medical ImR and fusion, processes are considered a valuable assistant to medical specialists, as these processes may assist specialists in diagnostics, empowering them to monitor disease progress and create decision-making power in the necessary therapies regarding the patient's condition.

It is routine that numerous functional and anatomical functions of the images appear throughout disease investigation, where the images show organs in different orientations due to the variable positioning of the subject, being possible a visual or numerical comparison of these heterogeneous images. Thus, superior outcomes can result from the transport of all imageries to a common frame of reference starting the current evaluation with corresponding images. If ImR is viable, then the matches the produce more informative and better descriptive images.

ImR and recording outcomes become a way to assist and extract meaning from imagery in many application areas, e.g., from biomedical engineering, and geographic information (GIS) in machine vision domains to weather forecasting in RS systems. These processes interpret routine multiple functional and anatomical images acquired in several stages of a disease investigation.

Once the imageries reference a common space, then it is possible to evaluate the results on a pixel-by-pixel basis. Among the leading available fusion options, one has

- Alpha blending, with an exchange of isocontours;
- Moving spyglass;
- Rendering of the final ImR obtained from a combination of several fused images;
- Overlay of image parts above a threshold set by the operator;
- Rendering of fused images with possible rotation and different viewpoints;
- Fused and original synchronized images visualized in parallel;
- Volume-of-Interest (VOI) definition straightforwardly from fused images and, ultimately, saving fused images as DICOM objects for later research.

2.2 Similarity Metrics

An Image Similarity (IS) measure quantifies the similarity between intensity patterns within related images. The choice of an IS measure is contingent on the sensing modality. Common IS examples include cross-correlation, Mutual Information (MI), the sum of squared intensity differences, and ratio image uniformity. MI and Normalized MI (NMI) are the most prevalent IS metrics for multimodal ImR. The other metrics are utilized for ImR within the same modality.

Figure 1 shows that many new features result from cost functions associated with matching methods via large deformations.

Given a *reference* image, $I_1(x, y)$, and an acquired image $I_2(x, y)$, find the mapping (Tp, g) that best transforms I_1 into I_2, i.e., where Tp denotes spatial mapping and g denotes a radiometric mapping.

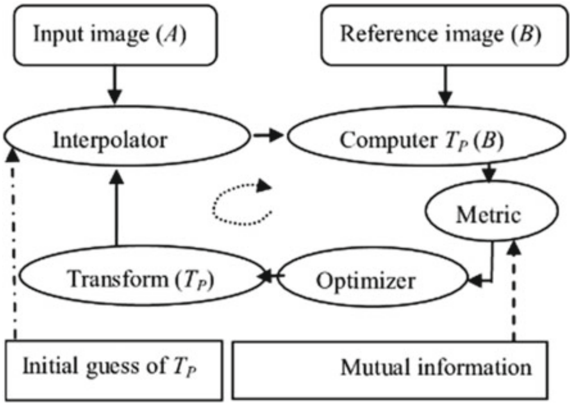


Fig. 1 The rationale behind UAV image registration

- **Spatial transformations** such as (translation, rigid, affine, projective, and perspective).
- **Radiometric transformations (resampling)**:
  - Bilinear, nearest neighbor, cubic convolution, spline

$$I_2(x, y) = g(I_1(Tp(x, y), Tp(x, y))), \tag{1}$$

  - Minimize the Sum of Squared Errors (SSD) along the overlapping sub-images

$$\mathrm{SSD}(x, y) = \sum_{m=1}^{M}\sum_{n=1}^{N}[\mathbf{I}(m, n) - \mathbf{I}'(m - x, n - y)]^2. \tag{2}$$

- **Cross-correlation**:
  - Maximize *cross-correlation* along the overlapping sub-images. It represents how correlated (in the least squares sense) the two data sets are. It is defined as:

$$\mathrm{CC} = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}[\boldsymbol{I}(i, j) - \mu_I]\left[\boldsymbol{I}'(i, j) - \mu_{I'}\right]}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}[\boldsymbol{I}(i, j) - \mu_I]^2}\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}\left[\boldsymbol{I}'(i, j) - \mu_{I'}\right]^2}} \tag{3}$$

  where μ_I and $\mu_{I'}$ are the mean of images of $\boldsymbol{I}(i,j)$ and $\boldsymbol{I}'(i,j)$, respectively. The maximum absolute value of *CC* is 1 and signifies perfect correlation.

- **Normalized cross-correlation (NCC)**:
  - Maximize normalized cross-correlation

$$\mathrm{NCC}(x, y) = \frac{\sum_{m,n}\left[\mathbf{I}(m,n) - \bar{\mathbf{I}}_{x,y}\right]\left[\mathbf{I}'(m-x, n-y) - \bar{\mathbf{I}}'\right]}{\left\{\sum_{m,n}\left[f(m,n) - \overline{f}y\right]^2 \sum_{m,n}\left[\mathbf{I}'(m-x, n-y) - \bar{\mathbf{I}}'\right]^2\right\}^{\frac{1}{2}}}. \tag{4}$$

- **Mutual Information (MI)**:
  – Maximizes the degree of statistical dependence between the images

$$\mathrm{MI}(\mathbf{I}, \mathbf{I}') = \sum\sum p_{\mathbf{I},\mathbf{I}'} \log\left[\frac{p_{\mathbf{I},\mathbf{I}'}}{p_{\mathbf{I}'} p_{\mathbf{I}'}}\right] \tag{5}$$

where M is the sum of all histogram entries, i.e., the number of pixels in the overlapping sub-images.

3 Multimodal Image Registration

Multimodal Image Registration (MIR) integrates data from different sources for the decision-making process and to attain more massive datasets for the long-term tracking of various phenomena. Change detection over time or scale is only possible if multi-sensor and multi-temporal data are correctly calibrated during registration and data can be extrapolated throughout some scales, whether spatially, spectrally, or temporally [6]. MIR is indispensable for UAVs, which are often equipped with radars, video/image cameras, and several types of sensors. MIR can provide situation understanding to avoid potential danger or threat and often relies on networked settings like the Internet of Things (IoT), and sensor networks. Typical mining tasks include semantic extraction, object recognition, tracking, and so on [7–9].

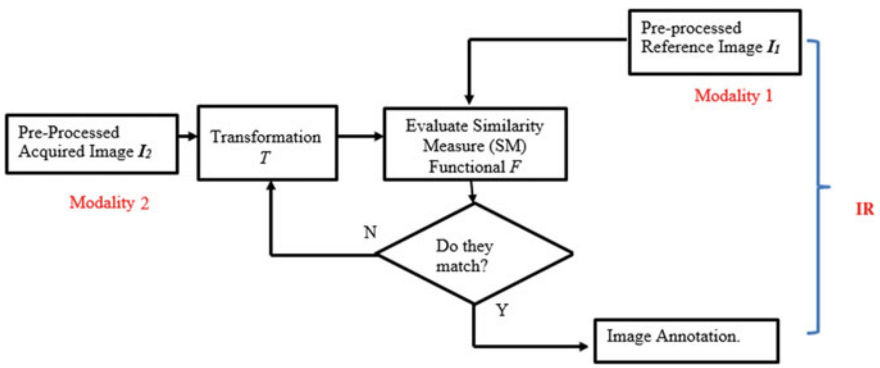


Fig. 2 Image registration framework

Figure 2 illustrates the ImR reasoning. A pre-processing stage ameliorates the contrast of each input imagery modality removing artifacts or poor contrast that may degrade diagnosis [10]. The regularly applied pre-processing algorithms rely on histogram methodologies, transform-based tactics, unsharp masking techniques, and filtering-based algorithms.

MIR combines an evolving imagery $\boldsymbol{I_2}$ with a reference picture $\boldsymbol{I_1}$ to obtain data that are more detailed or some specific features and it comprises three steps [6]:

(1) Extraction of distinct regions (sub-images), or features, to be matched taken with distinct modalities for better decision and management,

(2) Matching of the features by searching for a transformation $\boldsymbol{T} = (\boldsymbol{Tp}, \boldsymbol{g})$ that best aligns them. Given $\boldsymbol{I_1}(x, y)$ and $\boldsymbol{I_2}(x, y)$ find the spatial mapping $\boldsymbol{Tp}$, and the radiometric (resampling) mapping $\boldsymbol{g}$ that best transforms $\boldsymbol{I_1}$ into $\boldsymbol{I_2}$, such as

$$\boldsymbol{I_2}(x, y) = \boldsymbol{g}(\boldsymbol{I_1}(\boldsymbol{Tp}(x, y), \boldsymbol{Tp}(x, y)))$$

This work handles a rigid transformation model [i, g] that includes linear global transformations, such as rotation, scaling, translation, and other affine transforms, expressed by

$$\mathbf{T}_{Global} = \mathbf{Rx} + \mathbf{t},$$

where the 4×4 matrix $\boldsymbol{R}$ carries all the necessary facts to linearly map the $\boldsymbol{I_2}$ to a standard form representation, $\boldsymbol{x}$ is a set of coordinates, and $\boldsymbol{T_{Global}}$ is a vector containing homogeneous coordinates (refer to Fig. 3).

(3) Resampling one imagery to create a new one in the coordinate system of the other, based on the computed $\boldsymbol{T}$ (refer to Fig. 2). Then, $\boldsymbol{I_1}$ and $\boldsymbol{I_2}$ help to calculate a parameter set iteratively to optimize the cost function F, aka IS Metric (ISM), to compare the images registered under the current parameter set $\boldsymbol{T}$. If the images

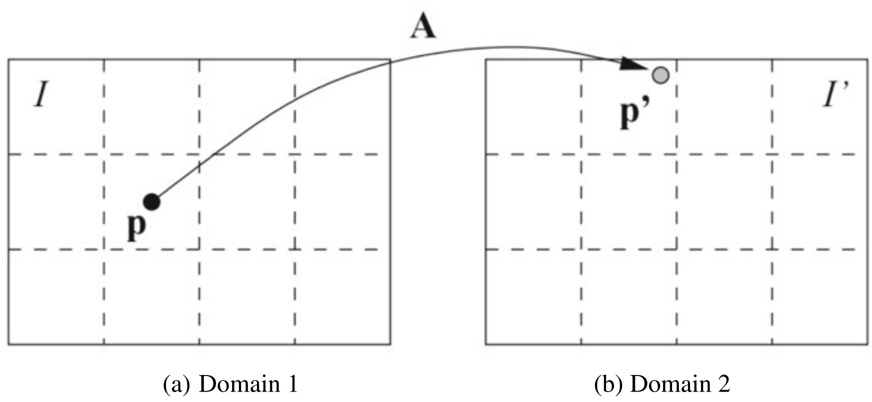


Fig. 3 2-D rigid transformation from the spatial domain to other domain

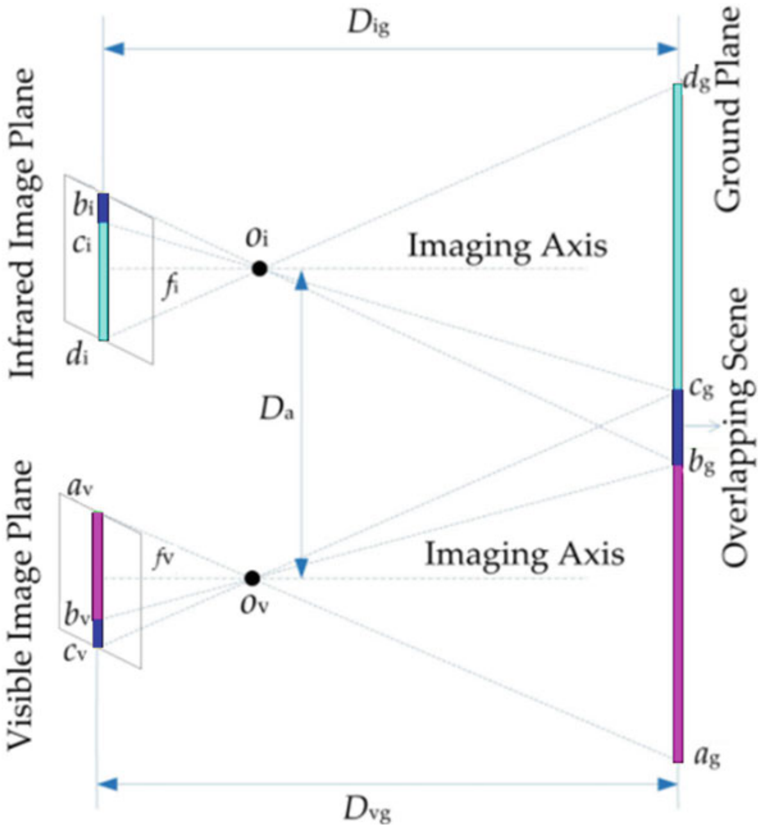


Fig. 4 Combination of infrared and color cameras

are not registered, an optimization scheme will correct the parameters, and a fresh iteration will commence as displayed in Fig. 2 [10] (Fig. 4).

An ISM quantifies similarities among images [11], its choice affects the cost function, and it is contingent on the images' modalities to be registered. This work uses the Mutual Information (MI) as the SM, and it maximizes the degree of statistical dependence between the images:

$$\mathrm{MI}(\mathbf{I}_1, \mathbf{I}_2) = \sum_{\mathbf{g}_1} \sum_{\mathbf{g}_2} p_{\mathbf{I}_1,\mathbf{I}_2}(\mathbf{g}_1, \mathbf{g}_2) \cdot \log\left(\frac{p_{\mathbf{I}_1,\mathbf{I}_2}(\mathbf{g}_1, \mathbf{g}_2)}{p_{\mathbf{I}_1}(\mathbf{g}_1) \cdot p_{\mathbf{I}_2}(\mathbf{g}_2)}\right),$$

where $\boldsymbol{g_1}$ and $\boldsymbol{g_2}$ are overlapping sub-images [12, 13] (Fig. 5).

MIR has two main problems: (i) the choice of optimization methods, and (ii) the need of very decent initial values and optimization constraints to evade the local minimum because the transformation parameters are usually nonconvex and irregular.

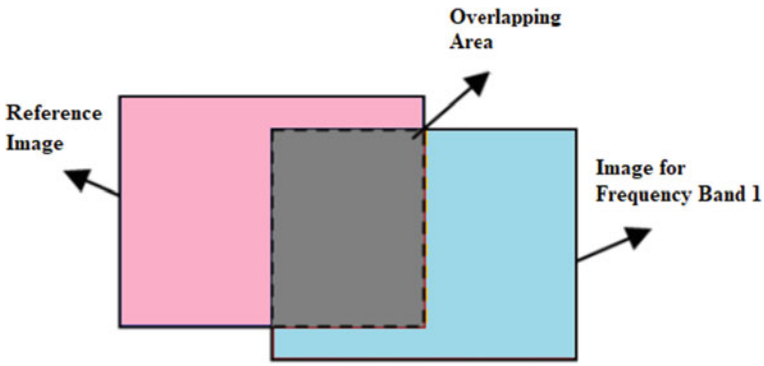


Fig. 5 Overlapping sub-images

Particle Swarm Optimization (PSO) remains a widespread alternative for solving intricate problems such as ImR [14], which are otherwise hard to solve by traditional optimization algorithms. This chapter scrutinizes the PSO algorithm family and, particularly, one of its hybrid varieties. Hybridization means combining two (or more) practices sensibly. Then, the consequential algorithm holds the positive characteristics of both (or all) the original procedures. Thought-provoking techniques worth using in hybridization embrace many local and global search schemes. Results from the usage of ImR are shown for the PSO and the HPSO algorithms [15].

4 Multimodal ImR for Visible and Infrared Images in UAVs

Various sensing categories capture different evidence for a UAV to help to understand events [16, 17]. Satellite image fusion can render images with more detailed geomorphic information, and multi-focus image fusion can produce a clearer image.

Visual and Infrared (VI) ImR (VIIR) can be widely used in surveillance systems. The visible images are fit for the Human Vision System (HVS) and possess a high spatial resolution but perform poorly without sufficient lighting (Fig. 4). The infrared data often reveals invisible details of objects to help comprehend the whole scene. Therefore, the ideal VIIR result should not only substantially integrate the significant infrared image features to reveal the potentially suspicious objects, but also preserve much rudimentary knowledge for a better scene perception, creation of augmented realities and remediate occlusion.

Algorithms to extract the salient features from VI images in some kind of scale-space can be integrated to produce the final IR image whose results are usually pleasant for the HVS. However, these algorithms often have a blurring effect or visual detail loss, which would influence the visual quality of the resulting fused images. Infrared and visual imagery fusion is mostly in need of supervising the low-light circumstance, under which the optical image encloses much visible material from

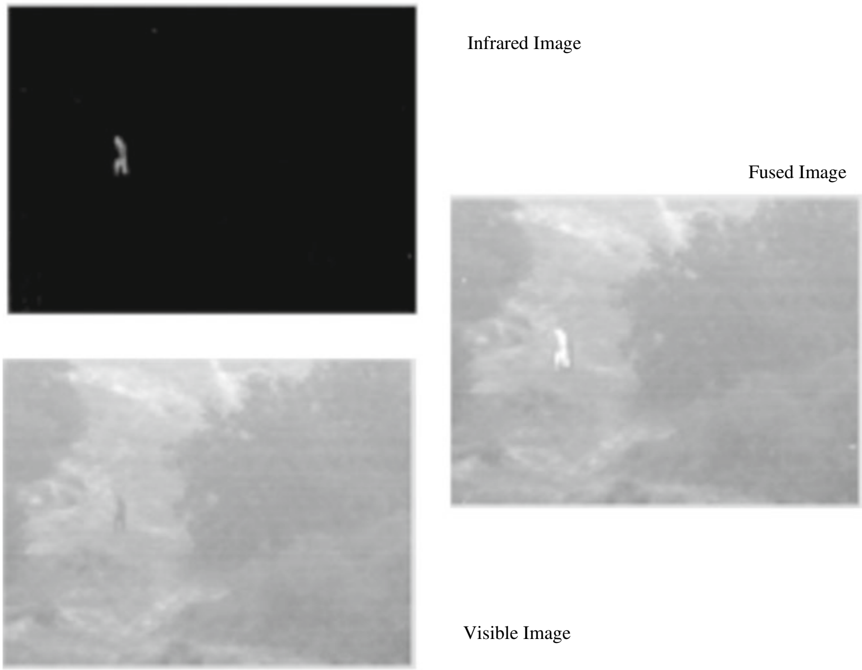


Fig. 6 General framework for the image registration algorithm

the surveyed scene, and the infrared map usually detects few invisible but critical data. Moreover, under the low light, the infrared and the visual images will often have low correlation, thus an intuitive way to combine both images is by directly injecting the bright infrared features to the visual image.

Following the effective infrared feature recovery in addition to ImR strategies, the produced fusion image could well join in the proper bright features from the infrared image. Most optical image information from also needs preservation. Figure 6 clarifies the basic idea underneath the proposed algorithm framework [16–18].

Numerous papers describing PSO UAV applications are present in the literature. Motion planning and control, path planning, UAV running, collective UAV search, unsupervised UAV learning, obstacle avoidance, UAV swarms, vision in UAVs, and environment mapping.

5 Computational Intelligence in ImR

Many metaheuristics [19, 20] techniques have been suggested to handle IR. Among them, Particle Swarm Optimization (PSO) as it is an arduous problem to unravel by traditional optimization algorithms. Subsequently, two variants of PSO are scrutinized [21–27].

5.1 Classical Particle Swarm Optimization (CPSO)

CPSO is a simple population-based optimization structure, which involves minimal computational effort [21]. It employs a search motivated by a paradigm of social influence and collective learning. Individuals emulate the achievements of their neighbors.

Consider a present diffuse population of size S known as a swarm. Each swarm member is dubbed a particle and it as a point belonging to the search space. A particle group has a tendency to cluster at an optimized position (maximum or minimum). Each particle corrects itself by comparing its previous SM to the SMs of its neighbors to reach the best result [21].

If $f(\boldsymbol{x_i}^{\boldsymbol{k}})$: $\mathbb{R}^n \rightarrow \mathbb{R}$ is the cost functional to be minimized, then the real number output corresponds to the result of the optimized fitness function, where $\boldsymbol{x_i}$ is the parameter vector to be estimated. Given an unknown gradient f of f, an optimum solution $\boldsymbol{x}^*$ for which $f(\boldsymbol{x}^*) \leq f(\boldsymbol{y})$, for all $\boldsymbol{y}$ in the search space, is sought. At iteration $k, \boldsymbol{x}_i \in \mathbb{R}^n$ is the position vector of the $\boldsymbol{i}$-th particle in the search space whose velocity is $\boldsymbol{v}_i \in \mathbb{R}^n, \boldsymbol{p_{besti}}$ is the best $\boldsymbol{x_i}$ for particle i, $\boldsymbol{g_{best}}$ is the global best-known position among all particles of the entire swarm, and w means the weight. c_1 and c_2 are acceleration constants whose pdfs are uniformly distributed in the interval sandwiched between *0* and *1*. $\boldsymbol{p_{besti}}$ is the maximum cost functional $f(\boldsymbol{x_i}^{\boldsymbol{k}})$ for each particle, and $\boldsymbol{g_{best}}$ corresponds to the best cluster.

5.1.1 CPSO-based ImR Algorithm

(1) Initialize the swarm randomly with initial values for variables $\boldsymbol{x_i}^{\boldsymbol{0}}$, $\boldsymbol{v_i}^{\boldsymbol{0}}$, $\boldsymbol{p_{besti}}$, $\boldsymbol{g_{best}}$
(2) For each particle i, repeat until the whole population has been analyzed:
(3) Set initial values for variables $\boldsymbol{x_i}^{\boldsymbol{0}}$, $\boldsymbol{v_i}^{\boldsymbol{0}}$, $\boldsymbol{p_{besti}}$, $\boldsymbol{g_{best}}$
(4) $\boldsymbol{v}_{\boldsymbol{i}}^{\boldsymbol{k+1}} = w\boldsymbol{v}_i^{\boldsymbol{k}} + c_1 \operatorname{rand}\left[\boldsymbol{p_{besti}} - \boldsymbol{x}_i^{\boldsymbol{k}}\right] + c_2 \operatorname{rand}\left[\boldsymbol{g_{best}} - \boldsymbol{x}_i^{\boldsymbol{k}}\right]$.
(5) $w^{(k+1)} = w^k + dw$.
(6) $dw = \frac{(w_{min} - w_{max})}{T}$.
(7) $x_{\mathbf{i}}^{\mathbf{(k+1)}} = \mathbf{x_i^k} + \mathbf{v_i^{k+1}}$.
(8) If $f(\boldsymbol{x_i}^{\boldsymbol{k}}) > f(\boldsymbol{p_{besti}})$ then $\boldsymbol{p_{besti}} = \boldsymbol{x_i}^{\boldsymbol{k}}$.
(9) If $f(\boldsymbol{x_i}^{\boldsymbol{k}}) > f(\boldsymbol{g_{best}})$ then $g_{best} = \boldsymbol{x_i}^{\boldsymbol{k}}$.
(10) If it converges, then stop.
(11) If the maximum number of iterations is not reached then go to step 2.

The stopping criterion can be a maximum number of iterations, and/or a solution with adequate *MI* value. The parameters w, c_1, and c_2 are chosen by the developer and adjust the performance and effectiveness of the CPSO method.

5.2 Hybrid PSO (HPSO) Algorithm for ImR

PSO is an effective optimization technique that has been applied to an ample range of optimization problems. Still, its performance can be improved by employing certain variants called Hybrid PSOs (HPSOs). The PSO changes can be done in one of the stages below-mentioned, or they can involve a combination of these strategies.

PSO methods can be associated with other algorithms as follows. (1) One procedure works as a pre-optimizer for the preliminary population of the next algorithm; (2) The entire population is divided into subpopulations, which evolve using PSO and other algorithms; and (3) The unique operators of an algorithm are inserted as local search improvement for the other algorithm.

PSO has been effective for ImR [28–30]. When conventional GA and PSO find problems to determine the global optimum, then HPSO approaches relying on two GA concepts: subpopulation and crossover. They are incorporated into the CPSO method to improve the accuracy of that conventional GA and CPSO since these traditional procedures cannot find the global optimum when there is a huge number of parameters to be estimated.

The particles are split into $m = 1, \ldots, M$ subpopulations, where each one has its personal best optimum $\boldsymbol{g}_{\boldsymbol{sub\text{-}best\text{-}m}}$. The PSO process is applied for each subpopulation. If $\boldsymbol{g}_{\boldsymbol{sub\text{-}best\text{-}m}}$ is better than $\boldsymbol{g}_{\boldsymbol{best}}$, then $\boldsymbol{g}_{\boldsymbol{best}}$, is replaced by the $\boldsymbol{g}_{\boldsymbol{sub\text{-}best\text{-}m}}$ where m is the subpopulation number.

The $\boldsymbol{g}_{\boldsymbol{sub\text{-}best\text{-}i}}$ are organized in ascending order of fitness function values. The top two $\boldsymbol{g}_{\boldsymbol{sub\text{-}best\text{-}i}}$ are taken as parents ($\boldsymbol{x_i}$, and $\boldsymbol{x_j}$) for a crossover with i and j as their corresponding subpopulation number. The offspring are produced for each by arithmetic crossover, using the relationships

$$\boldsymbol{x}_i' = r\boldsymbol{x}_i + (1 - r)\boldsymbol{x}_j, \quad \text{and} \tag{6}$$

$$\boldsymbol{x}_j' = r\boldsymbol{x}_j + (1 - r)\boldsymbol{x}_i, \tag{7}$$

with velocities

$$\boldsymbol{v}_i' = \boldsymbol{v}_i V, \tag{8}$$

$$\boldsymbol{v}_j' = \boldsymbol{v}_j V, \quad \text{and} \tag{9}$$

$$V = \left(\boldsymbol{v}_i + \boldsymbol{v}_j\right) / \left|\left|\boldsymbol{v}_i + \boldsymbol{v}_j\right|\right|, \tag{10}$$

where r is uniformly distributed between *0* and *1*. The offspring replaces the worst particle in the same subpopulation.

5.2.1 HPSO-Based ImR Algorithm

(1) Initialize the swarm randomly with initial values for variables $\boldsymbol{x_i^0}$, $\boldsymbol{v_i^0}$, $\boldsymbol{p_{besti}}$, $\boldsymbol{g_{best}}$
(2) For each particle i, repeat until the whole population has been analyzed:
(3) Set initial values for variables $\boldsymbol{x_i^0}$, $\boldsymbol{v_i^0}$, $\boldsymbol{p_{besti}}$, $\boldsymbol{g_{best}}$
(4) Generate M subpopulations
(5) Perform crossover
(6) If $f(\boldsymbol{x_i^k}) > f(\boldsymbol{p_{besti}})$ then $\boldsymbol{p_{besti}} = \boldsymbol{x_i^k}$
(7) Compute $\boldsymbol{g_{sub\text{-}best\text{-}m}}$ for each subpopulation m
(8) If $f(\boldsymbol{x_i^k}) > f(\boldsymbol{g_{best}})$ then $\boldsymbol{g_{best}} = \boldsymbol{x_i^k}$
(9) If it converges, then stop.
(10) If the maximum number of iterations is not reached, then go to step 2.
(11) Stop.

6 Discussion

Image recording and analysis is widely used in the fields of safety, medicine, astronomy, geology, and mechanics, among others. They are everyday image processing tasks, necessitating image comparisons grounded on similarity metrics, and limited to the specific region of each image.

Likewise, ISMs can perform a quantitative assessment of the similarity among image regions or even two images. These techniques are generally employed as a basis for ImR methodologies because they afford knowledge to indicate when the ImR course goes in the right direction with excellent outcomes.

In medical image and computer vision realms, a large number of Image Similarity Metrics exist. There is no right image similarity but a set of metrics appropriate for particular applications. However, one should bear in mind that metrics are undoubtedly the most critical constituent of an ImR problem, where they define what the process objective is, measuring how well the target resembles the reference object after the transform has been applied to it. Furthermore, just as some metrics have a rather huge capture region relating in general large capture areas are associated with low precision. Other metrics can deliver high precision ImR. Nevertheless, in general, initialization must be close to the optimal value.

Regrettably, there are no flawless rules about similarity metric selection, although in some cases it could be an advantage to employ a particular metric to achieve an initial approximation of the looked-for transformation, and, then, shift to another metric with greater sensitivity to accomplish better precision.

The experimental performance of the probed algorithms relied on optical and infrared imageries from these databases (see Figs. 6, 7, 8 and 9): MX-15 dataset [31], RGB-NIR scene—IVRL [32], and the OSU Color-Thermal [33]. Tables 1, 2, and 3 display MIR success rates for image pairs 1, 2 and 3.

- **MIR challenges**:
  - (A) Initial conditions impact the PSO performance. Poor initialization leaves the algorithm examining an objectionable area, and tarnish the optimal solution. The performance of CPSO is sensitive to the initialization of the swarms.
  - (B) The random weights governing the CPSO parameters may provoke an explosion as the particles' velocities, in addition to positional coordinates escalating towards infinity. This caveat has been traditionally restricted by a superior limit on the particle velocity, or to the step-size. Despite all the protective measures, the implementation of a good constriction factor can inhibit the explosion and improve convergence to local optima.

(a) (b)

Fig. 7 Image pairs corresponding to visible (left) and infrared (right) images. Frame 1

(a) (b)

Fig. 8 Image pairs corresponding to visible (left) and infrared (right) images. Frame 2

(a) (b)

Fig. 9 Image pairs corresponding to visible (left) and infrared (right) images. Frame 3

(C) The inertia weight (w) balances the exploration–exploitation trade-off. A big value of w upsurges the exploration, and a minor value strengthens the exploitation.
(D) The mutation operator augments the performance of PSO and permits evading the local minima. Different PSO alternatives utilizing the mutation of the global best particle together with the mutation of the local best particle were suggested to prevent the PSO from stagnation in local minima.
(E) Different data sources entail different representations, dimensionalities and formats. Whereas cameras provide data over a lattice (pixels), LIDAR generates a set of sparse as well as non-uniformly spaced achievements (multivariate real values). Radar has complex values and conversion to ordinarily utilized formats for jointly ImR processing may not be straightforward.
(F) The ImR must consider the sensors' characteristics (e.g. different resolutions or geometries). E.g., the combination of a Synthetic Aperture Radar (SAR) with an optical image must consider the SAR image locations of the objects' contributions are contingent to their distance to the sensor while the echoes their position on the ground. The SAR image can also demonstrate patterns without correspondences in the optical imagery, which may deliver worthless results. The joint use of the two modalities can only take place with the best fitting model, which encapsulates the acquisition underlying principles and knowledge of the scene.
(G) MIR is not always better than unimodal ImR because irrelevant data can spoil the analysis. Since using different modalities increases the complexity and the computational load, their use should be done when there is an actual need. To address this last aspect, a priori knowledge on the application and the characteristics of the imaging modalities should be built-in before hand.

Intricate optimization problems involving multi-dimensional spaces attain fast results in a broad assortment of applications, but it falls straightforwardly into a local optimum even in high-dimensional spaces, although with slow convergence. The last caveat is the Swarm Stagnation (SS) phenomenon.

7 Future Trends

This section contemplates unsolved issues and hindrances that may compromise the solutions of aerial image registration problems when using methods of the PSO family.

The dimensionality curse poses a substantial difficulty for any optimization handling high dimensional applications [as it is the case with computer vision and RS). This problem consists of a slowdown due to the massive size of the domain where the optimization has to work (high-dimensional hyperspace)]. Although metaheuristics such as the PSO are simple and effective multi-agent approaches, any HPSO can still face hindrances [34–37]. The dimensionality curse can impact and impair significantly color image processing that requires the application of metaheuristics to each color channel and in each pixel of the imagery [38–41]. Moreover, visualization becomes demanding or unmanageable when the dataset belongs to high dimensional hyperspaces. Therefore, there should be a dimensionality reduction from the highest dimensional space to a lower-dimensional one. Dimensionality reduction facilitates feature selection is related to acquiring the essential relevant features to describe the model.

Principal Component Analysis (PCA), in addition to computational intelligence, can be merged advantageously [42, 43]. PCA is one of the most prevalent multivariate analysis procedures to lessen the dimensionality impact of the variable space by expressing the data with a few orthogonal uncorrelated variables capturing most of the signal variability. The PCA spectral decomposition of a correlation coefficient or covariance matrix upgrades PSO variants.

Sequential Forward Selection (SFS), as well as Sequential Backward Selection (SBS) [44–46], are commonly utilized dimensionality reduction procedures relying on greedy hill-climbing to search for the optimal attribute subset, but with different starting points. Explicitly, SFS uses an empty set in the initialization step, whereas SBS begins with a set of full attributes. Nonetheless, both SFS and SBS may easily get stuck into local optima and they are expensive when the number of attributes grows. Later, the Sequential Backward Floating Selection (SBFS) and Sequential Forward Floating Selection (SFFS) appeared to automatically get the sought after values with dynamic control, instead of depending on fixed values. Based on the best first algorithm and SFFS. A Linear Forward Selection (LFS) can limit the number of attributes in each step to improve the SFS efficiency with an excellent classification performance akin to the wrapper fitness function using local search as a filter criterion. This memetic method achieved for dimensionality reduction proved to have superior performance when compared to the ones using GAs alone.

The PSO paradigm has gotten many exciting innovations and successes. The task of detecting a global amid many local optima, the aleatory nature of the search space, and the intractability of using traditional mathematical abstractions on a more comprehensive range of objective functions coupled with little or no prior guarantees about the existence of any optima challenge the search process. There are many success stories involving PSO-based algorithms with very diverse objective functions: tractable or intractable, continuous or discontinuous, even for those cases where initialization renders solution quality. Some challenging issues related to variants of the PSO are listed below:

(i) Parameter sensitivity is a consequence of the quality of the metaheuristics solution. PSO methods are sensitive to parametric developments: the same parameter selection strategy does not work for all problems.
(ii) Convergence to local optima may require the betterment of the simple PSO to consider the information on image modalities within the objective function. Often PSO varieties fall victim to local optima in the exploration space for adequately sophisticated objective functions.
(iii) Multi-objective optimization performance for high-dimensional problems poses even more questions. Although niching techniques result in satisfactory solutions for multimodal functions (in both static and dynamic environments), the solution quality may degrade abruptly as the problem dimensionality rises.

However, the PSO still requires much investigation to improve the tracking performance and other key features that would improve such algorithms. Likewise, many PSO variants address multi-objective optimization and multi-swam PSO [47, 48]. Some future research trends may address methods and changes such as multi-swarm PSO, sub-swarm scheme, the formulation of new fitness in addition to measuring functions, introduction to new search space tactics and partitioning. Some dimensionality reduction procedures can be incorporated into PSO to address its parameters' sensitivity.

The exploration of the search space is paramount to find the fittest current solution(s). In the limit situation, if the selection excludes all tentative search points, then the metaheuristic behavior will be equivalent to no exploration at all (like in the hill-climbing method). The selection effects on exploration are imperative and this deficit can be addressed by recognizing the occurrence of a failed exploration. Hence, the selection can prevent the metaheuristic from being trapped in a less favorable search space part [49–52].

Some works try to combine PSO and deep learning to obtain the internal parameters of de deep network processing layers in a simplified and fast fashion [53–58].

The use of Database as a Service (DBaaS) within a Cloud Computing (CC) permits to store structured data and manage them. As a side effect, one gets the functionality of a database similar to that found in management systems and relational databases (RDBMSes) such as SQL Server, MySQL, and Oracle. Still, CC can function together with Onboard Processing for troubleshooting [59, 60].

8 Conclusions

This chapter focuses on the registration of multimodal data (infrared and visual images) relying on PSO to circumvent problems related to the high computational cost of statistical optimization methods [42, 61–63] and multiresolution approaches [64, 65].

PSO optimizes a problem by iteratively ameliorating a candidate solution employing a given ISM. It unravels a problem by devising a population of candidate solutions, named particles moving in the search space in accord with unpretentious mathematical equations over the particle's location and velocity. Its local best-known locus influences the movements but it goes in the direction of the best-known search space positions that are updated to better estimates of the positions and depend on other particles. This procedure moves the swarm on the way to the best solutions. PSOs adaptively explore the solution space in a hyper-dimensional way, to augment computational efficiency.

The proposed HPSO algorithm can handle the VIImR for image sets captured under low-light conditions. The outcome can not only disclose the hidden but critical infrared objects by integrating the infrared bright features, but also show good visual quality by preserving much original visual information [64]. Moreover, experimental outcomes corroborate that the recommended HPSO algorithm is satisfactory in both qualitative and quantitative evaluations. Specifically, since the HPSO procedure performs image fusion by tallying the infrared structures to the optical image, thus our algorithm mainly suits for fusing the low-light infrared and visual image pairs. However, the low-light environment is the place where most need the supervision of the infrared imaging system. Likewise, if the infrared features were extracted with much background information, the contrast of the fusion image may be degraded. Experimentally, in most of the cases, the infrared features mined contain little background information, and contrast of the corresponding fusion image is well preserved from the visual image. In the future, the possible researches towards our framework could be to develop more general algorithms for fusing the infrared and visual image sets captured under more varied circumstances.

Ensemble optimizers, although auspicious, do not address the fundamental PSO underlying shortcomings. Theoretical concerns, e.g., the particle explosion problem, particle diversity loss, as well as stagnation to local optima, deserve more attention to realize a unified algorithmic structure with smarter self-adaptation in addition to less user-dependent customizations for pending applications [15, 66–68].

HPSOs algorithms are still motivating and promising because they offer further comprehensions regarding the behavior and potential benefits/disadvantages of numerous metaheuristics. This revision may encourage and support the development of new hybrid models that rest on existing models for new applications [69–87].

References

1. Alharthi A, Krotov V, Bowman M (2017) Addressing barriers to big data. Bus Horiz 60(3):285–292
2. Douglas M (2013) Big data raises big questions. Gov Technol 26(4):12–16
3. Johnson JE (2012) Big data + big analytics = Big opportunity. Financ Execut 28(6):50–53
4. McAfee A, Brynjolfsson E (2012) Big data: the management revolution. Harv Bus Rev 90(10):60–68
5. da Silva FD, Estrela VV, Matos LJ (2011) Hyperspectral analysis of remotely sensed images. In: Sustainable water management in the tropics and subtropics—and case studies in Brazil, vol 2. University of Kassel. ISBN 978-85-63337-21-4
6. Le Moigne J, Netanyahu NS, Eastman RD (2011) Image registration for remote sensing. Cambridge University Press
7. Rafailidis D, Manolopoulou S, Daras P (2013) A unified framework for multimodal retrieval. Patt Rec 46(12):3358–3370
8. Liang Q, Cheng X, Huang S, Chen D (2014) Opportunistic sensing in wireless sensor networks: theory and applications. IEEE Trans Comput 63(8):2002–2010
9. Xing B, Pan F, Feng X, Li W, Gao Q (2019) Autonomous landing of a micro aerial vehicle on a moving platform using a composite landmark. Int J Aerosp Eng. https://doi.org/10.1155/2019/4723869
10. Lin CL, Mimori A, Chen YW (2012) Hybrid particle swarm optimization and its application to multimodal 3D medical image registration. Comp Intell Neurosci, Hindawi
11. Senthilnath JA, Katti R, Omkar SN, Diwakari P (2010) An approach to multimodal satellite image registration using particle swarm optimization. Proc ICEAE 2009:1495–1499
12. Szeliski R (2010) Computer vision: algorithms and applications, Springer
13. Zheng Y, Fan J, Zhang J, Gao X (2017) Hierarchical learning of multi-task sparse metrics for large-scale image classification. Pattern Recogn 67:97–109. https://doi.org/10.1016/j.patcog.2017.01.029
14. Poli R (2008) Analysis of the publications on the applications of particle swarm optimisation. J Artifi Evol Appl, 1–10. https://doi.org/10.1155/2008/685175
15. Sengupta S, Basak S, Peters RA (2018) Particle swarm optimization: a survey of historical and recent developments with hybridization perspectives. Mach Learn Knowl Extract 1:157–191
16. Senthil Kumar K, Kavitha G, Subramanian R, Ramesh G (2011) Visual and thermal image fusion for UAV based target tracking. MATLAB—a ubiquitous tool for the practical engineer. InTechOpen
17. Li H, Ding W, Cao X, Liu C (2017) Image registration and fusion of visible and infrared integrated camera for medium-altitude unmanned aerial vehicle remote sensing. Rem Sens MDPI 9:441
18. Kim S, Ban Y, Lee S (2017) Tracking and classification of in-air hand gesture based on thermal guided joint filter. Sensors 17(1):166, MDPI. https://doi.org/10.3390/s17010166
19. Razmjooy N, Estrela VV, Loschi HJ (2019) A study on metaheuristic-based neural networks for image segmentation purposes. In: Memon QA, Khoja SA (eds) Data science theory, analysis and applications. Taylor and Francis
20. Razmjooy N, Estrela VV, Loschi HJ (2019) A survey of potatoes image segmentation based on machine vision. In: Applications of image processing and soft computing systems in agriculture. IGI Global, pp 1–38. https://doi.org/10.4018/978-1-5225-8027-0.ch001
21. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of the 1995 IEEE international conference on neural networks. IEEE, pp 1942–1948. https://doi.org/10.1109/icnn.1995.488968
22. Karekar A, Kulkarni A, Kshirsagar K, Vyavhare A (2015) 2D to 3D image conversion and disparity map estimation using PSO algorithms. In: Proceedings of the 2015 international conference on computing communication control and automation, pp 817–821

23. Khadhraoui T, Ktata S, Benzarti F, Amiri H (2016) Features selection based on modified PSO algorithm for 2D face recognition. In: 2016 13th international conference on computer graphics, imaging and visualization (CGiV), pp 99–104
24. Hamdaoui F, Sakly A, Mtibaa A (2015) An efficient multi level thresholding method for image segmentation based on the hybridization of modified PSO and Otsu's Method. In: Azar A, Vaidyanathan S (eds) Computational intelligence applications in modeling and control. Studies in computational intelligence, vol 575. Springer, Cham. https://doi.org/10.1007/978-3-319-110 17- 2_14https://doi.org/10.1007/978-3-319-11017-2_14
25. Imran M, Hashima R, Khalidb NEA (2013) An overview of particle swarm optimization variants. In: Proceedings of the 2012 Malaysian technology. Universities conference on engineering and technology (MUCET 2012), part 4: information and communications technology. Proc Eng 53:491–496
26. Na L, Yan J, Shu L (2017) Application of PSO algorithm with dynamic inertia weight in medical image thresholding segmentation. In: Proceedings of the 2017 IEEE 19th international conference on e-health networking, applications and services (Healthcom), pp 1–4
27. Duan, Y., Harley, R.A., Habetler, T.G. (2009). Comparison of Particle Swarm Optimization and Genetic Algorithm in the design of permanent magnet motors. In: Proceedings of the 2009 IEEE 6th international power electronics and motion control conference, pp 822–825
28. Thangaraj R, Pant M, Abraham A, Bouvry P (2011) Particle swarm optimization: hybridization perspectives and experimental illustrations. Appl Math Comput 217:5208–5226
29. de Jesus MA, Estrela VV, Saotome O, Stutz D (2018) Super-resolution via particle swarm optimization variants. In: Hemanth J, Balas V (eds) Biologically rationalized computing techniques for image processing applications. Lecture notes in computational vision and biomechanics, vol 25. Springer, Cham. https://doi.org/10.1007/978-3-319-61316-1_14
30. Li Q, Sato I, Li Q, Sato I (2007) Multimodality image registration by particle swarm optimization of mutual information. In: Huang DS, Heutte L, Loog M (eds) Advanced intelligent computing theories and applications. With aspects of artificial intelligence. ICIC 2007. Lecture notes in computer science, vol 4682. Springer, Berlin
31. Mitianoudis M, Stathaki T (2007) Pixel-based and region-based image fusion schemes using ICA bases. Infor Fusion 8(2):131–142
32. Fredembach C, Süsstrunk S (2008) Colouring the near infrared. In: Proceedings of the IS&T 16th color imaging conference, pp 176–182
33. OTCBVS Benchmark Dataset Collection. Available online: https://vcipl-okstate.org/pbvs/bench/. Accessed on 8 Dec 2019
34. Nasiri JA, Fard AM, Naghibzadeh M, Rouhani M (2009) High dimensional problem optimization using distributed multi-agent PSO. In: 2009 third UKSim European symposium on computer modeling and simulation, pp 245–250
35. Gardeux V, Chelouah R, Siarry P, Glover FW (2011) EM323: a line search based algorithm for solving high-dimensional continuous non-linear optimization problems. Soft Comput 15:2275–2285
36. Chen S, Montgomery J, Röhler AB (2014) Measuring the curse of dimensionality and its effects on particle swarm optimization and differential evolution. Appl Intel 42:514–526
37. Cheng R, Jin Y (2015) A competitive swarm optimizer for large scale optimization. IEEE Trans Cybernet 45:191–204
38. Razmjooy N, Mousavi BS, Khalilpour M, Hosseini H (2014) Automatic selection and fusion of color spaces for image thresholding. SIViP 8(4):603–614
39. Mousavi BS, Soleymani F, Razmjooy N (2013) Color image segmentation using neuro-fuzzy system in a novel optimized color space. Neural Comput Appl 23(5):1513–1520
40. Moallem P, Razmjooy N, Mousavi BS (2014) Robust potato color image segmentation using adaptive fuzzy inference system. Iran J Fuzz Syst 11(6):47–65
41. Mousavi B, Somayeh F, Soleymani F (2014) Semantic image classification by genetic algorithm using optimised fuzzy system based on Zernike moments. SIViP 8(5):831–842
42. Coelho AM, Estrela VV (2012) EM-based mixture models applied to video event detection. In: Principal component analysis—engineering applications. InTech. https://doi.org/10.5772/38129

43. Coelho AM, de Assis JT, Estrela VV (2009) Error concealment by means of clustered blockwise PCA. In: Proceedings of the 2009 IEEE picture coding symposium. https://doi.org/10.1109/PCS.2009.5167442
44. Kohavi R, John GH (1997) Wrappers for feature subset selection. Artif Intell 97:273–324
45. Xue B, Lane MC, Liu I, Zhang M (2016) Dimension reduction in classification using particle swarm optimisation and statistical variable grouping information. In: IEEE symposium series on computational intelligence (SSCI), pp 1–8
46. Gutlein M, Frank E, Hall M, Karwath A (2009) Large-scale attribute selection using wrappers. In: Proceeding of the IEEE symposium on computational intelligence and data mining (CIDM'09). IEEE, pp 332–339
47. Esmin AA, Coelho RA, Matwin S (2013) A review on particle swarm optimization algorithm and its variants to clustering high-dimensional data. Artif Intell Rev 44:23–45
48. Zheng H, Zhou Y (2013) A cooperative coevolutionary cuckoo search algorithm for optimization problem. J Appl Math, 912056:1–912056:9
49. Shirazi MZ, Pamulapati T, Mallipeddi R, Veluvolu KC (2017) Particle swarm optimization with ensemble of inertia weight strategies. In: Proceedings of the advances in swarm intelligence. ICSI 2017. Lecture notes in computer science, vol 10385. Springer
50. Engelbrecht AP (2012) Particle swarm optimization: velocity initialization. IEEE Congr Evol Comput 2012:1–8
51. Chen S, Röhler AB, Montgomery J, Hendtlass T (2019) An analysis on the effect of selection on exploration in particle swarm optimization and differential evolution. IEEE Congr Evol Comput (CEC) 2019:3037–3044
52. Lynn N, Suganthan PN (2015) Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation. Swarm Evol Comput 24:11–24
53. Hemanth DJ, Estrela VV (2017) Deep learning for image processing applications. In: Advances in parallel computing series, vol 31. IOS Press. ISBN 978-1-61499-821-1 (print), ISBN 978-1-61499-822-8 (online)
54. Junior FE, Yen GG (2019) Particle swarm optimization of deep neural networks architectures for image classification. Swarm Evol Comput 49:62–74
55. Wang B, Sun Y, Xue B, Zhang M (2018) Evolving deep convolutional neural networks by variable-length particle swarm optimization for image classification. In: Proceedings of the 2018 IEEE congress on evolutionary computation (CEC), pp 1–8
56. Sun Y, Xue B, Zhang M, Yen GG (2018) A Particle swarm optimization-based flexible convolutional autoencoder for image classification. IEEE Trans Neural Netw Learn Syst 30:2295–2309
57. Wang B, Sun Y, Xue B, Zhang M (2019) In: Proceedings of the genetic and evolutionary computation conference (GECCO'19), Prague, Czech Republic, pp 490–498. https://doi.org/10.1145/3321707.3321735
58. Shi W, Liu D, Cheng X, Li Y, Zhao Y (2019) Particle swarm optimization-based deep neural network for digital modulation recognition. IEEE Access 7:104591–104600
59. Ali W, Shafique MU, Majeed MA, Raza A (2019) Comparison between SQL and NoSQL databases and their relationship with big data analytics. Asian J Res Comp Sci 4(2):1–10. https://doi.org/10.9734/ajrcos/2019/v4i230108
60. Soundararajan G, Lupei D, Ghanbari S, Popescu AD, Chen J, Amza C (2009) Dynamic resource allocation for database servers running on virtual storage. In: Proccedings of the 7th conference on File and storage technologies (FAST 2009), February 2009 pp 71–84
61. Breder RLB, Estrela VV, de Assis JT (2009) Sub-pixel accuracy edge fitting by means of B-spline. In: 2009 IEEE international workshop on multimedia signal processing. IEEE. https://doi.org/10.1109/mmsp.2009.5293265
62. Coelho AM, Estrela VV, do Carmo FP, Fernandes SR (2012) Error concealment by means of motion refinement and regularized Bregman divergence. In: Proceedings of the intelligent data engineering and automated learning (IDEAL 2012), pp 650–657. Springer. https://doi.org/10.1007/978-3-642-32639-4_78

63. Estrela VV, Magalhães HA, Saotome O (2016) Total variation applications in computer vision. In: Handbook of research on emerging perspectives in intelligent pattern recognition, analysis, and image processing. IGI Global, pp 41–64. https://doi.org/10.4018/978-1-4666-8654-0.ch002
64. Zhang Y, Zhang L, Bai X, Zhang L (2017) Infrared and visual image fusion through infrared feature extraction and visual information preservation. Infrared Phys Technol 83:227–237. https://doi.org/10.1016/j.infrared.2017.05.007
65. Rivera LA, Estrela VV, Carvalho PCP, Velho L (2004) Oriented bounding boxes based on multiresolution contours, Journal of WSCG. In: Proceedings of the 12-th international conference in central Europe on computer graphics, visualization and computer vision'2004, WSCG 2004, University of West Bohemia, Campus Bory, Plzen- Bory, Czech Republic, February 2–6, 2004 (Short Papers) 2004:219–212.
66. Ma J, Jiang J, Zhou H, Zhao J, Guo X (2018) Guided locality preserving feature matching for remote sensing image registration. IEEE Trans Geosci Rem Sens 56:4435–4447
67. Yu G, Liu M, Liu T, Guo L (2018) Estimation of point cloud object pose using particle swarm optimization. In: Proceedings of the international conference on machine vision and applications (ICMVA 2018), pp 1–7. https://doi.org/10.1145/3220511.3220512
68. Chen Q, Sun J, Palade V (2019) Distributed contribution-based quantum-behaved particle swarm optimization with controlled diversity for large-scale global optimization problems. IEEE Access 7:150093–150104
69. Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69:46–61. ISSN 0965-9978
70. Rashedi E, Nezamabadipour H, Saryazdi S (2009) GSA: a gravitational search algorithm. Inf Sci 179(13):2232–2248. ISSN 0020-0255
71. Mirjalili S, Hashim SZM (2010) A new hybrid PSOGSA algorithm for function optimization, In: Proceedings of the 2010 international conference on computer and information application, pp 374–377
72. Xia X, Gui L, He G, Xie C, Wei B, Xing Y, Wu R, Tang Y (2017) A hybrid optimizer based on firefly algorithm and particle swarm optimization algorithm. J Comput Sci 26:488–500, ISSN 1877–7503
73. Sergeyev YD, Kvas DE, Mukhametzhanov MS (2018) On the efficiency of nature-inspired metaheuristics in expensive global optimization with limited budget. Sci Rep 8, Art. no. 453. https://doi.org/10.1038/s41598-017-18940-4
74. Kvasov DE, Mukhametzhanov MS (2018) Metaheuristic vs. deterministic global optimization algorithms: the univariate case. Appl Math Comput 318:245–259
75. Mahdavi M, Fesanghary M, Damangir E (2007) An improved harmony search algorithm for solving optimization problems. Appl Math Comput 188:1567–1579
76. Li Y, Zhan Z, Gong Y, Chen W, Zhang J, Li Y (2015) Differential evolution with an evolution path: a deep evolutionary algorithm. IEEE Trans Cybernet 45:1798–1810
77. Heidari AA, Pahlavani P (2017) An efficient modified grey wolf optimizer with Lévy flight for optimization tasks. Appl Soft Comput 60:115–134
78. Rajabioun R (2011) Cuckoo optimization algorithm. Appl Soft Comput 11:5508–5518
79. Coello CA, Cortés NC (2005) Solving multiobjective optimization problems using an artificial immune system. Genet Program Evolvable Mach 6:163–190
80. Kanakubo M, Hagiwara M (2007) Speed-up technique for association rule mining based on an artificial life algorithm. In: Proceedings of the 2007 IEEE GRC 2007, pp 318–318
81. Dhivyaprabha TT, Subashini P (2017) Performance analysis of synergistic fibroblast optimization (SFO) algorithm. In: Proceedings of the 2017 IEEE ICCTAC, pp 1–7
82. Majumder A, Behera L, Venkatesh KS (2014) Emotion recognition from geometric facial features using self-organizing map. Pattern Recogn 47:1282–1293
83. Karaboga D, Basturk B (2007) A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. J Glob Optim 39:459–471
84. Mirjalili SM, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51–67
85. Tilahun SL, Ong HC (2012) Modified firefly algorithm. J. Appl Math 467631:1–467631:12

86. Miranda V, Martins JD, Palma V (2014) Optimizing large scale problems with metaheuristics in a reduced space mapped by autoencoders—application to the wind-hydro coordination. IEEE Trans Power Syst 29:3078–3085
87. Razmjooy N, Estrela VV, Loschi HJ (2019) A survey of potatoes image segmentation based on machine vision. applications of image processing and soft computing systems in agriculture. IGI Global 2019:1–38. https://doi.org/10.4018/978-1-5225-8027-0.ch001

Using Metaheuristics in Discrete-Event Simulation

Reinaldo Padilha França, Ana Carolina Borges Monteiro, Vania V. Estrela, and Navid Razmjooy

Abstract Metaheuristics have proven to be a powerful tool for roughly solving optimization problems, applied to find answers to problems about which there is little information. In general, meta-heuristics use a combination of random choices and historical knowledge of the previous results acquired by the method to guide and search the search space in neighborhoods within the search space, avoiding premature stoppages in optimal locations. A strategy that guides or modifies a heuristic to produce solutions that surpass the quality of those commonly encountered. The discrete event simulation (DES) is a representation of a system as a sequence of operations by state transactions (entities), where these entities are discrete and may be relative to various types depending on the context of the problem that is being sought. In this way is brought to the eyes of interest, the union of discrete events simulation with metaheuristic science, whether direct or not, is successful.

Keywords Metaheuristics · Events · Entities · DES · Optimization · Simulation

1 Introduction

Modeling and simulation is a widely used tool in engineering because it allows the display of conditions of use, testing of equipment and devices, operating system

---

R. P. França (✉) · A. C. B. Monteiro
School of Electrical and Computer Engineering (FEEC), University of Campinas—UNICAMP, Av. Albert Einstein—400, Barão Geraldo, Campinas—SP, Brazil
e-mail: padilha@decom.fee.unicamp.com

A. C. B. Monteiro
e-mail: monteiro@decom.fee.unicamp.com

V. V. Estrela
Department of Telecommunications, Fluminense Federal University (UFF), Rio de Janeiro, Brazil
e-mail: vania.estrela.phd@ieee.org

N. Razmjooy
Department of Electrical Engineering, University of Tafresh, Tafresh, Iran
e-mail: navid.razmjooy@hotmail.com

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_14

evaluation, just as it is a method or which allows analyzing the behavior of an object, aa from mathematical models employed by a computer. Since the 1980s, coinciding with the evolution of microcomputers and the availability of software, several scientific works have emerged with the aim of computer simulation or behavior of various processes [1].

With advances in computer science, modern equipment and new programming and simulation languages have allowed the use of simulation techniques in various areas of human knowledge. For the simulation uses finite element methods, which consists of treating its model as a continuous body and dividing it into small connected elements, consequently creating a kind of mesh that makes up the total object, thus making it possible to emulate more complex situations, such as COP (combinatorial optimization problems). Simulation is essential as it allows you to test different possibilities without spending on prototype development [1, 2].

The simulation methodology is widely used in various market sectors, such as aeronautics through flight simulators or medicine, where patients are virtual, as well as several other areas of science, being highly efficient as it helps to see the organization. As a whole with its processes, failures, and solutions. Since with computer simulation, it is possible to explore various scenarios and solutions, identifying faults in complex processes and can correct them, reducing the total time spent on prototype development and minimizing the costs involved [3].

The information obtained through the simulation shows the probable failures, thus decreasing the need for prototypes and erroneous solutions, besides having an important role in the optimization of existing processes, since with the simulation it is possible to simulate metaheuristic algorithms which have gained. It stands out as a prominent approach in solving real-world optimization problems, capable of dealing with nontrivial objective functions, such as multi-objective, non-smooth and noisy, non-convex functions, among others, even considering complex constraints and decision variables of different natures. Metaheuristics allow decision-makers, through simulation, to get nearly optimal solutions to large and complex problems at reasonably low computation times, sometimes even “real-time”, depending on computational processing power, a few seconds [1, 3].

A simulation is a tool provided by the operational research area that allows the generation of scenarios, from which one can: guide the decision-making process, carry out systems analysis and evaluations and propose solutions for the improvement of performance. In the specific case of engineering, the adoption of the simulation technique has brought benefits such as risk reduction in decision making, prediction of results in the execution of a certain action, identification of problems even before their occurrence, analysis of sensitivity, the elimination of procedures in industrial arrangements that do not add value to production, as well as the reduction of resource costs [4].

Just as a simulation model by definition can be considered as a set of rules such as flowcharts, equations, or state machines, which have the properties of defining system modeling, taking into account its current state and how it will change in the system, future. Just as a system can be defined as a collection of structures and resources that are interacted according to logic in order to achieve one or more objectives, where the studies of these systems through simulation can take place under different forms of approach. The simulation also allows researchers to study, construct and perform valid modeling of complex systems in a straightforward and straightforward manner, studying their characteristics and problems without any assumptions made about an analytically treatable optics of the model, as this may compromise their veracity. Simulation is the process of executing a given model leading to state changes over time, which may occur discretely or continuously over time [5].

Thus, a discrete simulation relies on countable phenomena that results in changing the behavior of the modeled system only in response to specific events and generally models the changes in a system resulting from a finite number of events distributed over time, whereas for in continuous models, the advancement of the time counting in the simulation occurs continuously, which makes it possible to determine the values of the state variables at any time. With the simulation, it is possible to use different paradigms that can be integrated into a similarity, starting from the most significant, from faster (less computationally intensive) to richer (more computationally intensive) approaches [2].

Thus, a discrete simulation relies on countable phenomena that results in changing the behavior of the modeled system only in response to specific events and generally models the changes in a system resulting from a finite number of events distributed over time, whereas for In continuous models, the advancement of the time counting in the simulation occurs continuously, which makes it possible to determine the values of the state variables at any time. With the simulation, it is possible to use different paradigms that can be integrated into a similarity, starting from the most significant, from faster (less computationally intensive) to richer (more computationally intensive) approaches [2].

So, metaheuristics are more sophisticated generic heuristics, where a simpler heuristic is managed by a procedure that aims to intelligently explore the instance of the problem and its solution space. Among the heuristics, metaheuristics are considered more generic strategies that, controlled by probabilistic criteria, make it possible to escape from great locations and, therefore, explore more promising regions. In this context, approaches combining simulation with metaheuristics, called simheuristics, allow in a virtual environment to model the uncertainty of real life in a natural way, integrating the simulation, in any of its variants, with a structure-oriented by metaheuristics. Since these optimization algorithms depend on the fact that efficient metaheuristics already exist for the corresponding deterministic version of COP, that is combinatorial optimization problems. Thus, simheuristics also make it easy to introduce reliability criteria and/or risk analysis when evaluating high-quality alternative solutions for stochastic COPs. Metaheuristics can be conceived as intelligent strategies for the design of more generalist and high-performance heuristic procedures [6].

Finally, metaheuristics can be defined as solution methods that have the ability to orchestrate the interaction between top-level local strategies to create a process capable of escaping optimum sites and improvement processes by conducting a robust space search, solution. This intelligent strategy in which metaheuristics are sustained is basically to improve the solution along with the algorithm development through intensification and diversification actions [7].

With regard to the discrete event simulation (DES) technique, it consists of a modeling method for stochastic dynamic models, where the system state variables modeled in the simulation can change only in instantaneous and time-separated moments, called and considered discrete events. The process-oriented worldview with a discrete-event technique typically consists of describing how entities move through various processes, where each process may require one or more resources, leading to a certain commonly stochastic amount of time. From another perspective, event-oriented view works at a more basic and fundamental level, that is, with logic occurring in the instantaneous discrete events themselves, rather than with entities and resources. The clock simulation time of the occurrence of events can be continuous, i.e. a real number, as well as the time between successive events, but the occurrence of events is instantaneous. Strictly this excludes state variables that change continuously over time with respect to their stochastic characteristics, in general, several simulation runs are performed in order to obtain statistically more accurate results [8–10].

Therefore, this chapter aims to discuss metaheuristics in discrete-event simulation, categorizing and synthesizing the potential of technologies that involves thematic. Thus, Sect. 2 explains the optimization role in simulation. Discrete Events Simulation (DES) is explored in Sect. 3. Section 4 deals with Metaheuristics in DES. Section 5 shows some case studies. Lastly, Sect. 6 outlines future trends in technology. And finally, Sect. 7 concludes the chapter.

2 Optimization Role in Simulation

Optimization problems are often encountered in different technical fields, such as Engineering, Economics, or even day to day since such problems often aim for contradictory objectives, relating costs, deadlines, profits, quality, efficiency, and other variables. The formulation of any optimization problem involves the composition of an objective function, which lists the different variables considered, as well as the constraints imposed on each one. Optimizing a given problem consists of identifying its solution, or its values for the variables considered, that maximize or minimize the value of the objective function, depending on the nature of the problem, so that no other solution assigns a higher or lower value to the function, respectively respecting the constraints of the problem [11].

2.1 Gradient-Based Search Methods

Often used gradient estimation methods, **Gradient-based methods** are iterative methods that extensively use the gradient information of the objective function during iterations, they have features to estimate the response function gradient (Ñf) to assess the shape of the objective function and employ deterministic mathematical. They are iterative methods that extensively use the gradient information of the objective function during iterations, to solve smooth nonlinear optimization problems, since these methods use only local information (functions and their gradients at a point) in their search process, which converges only to a local minimum point for the cost function [12].

These methods use gradients of the problem functions to perform the search for the optimum point, so all of the problem functions are assumed to be smooth and at least twice continuously differentiable everywhere in the feasible design space, just as the design variables are assumed to be continuous that can have any value in their allowable ranges. Though gradient-based methods can be very efficient, the final solution tends to depend on the starting point, because if this is very far from the optimal solution, the algorithm can either reach a completely different solution for multimodal problems or simply fail in some cases, there is no guarantee that the global optimal solution can be found [12, 13].

The **Finite Difference** method consists of reformulating the continuous problem into a discrete problem using finite difference formulas taken over an appropriate mesh since it is an older method and is easy to implement. Finite differences are the crudest method of estimating the gradient, and its principle is derivatives in the partial differential equation are approximated by linear combinations of function values at the grid points [14].

Where to estimate the gradient at a specific value of x, at least n + 1 configurations of the simulation model must be run, so get the most reliable estimate of Ñf there may be a need for multiple observations for each partial derivative, further increasing the already high computational cost. It is one of the methods used to solve differential equations that are difficult or impossible to solve analytically. The concept is focused on approximating differentials, uses an approximation of the differential equation. It is, therefore, a differential approximation. The finite difference method was among the first approaches applied to the numerical solution of differential equations, is directly applied to the differential form of the governing equations. It is a very versatile modeling technique, which is comparatively easy for users to understand and implement [14, 15].

The **Likelihood Ratio (LR)** method relates the gradient of the expected value of an output variable with respect to an input variable expressed as the expected value of a function of both input parameters and simulation parameters, i.e. has ranged over simulation run length and output variable value. The derivative method of the gradient estimation method, having implementation and formulation in the system-based Monte Carlo approach which is generally used in dynamic reliability applications is first given, with the focus on speed up the simulation [16].

The idea behind the perturbation **Analysis Method (PA)** is that in a system, all partial gradients of an objective function are estimated from a single simulation run. Since if an input variable is disturbed, the sensitivity of the output variable to the parameter can be estimated by tracing its pattern of propagation. The differential fact is that all derivatives can be derived from a single simulation run, which represents a significant advantage in terms of computational efficiency, however, the estimators derived using PA are often biased and inconsistent [17].

DES is a tool used for experimentation in discrete-event systems with perturbation analysis, in which the technique expedites the data collection process by performing various experiments concurrently and communication systems can be described as discrete-event systems [18, 19].

Frequency Domain Method (FDM) is assigned a frequency range for each channel, where this signal must be frequency shifted to its position before channel multiplexing. Such channel shift to a specific frequency spectrum position is by of a modulation process which must be done in such a way that the modulated signal does not interfere with the other channels to be multiplexed. Thus, FDM is basically a frequency separation of the z channels to be multiplexed, resulting in a time overlap of signals [20].

2.2 Stochastic Optimization

In a problem where there are several objectives considered simultaneously, some of which may conflict with each other, the decision-maker will choose solutions that provide a balance between these conflicts and meet the interests of the company. In many practical problems in the industrial and business context, the decision-maker must optimize these conflicting objectives, taking into account the uncertainty in the coefficients of objective functions and constraints. In general, the coefficients of objective functions can be stochastic. In stochastic models, there are model parameters being probability densities [21].

In summary, **stochastic optimization** is the problem of finding a local optimum for an objective function whose values are not known analytically but can be estimated or measured. One of the advantages of using is the ability to extract a set of relevant information related to the problem in question, and thus enable the analysis of different scenarios. Classical stochastic optimization algorithms are iterative schemes based on gradient estimation [22].

2.3 Response Surface Methodology (RSM)

Response Surface Methodology (RSM) is a statistical technique used for modeling and analyzing problems in which the response variable is influenced by several factors, the purpose of which is to optimize this response. It is often applied for the purpose of streamlining the process (finding a combination of levels of factors that drive the response to the best possible value, usually in terms of expected value) [23].

The method aims to find the combination of process factors (X1, X2, …, Xk) leading to the best response, which is the optimal (maximum or minimum), as well as to know the behavior of the response (or expected response value) when changes in process factor levels occur [24].

2.4 *Heuristic Methods*

Heuristic Methods are feasible solutions derived from prescriptive analysis but not guaranteed to be the exact optimum. Many of these techniques balance exploration with exploitation thereby resulting in efficient global search strategies [25].

Genetic Algorithms (GAs) differ from traditional search and optimization methods since GAs work with a parameter set coding and not with the parameters themselves; work with a population rather than a single point; they use probabilistic and non-deterministic transition rules and in the same way, they use non-derivative cost or reward information or other ancillary knowledge [26].

Genetic algorithms are very effective at finding optimal or approximately optimal solutions for a wide range of problems, as they do not impose many of the limitations found in traditional search methods. They represent a class of optimization algorithms that employ a probabilistic search engine based on the process of biological evolution, combining aspects of the mechanics of genetics and natural selection of individuals. In addition to being a very elegant generate-and-test strategy, they are able to identify and exploit environmental factors and converge on optimal or approximately optimal solutions at global levels [26, 27].

Evolution strategies (ES) are a class of evolutionary algorithms primarily used to solve parameter optimization problems. They initially dealt with optimization problems in fluid mechanics, and then dealt with function optimization problems more generally, focusing on the case of real functions. Historically, the first evolutionary strategy algorithms operated with a single individual in the population, subject to mutation and selection. Nowadays, an important idea introduced in the most recent algorithms is the online adaptation (self-adaptation) of the strategy parameters during the evolutionary process, by introducing them in the genetic representation of individuals [28].

Simulated annealing is a technique consisting roughly of ideas similar to those of genetic algorithms, based on statistical mechanics and material annealing. A possible solution is initially generated, and successive modifications are made, each of them proportional to a parameter called temperature, which is initially large and gradually reduced as converges to the desired solution. It is based on Statistical Mechanics, in which only the most likely behavior of the system can be observed, and more specifically in the search for the state of these systems at low temperatures. By adopting this technique for system optimization in general, the "energy" of the system is determined by a "cost function", and the "temperature" becomes a control parameter in the system same unit of the cost function [29].

The heuristic goal known as **Tabu Search (TS)** is an adaptive auxiliary procedure that guides a local search algorithm into a continuous exploration within a search space. It is a mathematical optimization method, belonging to the class-based trajectory techniques, it enhances the performance of a local search method using memory structures that describe the solutions visited, once the potential solution has been determined, is marked as "tabu", so that the algorithm doesn't repeatedly visit this possibility. Unlike what may happen with other procedures, Tabu Search (TS) is not confused by the absence of enhancing neighbors, this means that the method avoids returning to a previously visited optimal location in order to overcome local optimality and achieve an optimal result, next to the great global [30].

The **Nelder-Mead method** or (slope simplex method, or amoeba method) is an applied numerical method used to find the maximum or minimum of an objective function in a multidimensional space. Problems for which derivatives cannot be known are applied to nonlinear optimization. However, the technique is a heuristic research method that can converge to non-stationary points on problems that can be solved by alternative methods. The method uses the concept of a simplex, which is a special polyepitope of n + 1 vertices in n dimensions. Examples of simplicity include a line segment of a line, a triangle on an airplane, a tetrahedron in three-dimensional space, and so on. It is a technique for giving numerical solutions to linear programming problems, and it is a numerical method for optimizing multidimensional free problems belonging to the more general class of search algorithms. In either case, the method uses the concept of a simplex, which is a polytope of N + 1 vertices in N dimensions, i.e., a line segment on a line, a triangle on a plane, a tetrahedron in a space of three dimensions and so on [31].

2.5 A-Teams

A-team as a process that is both fast and robust, consisting of GAs and conventional algorithms for solving sets of nonlinear algebraic equations, which result in considerable savings in the amount of computational effort (number of function evaluations) with respect to the need for computational resources necessary for finding solutions [11].

2.6 Statistical Methods

Statistical Methods has its importance related to sampling which is effectively used to achieve relevant speed ups in simulations involving rare events, such as failure in a reliable computer system. Sampling simulates the system under a different probability measure, with different underlying probability distributions, so as to increase the probability of typical sample paths involving the rare event of interest [32].

Ranking and Selection Methods are generally employed for practical problems related to finding the best location for a new facility to minimize cost, or even finding the best combination of parts manufactured on various machines to maximize productivity. Which in these optimization problems, is available some knowledge of the relationship among the alternatives, as long as these methods have the ability to treat the optimization problem as a multi-criteria decision problem [33].

There are also the **Multiple Comparisons with The Best (MCB)** method, which if in a given problem fits the best of a finite number of system designs, this method is a great alternative to ranking and selection. Which performs procedures inference about the relative performance of all tested alternatives, where such inference is critical if the performance measure of interest is not the sole criterion for decision making, that is, expected throughput of a manufacturing system may be the performance measure of interest however the cost of maintaining the system is also important [34].

3 Discrete Events Simulation (DES)

Simulation is not an optimization technique, but it is through it that performance measurements of a modeled system are estimated. Discrete Models relate system states whose change occurs only at the moment an event occurs, for every other time, nothing changes in the system. The timescale is real, that is, events occur and are handled at the same timescale as the real system. For this, we use the concepts of queues, which a generic definition of Events, is related in which all Discrete Event Simulation describes, directly or indirectly, queuing situations, where clients arrive, waiting in line if necessary and then receive service before leaving the system. In general, there are usually only two events that control simulation, arrivals, and calls [35].

The difference between continuous-discrete models lies in the fact that in continuous models the values of variables change gradually over time and are usually represented by differential equations, such as growing a plant, inflating a tire, car or varying the level of a fuel tank. Discrete Events, on the other, they evolve as system states change and are easily identified, such as a stationary train stop or the mounting of a chair base [8–10].

In a discrete event system, one or more phenomena of interest change their value, or state, at discrete points rather than continuously in time. The occurrence of these events changes the state of the system at each moment. Thus, it is assumed that there are no system changes between one event and another. Even if there are fixed increments of time advance, which is not very common, system evolution does not occur continuously over time. In it, time is divided into small slices and the state of the system is updated according to the activities that occur in each slice of time. Since not every slice of time has activity occurring, this simulation is faster than continuous simulation. Objects in a discrete event system are called entities. In short, the simulation logic consists of checking the next scheduled event and updating the simulation clock and other system state variables. Execution termination can be controlled through a simulation time limit or a limit on the number of events that occur, and at the end of the simulation, reports will be issued with the simulation results [36].

The basic elements of a discrete event simulation are the state of the event is represented by variables that represent the properties of the system to be studied. The simulation clock keeping track of time evolution with respect to the chosen unit of time. And the event list is called the pending event list at the start of the simulation, which as the simulation clock progresses, the events are performed, and the system state is updated. Pending events are organized in a priority list, sorted by event duration. Regardless of how they are sorted, events are removed from the list in chronological order of the simulation [18].

The objective of the discrete event simulation is to reproduce the activities of the entities that make up the system and, from there, to know the behavior and performance of the system. For this, we need to define the state of the system and the activities that drive the system from state to state. In discrete event simulation, the state change is determined by the occurrence of an event at a deterministic or stochastic time.

The main Discrete Event Simulation models are event-oriented models, which are defined by the state changes that can occur in each event; and process-oriented models, being defined by the process by which system entities are conducted, in this specific case, the process consists of the sequence of events. The process approach is one of the most commonly used since a modeled system uses event or activity-based approaches, where the process of each entity class is considered, dividing them into fundamental parts, either independent events or activity connections. Entities are system elements whose behavior is explicitly traced. They can be permanent or temporary. Thus, process-based simulation evaluates every process, or sequence of operations, that the entity needs to go through. Thus, each entity class has its own process. Thus, a model is formed by a set of processes [8–10].

4 Metaheuristics in DES

Metaheuristics throughout have been used for continuous optimization because they do not need specific requirements for robust and reliable performance, effectively exploit research spaces where specialized knowledge is scarce or difficult to code and where traditional optimization techniques have failed [37]. A differentiable or continuous objective function, yet considering global search capability, easy implementation, and implicit parallelism. This method is non-deterministic and approximate algorithms, they are not problem-specific, allowing for an abstract level of description, even though they may make use of domain-specific knowledge to enhance the search process, but do not always guarantee that they can find the ideal solution., but a good approach in a reasonable time [38–40]. It has the main objective in the optimization/learning procedures related to the exchange between intensification and diversification. The properties of intensification (exploration) imply the concentration of research in the local region where good solutions were found while diversification (exploitation) implies the generation of various solutions to explore the research space in hidden places [41, 42].

Metaheuristics may be present in Trajectory Methods, which are tabu search, simulated annealing, iterated local search, or variable neighborhood search. Consisting of a search process describing a trajectory in the search space can be seen as the evolution (discrete) in the time of a discrete dynamic system. Just as metaheuristics can be present in Population-based methods, such as evolutionary algorithms, such as genetic algorithms, evolutionary strategies, and evolutionary programming, and swarm intelligence techniques, such as particle optimization, which deal in all iterations of the population algorithm, with a population of solutions. In which, the research process can be seen as the evolution (discrete) in time of a set of points in the solution space [42].

Evolutionary algorithms are based on a computational paradigm that replicates mechanisms inspired by those of biological evolution, such as mutation, recombination, reproduction, and selection, with a focus on solving optimization problems. Unlike swarm intelligence methods that mimic the collective behavior of decentralized, self-organized artificial systems, where global research is an emerging behavior of a population of "agents" with uniquely programmed execution of simpler tasks. With respect to differential evolution is a newer algorithm for continuous optimization, which inherits characteristics from evolutionary algorithms and swarms intelligence methods [43].

As metaheuristics are seen in memetic algorithms such as dispersion search, which are hybrid global/local search methods, in which a local improvement procedure is incorporated into a population-based algorithm. Since some of these algorithms originated independently of existing ones and constituted a specific niche in this class. The idea is to mimic the effect of learning and social interaction over an individual's life by some kind of (local) improvement mechanism applied to the solutions found by the usual global search operators. Such a generic definition allows this type of algorithm to include a virtually infinite variety of possible hybridizations of existing methods [44].

The empirical evidence of the effectiveness of metaheuristics is clear, however, this technique is not perfectly defined as a unified and formal paradigm, as long as it does not guarantee the result, however, it is found in the literature relevant mathematical evidence related to the convergence properties of these algorithms demonstrating its power computational. In terms of asymptotic convergence, it is already well known the convergence properties of evolutionary algorithms in which the algorithms are generally limited to elitist mutation and substitution operators, and there is convergence for evolutionary strategies and swarm intelligence algorithms. Generally, the combination of the main components of such algorithms (intensification/exploitation and diversification/exploitation) ensures the overall optimization possible [45].

4.1 Statistical Methods

Optimization problems are based on three main points: the problem coding, the objective function to be minimized or minimized, and the associated solution space. Genetic algorithms use concepts of mutation and selection. It is a search technique used in computer science to find approximate solutions to optimization and search problems. It is a particular class of evolutionary algorithms that use techniques inspired by evolutionary biology such as heredity, mutation, natural selection, and crossing over [11].

Genetic algorithms are a family of evolutionary-inspired computational models that incorporate a potential solution to a specific problem in a chromosome-like structure and apply selection and cross-over operators to these structures to preserve critical information about each other, to the solution of the problem. They are usually viewed as function optimizers, although the amount of problems to which they apply is quite broad. In general, the **Genetic Algorithm** works by creating a population of strings and each of these strings is called chromosomes. Each of these chromosome strings is basically a vector of a point in the search space. New chromosomes are created by using selection, mutation and crossover functions. The selection process is guided by evaluating the fitness (or objective function) of each chromosome and selecting the chromosomes according to their fitness values (using methods such as mapping onto the Roulette Wheel). Additional chromosomes are then generated using crossover and mutation functions [26, 27].

One of the advantages of a genetic algorithm is the simplification they allow in formulating and solving optimization problems, they are indicated for solving complex optimization problems, NP-Complete. Also, in many cases where other optimization strategies fail to find a solution, this kind of algorithm converges. The cross over and mutation functions ensure that a diversity of solutions is maintained [27].

Simulated Annealing uses a probabilistic method, although it was originally meant for optimizing deterministic functions, the framework has been extended to the case of stochastic simulations. Having logic derived from the annealing process in which the material is slowly cooled so that, while its structure freezes, it reaches a minimum energy state. Being of ease of implementing procedure, it remains a popular technique used by several commercial simulation optimization packages, since this type of algorithm has exponential form, because the probability of acceptance of a neighborhood point is higher, and it searches for a large number of neighborhood points in the beginning, but a lower number of points as temperature is reduced. With respect to your implementation procedures, this requires choosing parameters such as the initial and final temperatures, the rate of cooling, and the number of function evaluations at each temperature [29].

Tabu Search uses special memory structures, consisting of a modified neighborhood search procedure employing adaptive memory to keep track of relevant solution history, together with strategies for exploiting this memory; with respect to short-term and long-term, during the search process allowing the method to go beyond local optimality to explore promising regions of the search space [30].

Scatter Search and its generalized form, path relinking, which differ from other evolutionary approaches as Genetic Algorithms with respect to the use of strategic designs and search path construction from a population of solutions as compared to randomization, respective mutation and crossover in Genetic Algorithms. Since Scatter Search also uses adaptive memory in storing the best solutions, it is similar to the Tabu search [46].

5 Case Studies

It was made a review based on research on metaheuristic papers in combination with discrete-event simulation (DES) where some examples were chosen showing the applicability of techniques, with emphasis on publications and indexing in renowned databases.

Solutions to increase production delivery were studied in 2010, but industrial systems are often disturbed by random events, where there is a lack of information about the consequences of random events on production time and the quality of manufactured products. Whereas surface treatment lines are time window constrained robotic cells and therefore random events can have important consequences in production management where products can be damaged by constraint violations and rejected by the quality control department. Since one of the most difficult operational issues to address on surface treatment lines is proper coordination of material handling resource management and job entry planning. In this scenario we studied the search schedules to avoid consequences of random events, leading to the focus of the Stochastic Hoist Scheduling Problem (SHSP), which is the HSP with random events that imply variations in transport times. Where a method based on the stochastic metaheuristics used to determine the production schedules for which the consequences of random events are low has been proposed. Taking decision-event event simulation modeling into account Stochastic minimization of stochastic metaheuristics focusing on task scheduling and metaheuristics by calculating task entry dates consequences of the discrete event simulation model with random event decision rules [47].

In 2012, the use of a swarm algorithm derived from the Ant Colony Optimization metaheuristic was studied. The algorithm works with a vehicle model and always obtained viable solutions, focusing on solving path planning problems for autonomous vehicles (ASV), aiming to obtain ideal trajectories for maneuvers of autonomous surface vessels. It is considered the whole process of building the solution as a hybrid system, since the ASV evolves in its continuous spatial state, governed by a set of discrete events, with the crossing of the cell boundaries. For each event, there is a change in ASV speed and heading setpoints, where only a finite set of discrete setpoints are available, and they can only be changed in discrete events. Depending on usage and sufficient time, it can also achieve very close trajectories, and as appropriate modifications, the algorithm has characteristics for application in solving other combinatorial optimization problems with an unrestricted number of viable solutions [48].

In 2013, a scheduling problem of a real-world production process in the metallurgical industry was addressed, which can be described as a stochastic offline store flow problem with limited buffers. In this way, a neighborhood-based variable solution approach was developed, in which several scenarios were used to evaluate the objective, and so the solution approach is adapted using a detailed simulation of discrete events to evaluate the production plans, and applying a metaheuristic solution approach to a stochastic problem, even if the objective value assessment is time consuming, where production plans have been statistically improved by 3–10% [49].

In 2014 was studied whereas artificial intelligence methodologies, such as the core of discrete control and decision support systems, have been extensively applied in the industrial production sector, resulting in tools producing excellent results, yet the difficult nature of many discrete controls or decision-making problems. Manufacturing decision making may require inaccessible computational resources, limited by the available time needed to arrive at a solution, thus applying a metaheuristic of genetic algorithms applied in the search for the best solutions in the solution space [50].

In 2015 was studied many combinatorial optimization problems encountered in real-world logistics, transportation, production, healthcare, finance, telecommunications, and computing applications are NP in nature, where metaheuristics have been noted to benefit from different randomization and parallelization paradigms, but often assume that the problem inputs, the underlying objective function, and the set of optimization constraints are deterministic, wherewith this focus a general methodology has been described that allows you to extend metaheuristics through simulation to solve stochastic combinatorial optimization problems, which It was also seen the use of simulation combining continuous-time, discrete-event, and state-transition systems, which mimics the production system since they use the mathematical optimization model to replicate the decision problem [51].

In 2016 was studied domestic service sectors where vehicles are used to visit customers with the focus of reducing the number of vehicles needed, however, there is complexity in coordinating the arrival times of employees and vehicles flexible discrete event-oriented heuristics to handle dynamic routing and scheduling scenarios using combined travel and hiking sharing [52].

In 2017 was studied complex workshop stochastic scheduling problems can be addressed by simulation-based optimization by combining meta-heuristic optimization capabilities with the system representation of simulation models and exploring the potential of coupling optimization and simulation techniques in different scenarios, workshop scheduling, being developed a genetic algorithm combined with discrete event simulation [53].

In 2018 was presented the development of a project optimization set focusing on a hybrid metaheuristic optimization set to solve multi-objective simulation optimization problems using a third-party library to solve discrete-event stochastic simulation optimization problems integrated with Siemens Tecnomatix Plant Simulation, a simulation software package developed by Siemens [54].

6 Future Trends

Trends in Metaheuristics, Algorithms and Optimization Approaches such as **Particle Swarm Optimization (PSO)** allowing solving a problem through the concept of particles represented by insects, birds, and other organisms. PSO emulates this behavior as it initializes a population with a number of particles that move in a space of solutions for any problem; **Genetic Algorithm (GA)** continues to be a growing trend since it has a wide range of applications aimed at improving the overall search capability of the process; Immune algorithms are inspired by nature as they mimic the behavior of genes and antibodies while defending the human body from viruses and infections or bacteria; **Differential Evolution (DE)** algorithm, which is a heuristic method, which does not use derivatives, but is a stochastic search algorithm, originated from natural selection mechanisms, and effective for solving discontinuous objective function optimization problems, since it does not require derivative information, where it aims to solve continuous optimization problems; **Ant Colony Optimization (ACO)** is a heuristic algorithm commonly used to troubleshoot graphs; since all are increasingly used optimization method, which is considered the latest advances and technologies in metaheuristics computing. As well as developing **hybrid metaheuristic algorithms** of these technologies is necessary to develop new approaches in order to make these techniques faster with regard to the optimization algorithms solution with respect to the optimization algorithms solution [55–57].

7 Conclusions

As can be seen, metaheuristic techniques have varied applications in different areas of science, having positive characteristics as to the implementation and the great efficiency of these techniques in relation to conventional heuristics with great potential of application has been in the problems related to genetic algorithms.

As can be seen throughout the text, the union of discrete events with metaheuristic, whether direct or not, is successful. Where using the metaheuristic together with its application fundamentals with regard to developing viable solutions of good quality, based on the characteristics of a determined and specific problem with reduced complexity in relation to that of traditional exact modeling, generating a unique solution being applied to that particular problem, is oriented to the overall optimization of a given problem.

Although all the procedures described by metaheuristics can be seen as algorithmic search strategies, besides the fact that these methods have been widely studied in the last decades, which deepens the knowledge about the problem-solving process complexes; in fact, its importance is due to other characteristics as regards the description of methods that provide ideas that can be applied to optimization problems for which efficient (sometimes not even heuristic) specific algorithms are known.

In other words, a metaheuristic can be understood as a strategy that tries to efficiently explore the space of viable solutions to this problem, where one has specific knowledge of the problem and can be used as a heuristic to assist in the process. In short, metaheuristics are high-level mechanisms that exploit a particular type of strategy for a specific solution.

References

1. Sokolowski JA, Banks CM (2010) Modeling and simulation fundamentals: theoretical underpinnings and practical domains. Wiley
2. Zeigler BP, Muzy A, Kofman E (2018) Theory of modeling and simulation: discrete event & iterative system computational foundations. Academic Press
3. Edmonds B, Hales D (2005) Computational simulation as theoretical experiment. J Math Sociol 29(3):209–232
4. Azar AT, Vaidyanathan S (eds) (2015) Computational intelligence applications in modeling and control. Springer International Publishing
5. Dubois G (2018) Modeling and simulation: challenges and best practices for industry. CRC Press
6. Du KL, Swamy MNS (2016) Search and optimization by metaheuristics. Techniques and algorithms inspired by nature. Birkhauser, Basel
7. Sörensen K, Sevaux M, Glover F (2018) A history of metaheuristics. In: Handbook of heuristics, pp 1–18
8. França RP, Iano Y, Monteiro ACB, Arthur R (2020) Improvement for channels with multipath fading (MF) through the methodology CBEDE. In: Fundamental and supportive technologies for 5G mobile networks. IGI Global, pp 25–43
9. França RP, Iano Y, Monteiro ACB, Arthur R (2020) A proposal of improvement for transmission channels in cloud environments using the CBEDE methodology. In: Modern principles, practices, and algorithms for cloud security. IGI Global, pp 184–202
10. França RP, Iano Y, Monteiro ACB, Arthur R (2020) Improvement of the transmission of information for ICT techniques through CBEDE methodology. In: Utilizing educational data mining techniques for improved learning: emerging research and opportunities. IGI Global, pp 13–34
11. Gosavi A (2015) Simulation-based optimization. Springer, Berlin
12. Venter G (2010) Review of optimization techniques. In: Encyclopedia of aerospace engineering
13. Andradóttir S (1998) A review of simulation optimization techniques. In: 1998 winter simulation conference. Proceedings (Cat. No. 98CH36274), vol 1. IEEE, pp 151–158
14. Hamilton B (2016) Finite difference and finite volume methods for wave-based modelling of room acoustics
15. Zienkiewicz OC, Morgan K, Morgan K (2006) Finite elements and approximation. Courier Corporation

16. Fu MC (2015) Stochastic gradient estimation. In: Handbook of simulation optimization. Springer, New York, pp 105–147
17. Wardi Y, Cassandras CG, Cao XR (2018) Perturbation analysis: a framework for data-driven control and optimization of discrete event and hybrid systems. Annu Rev Control 45:267–280
18. Padilha R, Iano Y, Monteiro ACB, Arthur R, Estrela VV (2019) Betterment proposal to multipath fading channels potential to MIMO systems. In: Proceedings of the 4th Brazilian technology symposium (BTSym'18): emerging trends and challenges in technology, vol 1. Springer, p 115
19. França RP, Iano Y, Monteiro ACB, Arthur R, Estrela VV, Assumpção SLDL, Razmjooy N (2019) Potential proposal to improvement of the data transmission in healthcare systems
20. Mishra M, Mattingly J, Mueller JM, Kolbas RM (2018) Frequency domain multiplexing of pulse mode radiation detectors. Nucl Instrum Methods Phys Res Sect A 902:117–122
21. Bertsekas DP, Scientific A (2015) Convex optimization algorithms. Athena Scientific, Belmont
22. Bubeck S (2015) Convex optimization: algorithms and complexity. Found Trends Mach Learn 8(3–4):231–357
23. Jensen WA (2017) Response surface methodology: process and product optimization using designed experiments. J Qual Technol 49(2):186
24. Khuri AI (2017) Response surface methodology and its applications in agricultural and food sciences. Biom Biostat Int J 5(5):1–11
25. Lowndes V, Berry S, Parkes C, Bagdasar O, Popovici N (2017) Further use of heuristic methods. In: Guide to computational modelling for decision processes. Springer, Cham, pp 199–235
26. Kramer O (2017) Genetic algorithm essentials, vol 679. Springer
27. Mousavi BS et al (2014) Semantic image classification by genetic algorithm using optimised fuzzy system based on Zernike moments. SIViP 8(5):831–842
28. Vikhar PA (2016) Evolutionary algorithms: a critical review and its future prospects. In: 2016 international conference on global trends in signal processing, information computing and communication (ICGTSPICC). IEEE, pp 261–265
29. Rabadi G (ed) (2016) Heuristics, metaheuristics and approximate methods in planning and scheduling, vol 236. Springer
30. Wiggins B, Berry S, Lowndes V (2017) The design and optimisation of surround sound decoders using heuristic methods. In: Guide to computational modelling for decision processes. Springer, Cham, pp 273–284
31. Audet C, Hare W (2017) Nelder-Mead. In: Derivative-free and blackbox optimization. Springer, Cham, pp 75–91
32. Mead R (2017) Statistical methods in agriculture and experimental biology. Chapman and Hall/CRC
33. Ravindran AR, Warsing Jr DP (2016) Supply chain engineering: models and applications. CRC Press
34. Wang FK, Tamirat Y (2016) Multiple comparisons with the best for supplier selection with linear profiles. Int J Prod Res 54(5):1388–1397
35. Padilha R, Martins IB, Moschim E (2016) Discrete event simulation and dynamical systems: a study of art
36. Padilha RF (2018) Proposta de um método complementar de compressão de dados por meio da metodologia de eventos discretos aplicada em um baixo nível de abstração [Proposal of a complementary method of data compression by discrete event methodology applied at a low level of abstraction]
37. Namadchian A et al (2016) A new meta-heuristic algorithm for optimization based on variance reduction of Gaussian distribution. Majlesi J Electr Eng 10(4):49
38. Tian MW et al (2019) New optimal design for a hybrid solar chimney, solid oxide electrolysis and fuel cell based on improved deer hunting optimization algorithm. J Clean Prod 119414
39. Mir M et al (2019) Employing a Gaussian particle swarm optimization method for tuning multi input multi output-fuzzy system as an integrated controller of a micro-grid with stability analysis. Comput Intell

40. Cao Y et al (2019) Experimental modeling of PEM fuel cells using a new improved seagull optimization algorithm. Energy Rep 1(5):1616–1625
41. Gendreau M, Potvin JY (eds) (2010) Handbook of metaheuristics, vol 2. Springer, New York
42. Hussain K, Salleh MNM, Cheng S, Shi Y (2019) Metaheuristic research: a comprehensive survey. Artif Intell Rev 52(4):2191–2233
43. Bäck T, Fogel DB, Michalewicz Z (eds) (2018) Evolutionary computation 1: basic algorithms and operators. CRC Press
44. Kurniasih J, Utami E, Raharjo S (2019) Heuristics and metaheuristics approach for query optimization using genetics and memetics algorithm. In: 2019 1st international conference on cybernetics and intelligent system (ICORIS), vol 1. IEEE, pp 168–172
45. Bhattacharyya S (ed) (2018) Hybrid metaheuristics for image analysis. Springer
46. Siarry P (ed) (2016) Metaheuristics, vol 23. Springer, Switzerland
47. Fleury G, Gourgand M, Lacomme P (2010) Metaheuristics for the stochastic hoist scheduling problem (SHSP). Int J Prod Res 39(15):3419–3457
48. Escario JB, Jimenez JF, Giron-Sierra JM (2012) Optimisation of autonomous ship manoeuvres applying ant colony optimisation metaheuristic. Expert Syst Appl 39(11):10120–10139
49. Almeder C, Hartl RF (2013) A metaheuristic optimization approach for a real-world stochastic flexible flow shop problem with limited buffer. Int J Prod Econ 145(1):88–95
50. Latorre-Biel JI (2014) Control of discrete event systems by means of discrete optimization and disjunctive colored PNs: application to manufacturing facilities. Abstr Appl Anal 2014
51. Angel JA (2015) A review of simheuristics: extending metaheuristics to deal with stochastic combinatorial optimization problems. Oper Res Perspect 2:62–72
52. Fikar C (2016) A discrete-event driven metaheuristic for dynamic home service routing with synchronised trip sharing. Eur J Ind Eng 10(3):323–340
53. Vieira GE (2017) Evaluating the robustness of production schedules using discrete-event simulation. IFAC-PapersOnLine 50(1):7953–7958
54. Bamporiki T, Bekker J (2018) Development of a discrete-event, stochastic multi-objective metaheuristic simulation optimisation suite for a commercial software package. S Afr J Ind Eng 29(3):12–25
55. Amodeo L, Talbi EG, Yalaoui F (eds) (2018) Recent developments in metaheuristics. Springer International Publishing
56. Dorigo M, Stützle T (2019) Ant colony optimization: overview and recent advances. In: Handbook of metaheuristics. Springer, Cham, pp 311–351
57. Fishman GS (2013) Discrete-event simulation: modeling, programming, and analysis. Springer Science & Business Media

An AWGN Channel Data Transmission Proposal Using Discrete Events for Cloud and Big Data Environments Using Metaheuristic Fundamentals

Reinaldo Padilha, Yuzo Iano, Ana Carolina Borges Monteiro, and Rangel Arthur

Abstract Cloud computing consists of providing computing services and resources, including servers, storage, databases, networking, software, analytics, and intelligence, over the Internet ("the cloud") to deliver faster innovation, flexible capabilities, and economies of scale. Big Data refers to a large amount of data, coming from controlled or uncontrolled sources, in accordance with structured or unstructured data, its potential comes from this volume where through its analysis it allows extracting insights from analyzes of this information. Metaheuristics have proven to be a powerful tool for roughly solving optimization problems, applied to find answers to problems about which there is little information. This research aims to propose modeling to improve the transmission of content in wireless communication systems, employing the pre-coding process of bits based on the application of a discrete event in signals before the modulation process, using DQPSK modulation in an AWGN channel. The results show improvement achieving 74.61% in memory utilization and 131.29% at runtime with respect to information compression.

Keywords Metaheuristics · Cloud computing · Big data · Data · DQPSK · Modeling · Simulation

R. Padilha (✉) · Y. Iano · A. C. B. Monteiro · R. Arthur
School of Electrical and Computer Engineering (FEEC), University of Campinas—UNICAMP, Av. Albert Einstein—400, Barão Geraldo, Campinas, SP, Brazil
e-mail: padilha@decom.fee.unicamp.com

Y. Iano
e-mail: yuzo@decom.fee.unicamp.com

A. C. B. Monteiro
e-mail: monteiro@decom.fee.unicamp.com

R. Arthur
e-mail: rangel@ft.unicamp.br

N. Razmjooy et al. (eds.), *Metaheuristics and Optimization in Computer and Electrical Engineering*, Lecture Notes in Electrical Engineering 696,
https://doi.org/10.1007/978-3-030-56689-0_15

1 Introduction

The increase in the number of connected people is growing more and more around the world. This is due to the momentum of mobile devices, whether smartphones, as well as the computer, the quality of the internet, has also improved over the years, the speed improvement as well as the connection.

Big Data is related to large storage of data captured with velocity, variety, truth and value, the term is used to refer to the huge amount of data produced and stored daily, whether by people, companies or technological devices, where this data is divided in groups, analyzed and related by algorithms [1–7].

The importance of big data is not just about how or how much information the company gets, but what it does with that data, how it interprets and uses all that data for the purpose of making it a resource, i.e., using that data smart way to improve business [1–7].

Enterprises take advantage of Big Data through better comprehensive information management, as technology has tools that have built-in solutions, methodologies, and statistical models that enable you to manage this large amount of data by improving critical operating processes and thereby leading to better business intelligence. Strategic decisions, based on this knowledge generated from this data volume, improving the company's competitive advantages [1–7].

Big Data has the philosophy that information exists and you need to know how to interpret it quickly. Since consumers are increasingly demanding and the market is even more competitive, forcing daily innovations as an essential premise for success depending on the type of business. The exploitation of captured data, being processed and analyzed, tends to help companies improve their performance and therefore their relationship with the consumer significantly [1–7].

Cloud computing is the practice of using a network of remote servers, hosted on the internet, having the purpose of storing, managing and processing data, avoiding the need for local server deployments or infrastructure. In short, it concerns the ability to execute processes over the Internet, without the need to install programs or download files directly to the computer [8–19].

The ever-evolving technology brings with it a growing demand for innovations that make everyday life easier for people and businesses, so cloud computing is gaining more market share by offering a host of benefits to its end users [8–19].

Cloud computing gives users access to data access wherever they are, as long as there is an internet connection, whether through a tablet or smartphone, computer, in any environment, whether at the office or university, cloud computing provides easy access to information [8–19].

The business impact of the cloud is so great that it goes beyond the IT sectors of the enterprise, also reaching the financial sector, making companies that have adopted cloud technology improve their time to market, process efficiency and cost-saving in IT (Information Technology) [8–19].

In general, metaheuristics are techniques used in situations where mathematical or logical modeling is used for problem-solving by combining basic heuristics at a high-er structure level [20–27].

These are commonly used strategies to solve NP-Hard Problem problems since they offer better overall solutions with less processing time compared to other types of techniques. In short, metaheuristics are optimization tools and a viable alternative for solving complex or large problems, where a large number of variables and constraints make the use of exact methodologies unfeasible [20–27].

In general, meta-heuristics use a combination of random choices and historical knowledge of the previous results acquired by the method to guide and search the search space in neighborhoods within the search space, avoiding premature stoppages in optimal locations. A strategy that guides or modifies a heuristic to produce solutions that surpass the quality of those commonly encountered [20–27].

Fading can change over time, degrading the communication system performance with respect to a loss of signal power without decreasing the noise power over signal bandwidth. Experiencing fading in mobile channels has a proportional probability that all component channels simultaneously experience a fade, causing the channel drops on the channel limits to link performance related to bit errors as the Signal-to-Noise Ratio (SNR). The fading effects can be softened by incorporating diversity in signal transmission over multiple channels that face independent fading [28–33].

Multipath fading is when in a radio communications broadcast, the links interfere in one way or another, and the paths change causing propagating multiple versions of signals transmitted across different paths before they reach the receiver, still affecting their phases as there is the variability of path lengths and then distorting the radio signal [28–33].

Since the received signal is susceptible to several unstable factors, it is advisable to use statistical mathematical models that help to simulate fading and design mitigation proposals, such as Rayleigh, where their delays associated with different signal paths in a multipath fading channel change in an unexpected manner can only be statistically characterized, since it is a statistical model of propagation in a radio signal environment, used for environments with many objects in which they spread the radio signal before reaching the receiver, it is advisable for strongly constructed transmission models where there is no line of sight. Between transmitter and receiver [28–33].

The discrete event technique models the representation of a system as a sequence of operations by state transactions (entities), where these entities are discrete, and may be relative to various types depending on the context of the problem that is being sought, such as data. Packets, or bits, in the case of this search. The technique is generally used in modeling concepts of high-level abstraction, emails on a server, clients in a queue, flow of vehicles, or even transmission of data packets [34–45].

In this scenario, this chapter proposes the modeling of the wireless system built upon an AWGN channel with signals transmitted over the advanced modulation format (DQPSK), providing improved transmission capacity of information through this channel, following approaches and guidelines of metaheuristic science with regard to developing viable solutions of good quality, with design based on the characteristics of a problem with reduced complexity in relation to that of traditional exact modeling, being unique, oriented to the overall optimization of a problem and applicable to that particular problem [28–33].

Thus, the bit treatment with discrete events technique was developed by differentiating the application under the low abstraction level, relative to the bit generation step, which showed better computational performance regarding memory utilization as the runtime in a simulation environment.

The chapter is organized following in Sect. 2 discussing cloud computing and big data technologies and their link, Sect. 3 discusses the thematic concepts and fundamentals about heuristics and metaheuristics, Sect. 4 presents a proposed approach explaining the modeling used as well as the tools used, in Sect. 5 presents the results obtained, in Sect. 6 a brief discussion is held and finally, in Sect. 7, the conclusions are aligned in the same way as the potential of the research is delimited.

2 Cloud Computing and Big Data

Big Data is related to all the mechanism and technological equipment used to deal with the immense amount of data (structured and unstructured) growing exponentially every moment in the world [1–7, 46–51]. Cloud computing is the concept of using various types of applications over the Internet, as if these applications were installed on users' devices, anywhere and regardless of device or platform, is related to memory usage, processing, services and storage and computing capabilities for shared and web-connected computers and servers [8–19, 52–54], as illustrated in Fig. 1.

The link between these two concepts is due to the need for Big Data (volume, velocity, and variety), requiring an infrastructure that allows for large scale and constantly growing, processing, storage and retrieval of various types of data [8–19, 52–54].

With the technological advancement related to communication and interaction with the Internet increasingly present, necessary and widely used by the population, it provides a large amount of information and requires a high level of security.

Cloud computing has allowed the emergence of new models of information technology acquisition and commercialization, especially for companies (IT—Information Technology—services era), since these platforms have brought great benefits to com-panies and society, facilitating access to information in anywhere and the ability to store large amounts of data, along with Infrastructure as a Service (IaaS), Software as a Service (SaaS), and Platform with Service (PaaS) concepts [8–19, 52–54].

With Big Data there is the ability to understand a large amount of data, doing it all at once, where the sense of velocity is important, to do these analyses in less time; along with the ability to analyze unstructured data from different sources, having competence to visualize future situations and execute decision-making from them, based on the effective interest deduction of events, as illustrated in Fig. 2. With cloud computing, in addition to accessing data from anywhere, it is possible to pay only for

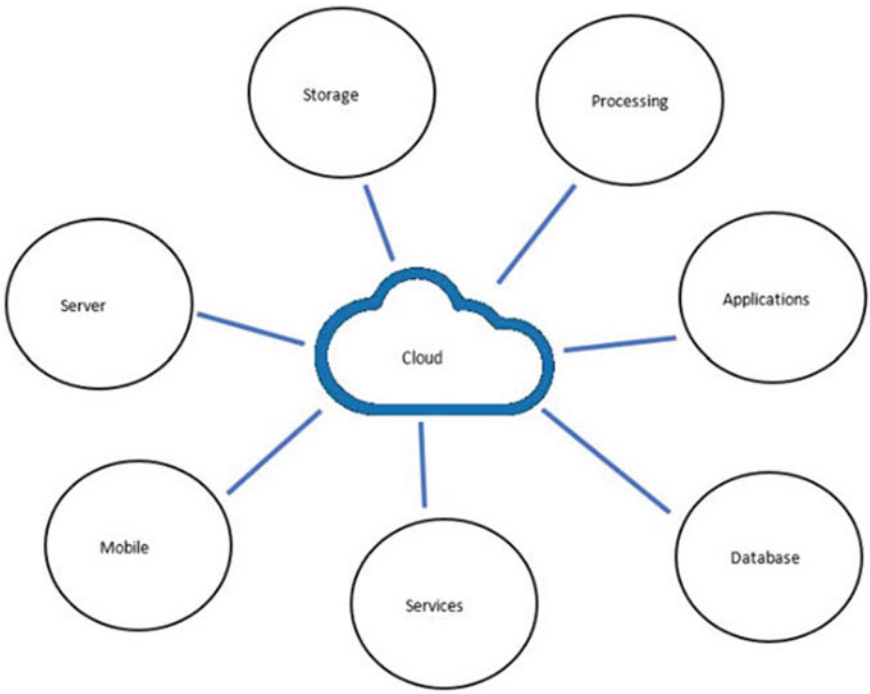


Fig. 1 Cloud computing concept

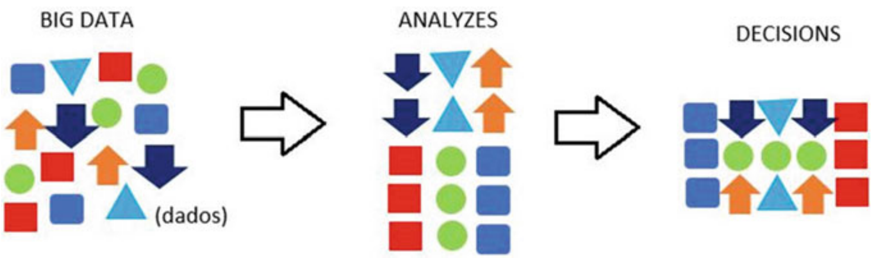


Fig. 2 Big data concept

usage and only what is needed, avoiding wasted resources, and along with scalability, resource availability is increased as needed [1–7, 46–51].

Software as a Service (SaaS) is the model in which companies purchase a computer system without having to use licenses, the software is not allocated internally and can be accessed from anywhere and with any mobile device via the web, it is paid only for the functionalities and for the contracted usage time [8–19, 52–54].

Infrastructure as a Service (IaaS) is the model for contracting IT infrastructure in a virtual way, it is paid according to the space used, the amount of data trafficked and stored, among other contracted items such as devices, servers, hardware, and soft-ware, giving more flexibility, cost control, scalability [8–19, 52–54].

Platform as a Service (PaaS) is the model for building and using applications without requiring hardware and software infrastructure maintenance [8–19, 52–54].

While big data provides insight into existing information, cloud computing is focused on simplifying the environment, reducing the degree of infrastructure investment [1–7, 46–51]. Since "technology engine" has become more robust, in view of this there must be consecutively also a steady decrease in costs brought about by increasingly specialized cloud computing [8–19, 52–54]. Where analysis, structuring, and decision-making from customer/person-generated data is most possible, it facilitates product and service innovation as well as relationships.

Thus, Big Data refers to concepts regarding the storage, processing, and study of data, usually from multiple sources [1–7, 46–51]. Cloud refers to the computing environment or infrastructure where the use occurs, from the users' point of view regardless of its location or the way data is stored or processed [8–19, 52–54].

In short, the Cloud is where Big Data can reside because of its technical qualities, especially elasticity and charge-for-use, rapid interpretation of information and knowledge creation for more assertive communication.

Finally, dialogue exists through Big Data tools within the infrastructure offered by Cloud.

3 Heuristics and Metaheuristics

As the telecommunications industry introduces new technologies into the marketplace, the volume of services offered to users tends to evolve, increasingly combining and disseminating technologies for services that use data, text, voice, and video. Where such improved changes imply new aspects in the planning of a telecommunications network, there is a need for improvement of optimization issues of these services [20–27, 55–60].

Where heuristics are found to be used in these contexts, they generally generate viable solutions of good quality, but without quality assurance, being understood as any approximate method, is designed based on the structural properties and characteristics of these problems, with reduced complexity in relation to that of conventional exact algorithms [20–27, 55–60].

Starting from a historical horizon, the term is derived from the Greek word heuriskein, meaning to discover, being related to describe methods based on judgment or experience, which leads to a good solution of a given problem, but does not guarantee the production of an optimal solution [20–27, 55–60].

In this sense, a heuristic is a logic designed to find a good solution, but that does not guarantee the quality of the optimal solution, is used to find reasonable solutions; It may also associate the term with a mathematically unverifiable, circumstantial knowledge [20–27, 55–60].

Since logic is considered a heuristic method when there is no complete mathematical knowledge about its behavior, i.e., usually with respect to time consumption, finding good quality solutions, it offers no guarantees that logic aims to solve complex problems using a not too large amount of resources. Thus, heuristic is unique and can only be applied to that particular problem which it is designed to have two common features that are rapidity (related to the requirement of the heuristic that can produce a satisfactory solution in a reasonable amount of time) and some applied strategy for the project [20–27, 55–60].

Metaheuristic derives from the composition of two obviously heuristic Greek words, along with meta, meaning "after", which indicates a higher level. It consists of several generic heuristics that adapt and are directed to the global optimization of a problem and may contain different heuristic procedures in its structure. Thus, it can be understood as a set of concepts that can be used in the definition of heuristic methods, which can be applied to a broad set of different problems [20–27, 55–60].

In other words, a metaheuristic can be understood as a strategy that tries to efficiently explore the space of viable solutions to this problem, where one has specific knowledge of the problem and can be used as a heuristic to assist in the process. In short, metaheuristics are high-level mechanisms that exploit a particular type of strategy for a specific solution [20–27, 55–60].

Although all the procedures described by metaheuristics can be seen as algorithmic search strategies, besides the fact that these methods have been widely studied in the last decades, which deepens the knowledge about the problem-solving process. complexes; in fact, its importance is due to other characteristics as regards the description of methods that provide ideas that can be applied to optimization problems for which efficient (sometimes not even heuristic) specific algorithms are known [20–27, 55–60].

3.1 Scientific Review

It was made a review based on research on metaheuristic papers in conjunction with discrete-event technology exploring a historical review and applicability of techniques data in the last 5 years, with emphasis on publications and indexing in renowned databases.

In 2014 was studied whereas artificial intelligence methodologies, such as the core of discrete control and decision support systems, have been extensively applied in the industrial production sector, resulting in tools producing excellent results, yet the difficult nature of many discrete controls or decision-making problems. Manufacturing decision making may require inaccessible computational resources, limited by the available time needed to arrive at a solution, thus applying a metaheuristic of genetic algorithms applied in the search for the best solutions in the solution space [61].

In 2015 was studied many combinatorial optimization problems encountered in real-world logistics, transportation, production, healthcare, finance, telecommunications, and computing applications are NP in nature, where metaheuristics have been noted to benefit from different randomization and parallelization paradigms, but often assume that the problem inputs, the underlying objective function, and the set of optimization constraints are deterministic, wherewith this focus a general methodology has been described that allows you to extend metaheuristics through simulation to solve stochastic combinatorial optimization problems, which It was also seen the use of simulation combining continuous-time, discrete-event, and state-transition systems, which mimics the production system, since they use the mathematical optimization model to replicate the decision problem [62].

In 2016 was studied domestic service sectors where vehicles are used to visit customers with the focus of reducing the number of vehicles needed, however, there is complexity in coordinating the arrival times of employees and vehicles flexible discrete event-oriented heuristics to handle dynamic routing and scheduling scenarios using combined travel and hiking sharing [63].

In 2017 was studied complex workshop stochastic scheduling problems can be addressed by simulation-based optimization by combining meta-heuristic optimization capabilities with the system representation of simulation models, and exploring the potential of coupling optimization and simulation techniques in different scenarios. Workshop scheduling, being developed a genetic algorithm combined with discrete event simulation [64].

In 2018 was presented the development of a project optimization set focusing on a hybrid metaheuristic optimization set to solve multi-objective simulation optimization problems using a third-party library to solve discrete-event stochastic simulation optimization problems integrated with Siemens Tecnomatix Plant Simulation, a simulation software package developed by Siemens [65].

4 Proposed Approach

In a context of data transmission using an AWGN channel model that naturally reproduces the noise present in today's digital communication, being the received signal equals to the transmitted signal plus noise, relative to additive characteristic; still this noise is intrinsic to the system representing the term white, and being statistical with a normal distribution, relative to Gaussian characteristic [28–33].

In the frequency domain, white noise generates losses to all frequencies leading to a uniform power spectrum over the course of all frequencies, and in the time domain follows a normal distribution with respect to the fact the probability distribution of the noise samples, regarding a Gaussian aspect. Still bearing in mind that the AWGN cane model is used for its simplicity in mathematical modeling [28–33].

Likewise, mobile wireless channels are affected to various interferences including fading, shadowing, multipath and various types of noise which cause significant degradation in the transmission of these signals and respectively in the performance of the system [28–33].

Rayleigh is the form of fading that usually occurs in an environment where a large number of reflections are present, in a heavily built urban environment with no dominant propagation along with the transmitter and the receiver, typically occurring when one of the paths a line-of-sight signal, is much stronger than the others. Caused by multipath reception, where the antenna receives a large number of reflected and scattered waves and, due to the effects of wave cancellation (by the partial cancellation of a radio signal by itself), the instant effects received by the energy seen by a Moving antenna becomes a random variable, depending on the location of the antenna, as long as the signal arrives at the receiver exhibiting multipath interference, with one of the paths being changing at least lengthening or shortening [28–33].

A DQPSK modulation is a specific form of QPSK modulation, which can be used to send a symbol corresponding to a pure phase parameter, this symbol represents a phase variation, in short it is a differential format in which the bits for a given symbol are determined by the phase change of the previous symbol, where each set of bits represented by a symbol causes a phase variation determined on the carrier signal, can be found with a total of 8 ideal state positions in its constellation, where the ideal state positions for symbols alternate between the four 45-degree states commonly used by QPSK and four on-axis states, in a configuration for/4 offset to the phase changes [28–33], as shown in Fig. 3.

The research abstracted the characteristics, from the theoretical foundations and fundamentals of heuristic and metaheuristic science with regard to developing viable

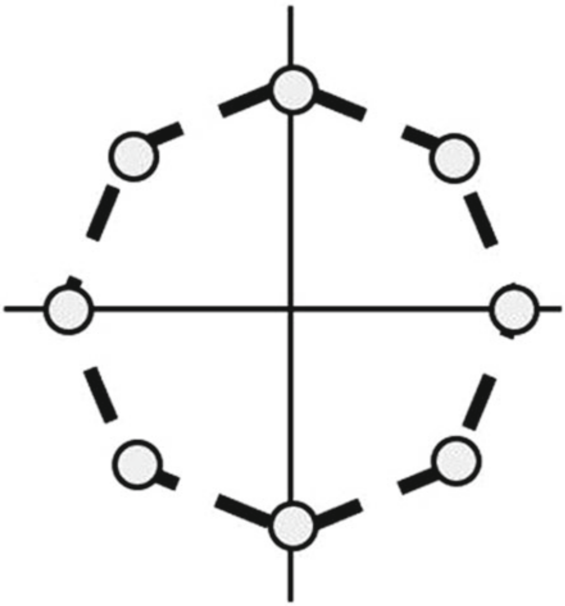


Fig. 3 Theoretical constellation DQPSK

solutions of good quality, being designed based on the characteristics of a problem with reduced complexity in relation to that of traditional exact modeling, being unique and can only be applied to that particular problem, oriented to the overall optimization of a problem; i.e., in improving the transmission of information on an AWGN channel by employing the discrete event technique on the physical layer, reaching the bit. This was done by using the Simulink environment of the MATLAB software, 64-bit version (2014a).

Figure 4 presents a model where the signals corresponding to bits are generated and then modulated in DQPSK, passing through a multipath Rayleigh fading channel with Jakes model with Doppler shift defined at 0.01 Hz, as also inserted a block incorporated which has a math function 1/u, which tracks the channel time-variability where the receiver implementation commonly encompasses an automatic gain control (AGC). So, the AWGN channel contains parameters specified at a sample time of 1 s, 1 W of input signal power, initial seed in the generator of the 37th and the 67th channels, with Eb/No ranging from 0 to 13 dB. Next, this signal is demodulated to perform the bit error rate (BER) of the channel processing and generating of the signal BER graph.

The addition of the discrete event technique in modeling follows first that the time-based signal must be converted to a specific type following the output parameter, an integer (the bit), through the Real-World Value (RWV) function, preserving the current value of the input signal, then rounding with the floor function, rounding the values to the nearest smallest integer, and finally doing a zero-order hold (ZOH), causing the reconstruction of the signal to the time domain and supporting each sample value by a specific time interval, performing the conversion back of the signal; and after the signal continues to be modulated, going through the AWGN channel, and then demodulated.

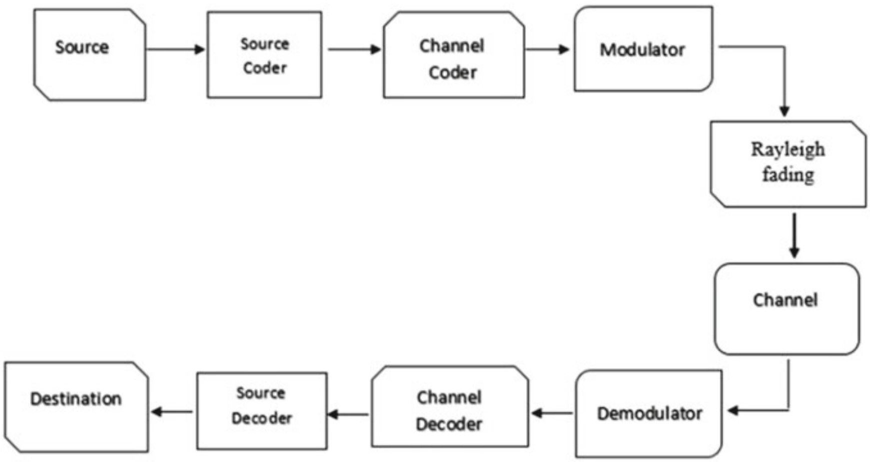


Fig. 4 Demonstration of traditional model

The model from Fig. 5 incorporates the conventional method with the proposed, and it also highlights the part modeled using discrete events in blue.

Figures 6 and 7 use 1000 s of simulation time, showing transmission for the DQPSK signal with Rayleigh fading (Fig. 6) in the proposed (right) and conventional modeling (left), noting that both generated the same transmission stream. The same can be seen in the constellation diagram (Fig. 07), viewing the points of the modulated digital signal, being used 13 dB for the proposed (left) and the traditional methods (right) in multipath fading with Rayleigh.

Was also used the “compass” function which displays compass graph with *n* arrows (related to PSKs) and how they are arranged in constellations with radial representa-tions of points, where n is the number of elements in Z, and the location of the base of each arrow is respective to the origin. In this sense in Fig. 8 is shown the relation-ship between a conventional methodology and the proposed methodology,

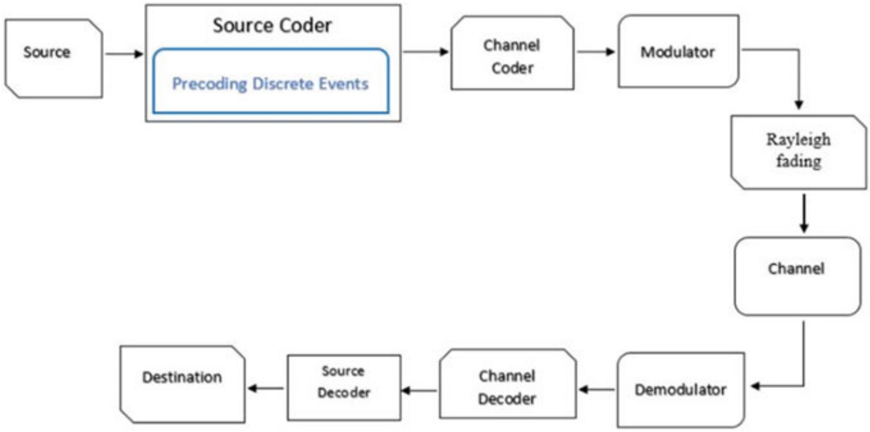


Fig. 5 Model with the proposal of this study

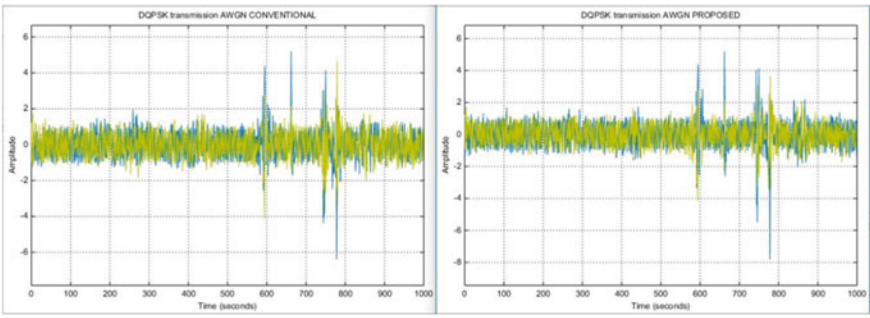


Fig.6 Transmission flow DQPSK with Rayleigh fading

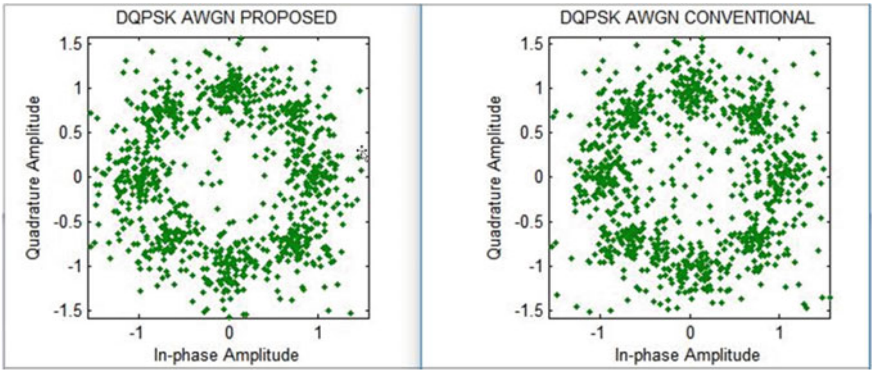


Fig.7 Simulated constellations DQPSK with Rayleigh fading

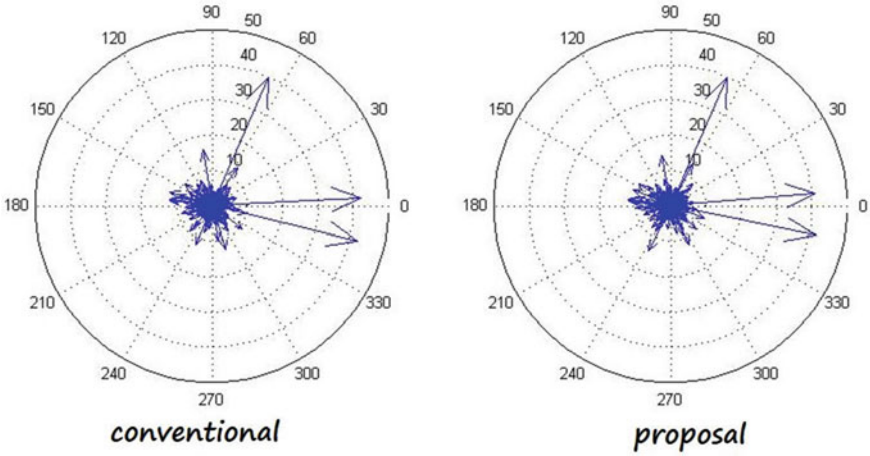


Figure 8. Simulated constellations DQPSK with Rayleigh fading

both simulated with 13 dB and 50,000 s of simulation time, making an analogy with Fig. 7 addressing the DQPSK constellation.

5 Results

The results correspond to a sequential simulation performed on the modeling presented in the previous session on a physical machine consisting of an Intel Core i3 processor and 4 GB RAM.

Using the Simulink environment along with the sldiagnostics function, allowed to display the diagnostic information about the modeling system, through the Process-MemUsage parameter obtains the sum of all the memory consumption for all model processes in the whole simulation and performs the counting returning the total amount of memory in use over each phase of the model in MB.

Together with the calculation of the simulated time spent is on the Elapsed Time parameter, being responsible for the accounting of the time spent in each phase of the simulation, in seconds, corresponding to the real-time of the model being simulated.

Both analyzes taking into account that is in the first simulation of both model-ing, that the variables are allocated, just as the memory is reserved for their execution, according to Figs. 9 and 10.

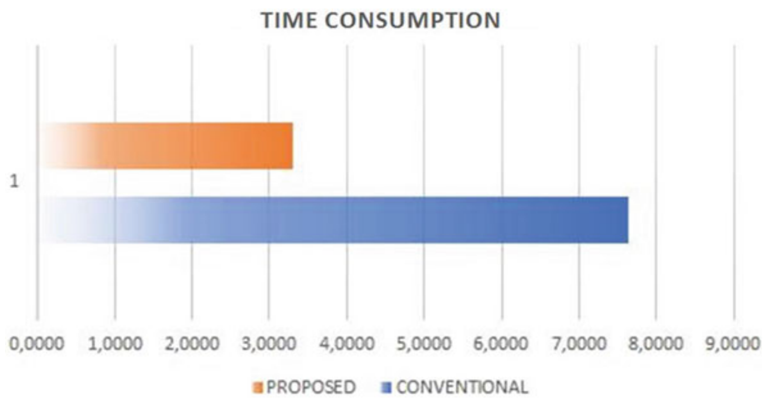


Fig. 9 Time consumption for DQPSK Rayleigh

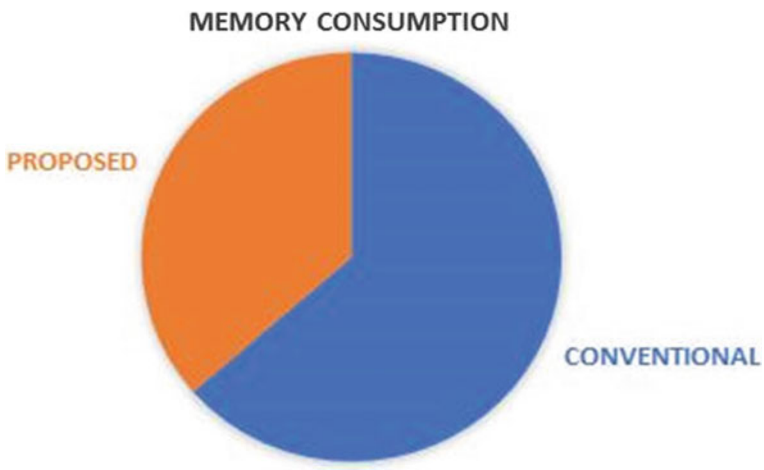


Fig. 10 Memory consumption for DQPSK Rayleigh

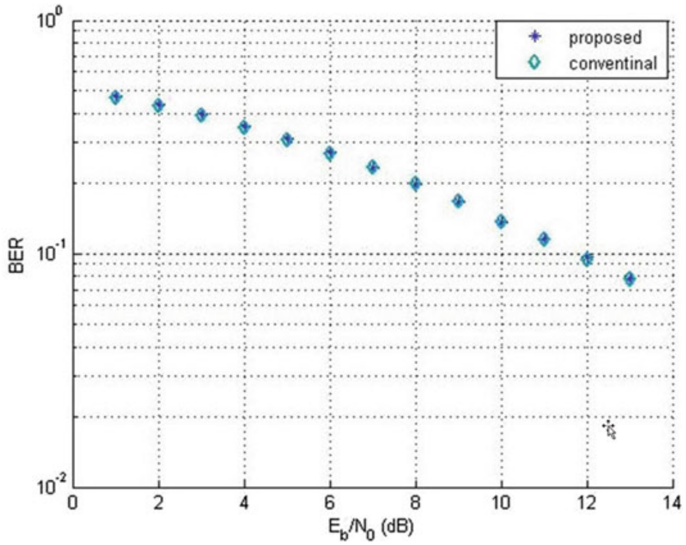


Fig. 11 BER DQPSK Rayleigh

In this bias, it is understood that if in a transmission channel containing the proposed modeling and in another the conventional, they passed the same information content (quantity of bits), without any loss (signal and constellation) and with the same qual-ity (BER), soon there was transmitted information compression.

Tables 1 and 2 are respectively from Figs. 9 and 10, related to the memory, where is shown **74.61%** improvement and regarding time consumption, showing an improvement of **131.29%** with respect to modeling.

Noise and interference are representative of the energy level as opposed to the desired signal, where it represents the energy level arriving at the receiving receiver from the transmitter, after the loss of intensity in free space, so, the difference between the signal and noise is calls the signal-to-noise ratio, that is, SNR (Signal-to-Noise Ratio)., which can be measured through BER [28–33].

Thus, analyzing the relationship between the simulation of the proposed × conventional modeling, and their impacts on the physical layer of the channel, it can be seen that both generated the same result through the graph BER, during transmission with noise ranging from 0 to 13 dB, according to the Figs. 11 [30–65].

6 Discussion

With the advent and merging of cloud computing and big data technologies, it is possible to accelerate processing time to gain insights leaving behind all the complexities of data analysis, proving the benefits of physical serverless data analysis services and being integrated with the server cloud-based and complete technologies that overcome the limitations of scale, performance, and cost-efficiency, solutions that rely on big data technology to manage both traditional and new datasets on a single cloud platform.

As can be seen throughout the text, the union of discrete events with metaheuristic science, whether direct or not, is successful. As it was presented the proposal using the technological ideas of metaheuristic together with its application fundamentals with regard to developing viable solutions of good quality, based on the characteristics of a determined and specific problem with reduced complexity in relation to that of traditional exact modeling, generating a unique solution being applied to that particular problem, oriented to the overall optimization of this problem, that is, to develop a new approach to signal transmission, performed in the discrete domain with the implementation of discrete entities in the bit generation process, resulting in positive results of time and memory consumption, which is possible to increase the capacity of information transmission for communication systems, even with the differential use of discrete events in a lower level of application acting on the physical layer.

Which contributes proportionately to the approaches to technologies in use in big data and cloud together, where the proposal can help in managing this pro-cessing by removing operational overhead by addressing performance, scalability, availability, security and compliance needs, according to Big Data analysis solutions, since an efficient and fluid data transmission flow is required.

Also taking into account that merging through cloud technologies provides greater integration for big data when it comes to using different types of data, rang-ing from batch data to streaming data, transforming it so it can be leveraged; driving greater management for big data by storing and processing your data more efficiently, securely and reliably by utilizing big data analytics built into cloud servers that uncover new information with advanced analytics and visualization.

Which is strongly linked to the proposal of this chapter which considers the velocity whether being used by a user, or organization, is a great allied to cloud computing systems, since the result pointed out 74.61% in memory utilization and 131.29% at runtime with respect to information in traffic on the communication channel.

Since the processing power of big data technologies nowadays brings the ability to analyze from gigabytes to petabytes of data at an incredibly fast speed when working in conjunction with cloud technologies, since easy integration between them is easily possible. technologies. Which no matter whether the data is at rest or in motion, internal or external data, all data is prepared and analyzed so that it can be used as decision-making items.

Table 1 Memory consumption

Machine	i3
Conventional modeling	159.4727
Proposed modeling	91.3320

Table 2 Time consumption

Machine	i3
Conventional modeling	7.6350
Proposed modeling	3.3010

7 Conclusions

As can be seen, metaheuristic techniques have varied applications in different areas of science, having positive characteristics as to the implementation and the great efficiency of these techniques in relation to conventional heuristics with great potential of application has been in the problems related to genetic algorithms.

The relevance of big data is not about the amount of data but about what is done with it, providing options for obtaining data from multiple sources and per-forming analysis that results in cost savings, time savings, new product development, and more. Optimization of products and offerings, as well as helping to make smarter decisions.

Cloud computing has enabled us to live the reality of making big data actionable, where you can bring all your acquired data sources together and quickly and easily make them available, combining with Big data technology is possible to per-form with high performance, risk calculations for the organization, a determination of failures pointing causes, problems and defects almost in real-time, i.e. direct aid in decision making.

And finally, this chapter showed that the proposal has great potential in the improvement of communication services potential, indicating better computational performance-related 74.61% in memory utilization and 131.29% at runtime related to information in traffic.

References

1. Assunção MD, Calheiros RN, Bianchi S, Netto MA, Buyya R (2015) Big Data computing and clouds: trends and future directions. J Parallel Distrib Comput 3–15
2. Bhat SA, Singh A (2017) Review on effective image communication mod-els. Austral J Basic Appl Sci 65–79
3. Bihl TJ, Young II WA, Weckman GR (2016) Defining, understanding, and addressing big data. Int J Bus Anal (IJBAN) 1–32
4. Dašic P, Dašic J, Crvenkovic B (2016) Service models for cloud computing: Search as a service (SaaS). Int J Eng Technol 2366–2273

5. Erevelles S, Fukawa N, Swayne L (2016) Big Data consumer analytics and the transformation of marketing. J Bus Res 97–904
6. Gandomi A, Haider M (2015) Beyond the hype: Big data concepts, methods, and analytics. Int J Inf Manag 137–144
7. Grover V, Chiang RH, Liang TP, Zhang D (2018) Creating strategic business value from big data analytics: a research framework. J Manag Inf Syst 388–423
8. Razmjooy N, Khalilpour M, Ramezani M (2016) A new meta-heuristic optimization algorithm inspired by FIFA world cup competitions: theory and its application in PID designing for AVR system. Journal of Control, Automation and Electrical Systems 27(4):419–440
9. Madni SHH, Latiff MSA, Coulibaly Y (2016) Resource scheduling for infrastructure as a service (IaaS) in cloud computing: challenges and opportunities. J Network Comput Appl 173–200
10. Pahl C (2015) Containerization and the PAAS cloud. IEEE Cloud Comput 24–31
11. Patel A, Patel M, Singh PK (2016) Study of security in the hybrid cloud. Int J Res Adv Computer Sci Eng 01–05
12. Rittinghouse JW, Ransome JF (2017) Cloud computing: implementation, management, and security. CRC press
13. Sen J (2015) Security and privacy issues in cloud computing. Cloud Technology: concepts, methodologies, tools, and applications. Information Resources Management Association, pp 1585–1630
14. Rittinghouse JW, Ransome, JF (2016) Cloud computing: implementation, management, and security. CRC press
15. de Bruin B, Floridi L (2017) The ethics of cloud computing. Sci Eng Ethics 23(1):21–39
16. Chang V (2015) A proposed cloud computing business framework. Nova Science Publisher
17. Rani BK, Rani BP, Babu AV (2015) Cloud computing and inter-clouds–types, topologies and research issues. Procedia Comput Sci 50:24–29
18. Schneider S, Sunyaev A (2016) Determinant factors of cloud-sourcing decisions: reflecting on the IT outsourcing literature in the era of cloud computing. J Inf Technol 31(1):1–31
19. Botta A, De Donato W, Persico V, Pescapé A (2016) Integration of cloud computing and internet of things: a survey. Fut Gener Comput Syst 56:684–700
20. Blum C, Roli A (2003) Metaheuristics in combinatorial optimiza-tion: Overview and conceptual comparison. ACM Compuzt Surveys (CSUR) 35(3):268–308
21. Nesmachnow S (2014) An overview of metaheuristics: accurate and efficient methods for optimisation. Int J Metaheurist 3(4):320–347
22. Razmjooy N, Estrela VV, Loschi HJ, Fanfan W (2019) A comprehensive survey of new meta-heuristic algorithms. Recent Advances in Hybrid metaheuristics for data clustering. Wiley, New York
23. Chiarandini M et al (2007) Experiments on metaheuristics: methodological overview and open issues. Technical Report DMF-2007-03-003. The Danish Mathematical Society, Denmark
24. Cao Y et al (2019) Multi-objective optimization of a PEMFC based CCHP system by meta-heuristics. Energy Rep 1(5):1551–1559
25. Razmjooy MR, Ramezani M (2016) Model order reduction based on meta-heuristic optimization methods. In: 1st international conference on new research achievements in electrical and computer engineering, Iran
26. Šenkeřík R (2020) A brief overview of the synergy between metaheuristics and unconventional dynamics. Lecture notes in electrical engineering
27. Tozer EP (2012) Broadcast engineer's reference book, 1th edn. Focal Press
28. Rama Krishna A, Chakravarthy ASN, Sastry ASCS (2016) Variable modulation schemes for AWGN channel based device to device communication. Indian J Sci Technol 9(20). https://doi.org/10.17485/ijst/2016/v9i20/89973
29. Couch II LW (2013) Digital and analog communication systems, 8th edn. Prentice Hall
30. Namadchian et al (2016) A new meta-heuristic algorithm for optimization based on variance reduction of Guassian distribution. Majlesi J Electr Eng 10(4):49
31. Freeman RL (1999) Fundamentals of telecommunications. Wiley, New York

32. Freeman RL (2004) Telecommunication system engineering, 4th edn. Wiley, New York
33. Proakis JG (2008) Digital communications, 5th edn. McGraw-Hill
34. Helal M (2008) A hybrid system dynamics-discrete event simulation approach to simulating the manufacturing enterprise. Ph.D. thesis, department of industrial engineering and management systems. College of Engineering and Computer Science, University of Central Florida
35. Packard H (1997) Digital modulation in communications Systems-an introduction. Hewlett-Packard Company, USA
36. Pereira FT, Takano AM, Leal F, Pinho FA (2013) Aplicação Da Simulação A Eventos Discretos Em Um Ambiente Hospitalar Visando A Me-lhoria No Processo De Atendimento. XLVSBPO, Natal, RN—Brasil
37. Pissinelli JG, Risso LA, Picanco SRA, Ignacio ASP, Silva LA (2015) Modelo De Simulação De Eventos Discretos Para Análise De Fluxo De Veículos. ENEGEP, Fortaleza, CE—Brasil
38. Rangel JJA, Costa JVS, Laurindo QMG, Peixoto TA, Matias IO (2016) Análise do fluxo de operações em um servidor de e-mail através de simulação a eventos discretos com o software livre Ururau. Produto & Produção, vol. 17, n. 1, pp. 1–12, Mar 2016
39. Gomes, E.N., Fernandes, M.S.R., Campos, C.A.V., Viana, A.C. "Um Mecanis-mo de Remoção de Mensagens Obsoletas para as Redes Tolerantes a Atrasos e Interrupções. CSBC, 2012.
40. Godoy EP, Lopes WC, Sousa RV, Porto AJV (2010) Modelagem E Simu-lação De Redes De Comunicação Baseadas No Protocolo Can - Controller Area Network. Revista SBA: Controle & Automação, vol. 21, no. 4 (2010)
41. Sharda B, Bury JS (2008) A discrete event simulation model for reliability modeling of a chemical plant. Winter simulation conference
42. Hu W, Sarjoughian HS (2005) Discrete-event simulation of network systems using distributed object computing. SPECTS'05
43. Zhao B, Lin F, Wang C, Zhang X, Polis PM, Wang YL (2017) Supervisory control of networked timed discrete event systems and its applications to power distribution networks. IEEE Trans Control Netw Syst **4**(2)
44. Xu X, Wang Z (2008) Networked modeling and simulation based on SimEvents. In: 7th International Conference on System Simulation and Scientific Computing, 2008. IEEE ICSC 2008. Asia Simulation Conference
45. Dagkakis G, Heavey C (2017) A review of open source discrete event simulation software for operations research. J Simul 193–206
46. Loebbecke C, Picot A (2015) Reflections on societal and business model transformation arising from digitization and big data analytics: a research agenda. J Strat Inf Syst 149–157
47. Lv Z, Song H, Basanta-Val P, Steed A, Jo M (2017) Next-generation big data analytics: state of the art, challenges, and future research topics. IEEE Trans Ind Inf 1891–1899
48. Ylijoki O, Porras J (2016) Perspectives to definition of big data: a mapping study and discussion. J Innov Manag 69–91
49. Zomaya AY, Sakr S (2017) Handbook of big data technologies. Springer, Berlin
50. Monino JL, Sedkaoui S (2016) Big data, open data and data development, vol 3. Wiley, New York
51. Oussous A, Benjelloun FZ, Lahcen AA, Belfkih S (2018) Big data technologies: a survey. J King Saud Univ Comput Inf Sci 431–448
52. McGrath G, Brenner P (2017) Serverless computing: design, implementation, and performance. In: 2017 IEEE 37th international conference on distributed computing systems workshops (ICDCSW). IEEE, pp 405–410
53. Wang H, He D, Tang S (2016) Identity-based proxy-oriented data uploading and remote data integrity checking in public cloud. IEEE Trans Inf Forensics Secur 1165–1176
54. Yang C, Huang Q, Li Z, Liu K, Hu F (2017) Big data and cloud computing: innovation opportunities and challenges. Int J Dig Earth, 13–53
55. Datta S, Roy S, Paulo Davim J (2019) Optimization techniques: an overview. Optimization in industry. Springer, Cham, 1–11
56. Cuevas E, Espejo EB, Conde Enríquez A (2019) Introduction to metaheuristics methods. Metaheuristics algorithms in power systems. Springer, Cham, 1–8

57. Bhattacharyya S ed (2018) Hybrid metaheuristics for image analysis. Springer, Berlin
58. Hussain K et al (2018) Metaheuristic research: a comprehensive survey. Artif Intell Rev 1–43
59. Abdel-Basset M, Abdel-Fatah L, Sangaiah AK (2018) Metaheuristic algorithms: a comprehensive review. Computational intelligence for multimedia big data on the cloud with engineering applications. Academic Press, pp 185–231
60. Fernandez SA et al (2018) Metaheuristics in telecommunication systems: network design, routing, and allocation problems. IEEE Syst J 12(4):3948–3957
61. Latorre-Biel J-I et al (2014) Control of discrete event systems by means of discrete optimization and disjunctive colored PNs: application to manufacturing facilities. Abstract Applied Analysis, vol 2014. Hindawi
62. Juan AA et al (2015) A review of simheuristics: extending metaheuristics to deal with stochastic combinatorial optimization problems. Oper Res Perspect 2:62–72
63. Fikar C et al (2016) A discrete-event driven metaheuristic for dynamic home service routing with synchronised trip sharing. Eur J Ind Eng 10(3):323–340
64. Vieira GE et al (2017) Evaluating the robustness of production schedules using discrete-event simulation. IFAC-PapersOnLine 50(1):7953–7958
65. Bamporiki T, Bekker J (2018) Development of a discrete-event, stochastic multi-objective metaheuristic simulation optimisation suite for a commercial software package. South Afr J Ind Eng 29(3):12–25