Application of Binary Slime Mould Optimization Algorithm for solving Unit Commitment Problem

by

Md. Ashaduzzaman Niloy

160021002

Mutasim Fuad Rizvi

160021005

Md. Sayed Hasan Rifat

160021024

An Undergraduate Thesis submitted to *Islamic University of Technology (IUT)* in partial fulfillment for the award of the degree of,

Bachelor of Sceince in Electrical and Electronic Engineering

Department of Electrical and Electronic Engineering (EEE)

Islamic University of Technology (IUT)

Gazipur, Bangladesh

March 10, 2021

CERTIFICATE OF APPROVAL

The thesis titled '**Application of Binary Slime Mould Optimization Algorithm for solving Unit Commitment Problem**' submitted by Md. Ashaduzzaman Niloy (St. Id. 160021002), Mutasim Fuad Rizvi (St. Id. 160021005) and Md. Sayed Hasan Rifat (St. Id. 160021024) of Academic Year 2016-17 has been found as satisfactory and accepted as partial fulfillment of the requirement for the Degree BACHELOR OF SCIENCE IN ELECTRICAL AND ELECTRONIC ENGINEERING on March 6, 2021.

Approved By,

Dr. Ashik Ahmed
Supervisor and Professor,
Electrical and Electronic Engineering Department,
Islamic University of Technology (IUT)
Boardbazar, Gazipur-1704.
Date: March 10, 2021

ACKNOWLEDGEMENTS

First and foremost, We offer gratitude to the Almighty Allah (SWT) for giving us the capability to do this work with good health.

We are grateful to our research supervisor, Professor Dr. Ashik Ahmed Sir for the support and guidance throughout the course of this work. He created a research environment for which we were able to explore many ideas without constraint. We have gained a wealth of knowledge and experience in science and engineering through his direction that is beyond value to our future endeavors.

We would like to thank all the faculty members of the department of EEE, IUT for their inspiration and help.

And last but not the least we are thankful to our family, friends and well-wishers for their support and inspiration. Without them it would never have been possible for us to make it this far.

Thankfully,

Md. Ashaduzzaman Niloy, 160021002

Mutasim Fuad Rizvi, 160021005

Md. Sayed Hasan Rifat, 160021024

Contents

List of Figures

List of Tables

Nomenclature

Unit Commitment Problem Symbols

N	Total number of generating units
i	Thermal unit index (i=1,2,3,...,N)
H	Total number of scheduling hours
t	Operating hour index (t=1,2,3,...,H)
F^i_{cost}	Fuel cost function of i^{th} unit
a^i,b^i,c^i	Fuel cost coefficients of i^{th} unit
d^i,e^i	Valve-point loading effect coefficients of i^{th} unit
P^i_G	Power generated by i^{th} unit
δ^i_t	Generating status bit of i^{th} unit at t^{th} hour
${SU_{cost}}^i_t$	Startup cost of i^{th} unit at t^{th} hour
$SU^i_{cost\_hot}$	Hot startup cost of i^{th} unit
$SU^i_{cost\_cold}$	Cold startup cost of i^{th} unit
T^i_{mu}	Minimum up time of i^{th} unit
T^i_{md}	Minimum up time of i^{th} unit
${T_{on}}^i_t$	Consecutive hours of committed state of i^{th} unit going into t^{th} hour
${T_{off}}^i_t$	Consecutive hours of de-committed state of i^{th} unit going into t^{th} hour
T^i_{hot}	Hot start hours of i^{th} unit
T^i_{cold}	Cold start hours of i^{th} unit
$P^i_{G\_max}$	Maximum power generated by i^{th} unit
$P^i_{G\_min}$	Minimum power generated by i^{th} unit
P_{Dt}	Load demand at t^{th} hour
SR_t	Spinning reserve required at t^{th} hour

Binary Slime Mould Algorithm Symbols

k	Current iteration number
max_k	Maximum number of iterations
$\vec{X}$	Slime mould location at current iteration
$\vec{vb}$,$\vec{vc}$	Decision making variables of slime mould
$\vec{X}_b$	Best slime mould location found from previous iterations
$\vec{W}$	Weight of the slime mould
$\vec{X}_A$,$\vec{X}_B$	Randomly selected slime mould positions for current iteration
S	Fitness value at current location $\vec{X}$
DF	Best fitness among all the iterations
bF,wF	Best and worst fitness of the ongoing iteration routine respectively
$\vec{X}^*$	Updated Slime mould location for current iteration
UB,LB	Upper and lower bounds of slime mould search area respectively
S_f	Sigmoid transformation function
NP	Overall population of slime mould
s	Slime mould index (s=1,2,3,...,NP)
z	Balance parameter for Exploration and Exploitation

Acronyms

BBSA Binary Bat Search Algorithm.
BFWA Binary Fireworks Algorithm.
BGWA Binary Grey Wolf Optimization Algorithm.
BSMA Binary Slime Mould Algorithm.

DE Differential Evolution.
DP Dynamic Programming.

ELD Economic Load Dispatch.
EP Evolutionary Programming.
ESA Enhanced Simulated Annealing.

GA Genetic Algorithm.
GWO Grey Wolf Optimization.

HGICA Hybrid Genetic-Imperialist Competitive Algorithm.
HHSRSA Hybrid Harmony Search/Random Search Algorithm.

LR Lagrangian Relaxation.

MFA Moth-Flame Algorithm.

PSO Particle Swarm Optimization.

QBPSO Quantum-Inspired Binary Particle Swarm Optimization.
QEA Quantum-inspired Evolutionary Algorithm.
QIBGWO Quantum-Inspired Binary Grey Wolf Optimization.

RCGA Ring Crossover Genetic Algorithm.

SMA Slime Mould Algorithm.
SSA Salp Swarm Algorithm.

UCP Unit Commitment Problem.

ABSTRACT

The Unit Commitment Problem (UCP) is a complex engineering optimization problem of electrical power generation domain. Determining the scheduling for economic consumption of production assets over a specific period of time is the premier objective of UCP. This paper presents a take on solving UCP with Binary Slime Mould Algorithm (BSMA) optimizer. SMA is a recently developed nature-inspired stochastic optimization technique that imitates the selective vegetative growth of slime mould while foraging. A binarized SMA with constraint handling through heuristic adjustment is proposed and implemented to unit commitment problem to generate optimal scheduling for available power resources. Implementing modern heuristic techniques ensures an efficient solution to this non-linear, non-convex and complex constraint driven optimization problem for any number of generating units with maximum profit. To test BSMA as a UCP optimizer, IEEE standard power generating systems ranging from 10 to 100 units along with IEEE 118-bus system are used and the results are then compared with existing classical, evolutionary and hybridized approaches. The comparison reveals superiority of BSMA over all the classical and evolutionary approaches and most of the hybridized methods that are considered in this paper in terms of total cost and convergence characteristics.

Keywords: Binary Slime Mould Algorithm (BSMA), Heuristic optimization algorithm, Unit Commitment Problem (UCP), Economic Load Dispatch (ELD), Power system optimization.

Chapter 1

Introduction

1.1 The Scheduling Problem in the Power System

Power demands vary according to activities of various consumers. This variation in demand results in the fact that the load on a power plant is never constant, it varies time to time. The inherent load fluctuations demanded by users cause most of the complexities of modern power plant operations. Unfortunately, electricity cannot be stored, and the power plant must therefore produce power to meet consumers requirements. Usually, the average demand on the power station is higher during the afternoon and early evening, and lower during the late evening and early morning.

Over a predetermined planning period, we've to determine the power output of each generating unit at each hour along with the startup and shutdown times. The electrical utilities in the regulated markets and the Independent System Operators in the deregulated markets must decide in advance which units to commit and connect to the grid such that the needs of the end-use consumers are satisfied. Turning ON the enough generators throughout the day can be a solution for meeting the required demand but it is not feasible because committing enough units and leaving them online is not economical. Moreover, turning ON or committing a unit is not so simple because various economic and physical factors come into play when taking this decision which are both equally relevant and often contradictory.

Since a functional electrical power system may consist of thousands of generating units, the issue of scheduling becomes a major problem in the system's operation and control that must be solved by the resources given by methods of mathematical optimization. The estimation process for addressing the scheduling issue is referred to as the day-ahead unit commitment process (UC). The Unit Commitment (UC) procedure is applied to assess the generation unit ON/OFF schedule in such a way that the system is most economically satisfied with its projected load demand. In addition, inherent physical constraints on the generating units and other special system conditions must be taken into consideration in this program. The objective of UCP is to reduce the cost of electrical energy generation by fulfilling various types of constraints regarding system and production units.

The UC problem is mathematically classified as a large scale, nonlinear and mixed integer optimization problem with highly constrained non-convex characteristics. The binary existence of the on/off decision is creating non-convexity. The UCP's non-linearity is due to the input-output heat fuel curves and non-linear transmission constraints of each unit [1]. The presence of a mixture of binary and non-linear variables allows the problem of combinatorial optimization to be formulated as a mixed-integer problem. Also, such problem may have multiple feasible regions and multiple locally optimal points within each region. These characteristics make the UC problem difficult to solve. Thus, the Researchers focused on finding an effective and efficient approach UC algorithms that are near-optimal and can be used for large power scale power systems. Two decisive steps are involved in the solution of the UC problem from a methodological point of view. The UC choice incorporates the assurance of the producing units to synchronize and running at every hour of the planed time-frame, considering the units limitations, the beginning up and closure, and the framework limit necessities, including holds. The "economic dispatch" choice incorporates the designation of the framework interest and the turning hold limit among the working units during every particular hour of activity [67].

1.2 Literature Review

Conventional power generation and supply systems at present deal with dynamic load demand with numerous generating units. The load demand keeps fluctuating as the time progresses from day to night, thus, the generating units must be turned on and off correspondingly. It is essential to decide which units are to be turned on and off. These decisions are aggregately called unit scheduling. A better scheduling leads to cost-effective generation, larger profit and efficient usage of electrical generation components, besides satisfying the load demand. The process of determining the scheduling of multiple generating units is Unit Commitment. The Unit Commitment Problem (UCP) is the process of finding the optimal economic consumption of production assets over a specific period of time. The objective of UCP is to reduce the cost of electrical energy generation by fulfilling various types of constraints regarding system and production units. UCP is a non-linear, non-convex, large scale mixed-integer optimization problem having complex constraint specifications.

Over the last few decades, numerous classical, evolutionary heuristic/metaheuristic and hybridized optimization techniques are designed, developed and applied by researchers to solve UCP. Classical deterministic optimization techniques include Priority list (PR) [64], Dynamic programming (DP) [65], Lagrangian relaxation (LR) [54], Benders decomposition (BD) [53], Mixed integer linear programming (MILP) [70], Second order cone programming [78], Semi-definite programming (SDP) [23]. Some other renowned methods of this class are: Branch and bound approaches (BBM) [16], Enhanced adaptive lagrangian relaxation [54], etc. The different advantages of classical deterministic approaches can be enlisted as simple, strong, straightforward characterization and approach, iteration-less swift convergence and integer type output. The downsides are substandard solving caliber (PL), difficulties with system extent (Dynamic and linear programming), complication in calculation, rapid increase of calculation time with larger systems (BBM) and costlier solution.

Several stochastic search algorithms have been developed imitating natural occurrences and preying behaviors of animals. These algorithms were later implemented to solve UCP. Particle swarm optimization by [36] and Grey wolf optimization by [49] are certainly the most practiced approaches with numerous development and hybridization in the UCP domain in this decade. PSO imitates the movement of a flock or a shoal and each bird or fish is considered a particle in the population. On the other hand, GWO replicates the hunting process of grey-wolf pack termed as α, β, δ and ω with hierarchical significance. Both the approaches, depending upon the feedback of the particles or wolves, corrects their position accordingly until they find the global best position. Some of the nature-inspired metaheuristic and evolutionary techniques applied in UCP are Genetic algorithm (GA) [35], Binary particle swarm optimization (BPSO) [25], Ant colony optimization (ACO) [69], Binary grey wolf optimizer (BGWO) [56], Imperialistic competition algorithm (ICA) [20], Enhanced PSO (EPSO) [77], Shuffled leaping frog algorithm (SFLA) [18], Enhanced simulated annealing (ESA) [61], Binary gravitational search algorithm (BGSA) [75], Evolutionary programming (EP) [29], Binary bat search [52], etc. Repair techniques or penalty functions are applied alongside these optimization approaches in order to find the feasible solution. However, drawbacks of these algorithms are longer simulation time, restriction of variables, uncertainty of convergence, excessive iterations and lastly exploration limitations. Different nature-inspired metaheuristic approaches to solve UCP are discussed in [47].

Hybridized methods incorporate both classical optimization techniques and evolutionary metaheuristic algorithms in order to solve UCP. Well known hybridized techniques are Hybrid PSO-GWO [30], Lagrangian relaxation and genetic algorithm (LRGA) [14], LRPSO [5], A solution modification process [39], Hybrid genetic ICA [60], Hybrid harmony random search [32], Improved pre-prepared power demand table and muller method [11], Local convergence average BPSO [72]. Quantum-inspired hybridization is another popular attempt to hybridize classical and evolutionary techniques. QEA [21], QIBGSA [27], QBPSO [26], QIBGWO [63] are some examples of that. Although hybrid methods attain solving credits of both classical and heuristic approaches, they still have some lackings. Most common drawback of hybridized methods is complex sequential procedure, hence longer iteration time and slower convergence. For example, convergence of hybrid PSO-GWO is much slower than NPSO due to sequential computation of PSO and GWO [30]. Some of these metaheuristic and hybridized methods are quite good in terms of solution quality. But, search for more efficient, cost-effective, modern and adoptive optimization approach is still going on as the power generation optimization problem domain has broadened in recent years with the inclusion of similar complex optimization problems like UCP. Dynamic Economic Dispatch (DED) adds non-linearity to the economic dispatch of UCP, which offers more accurate cost minimization based on actual physical and operational needs [80]. Combined Heat and Power Economic Dispatch (CHPED) is a more complicated form of economic dispatch where Combined Heat and Power (CHP) units and heat-only units are considered for cost minimization alongside conventional thermal units [81].

Newer optimization algorithms like Salp Swarm Algorithm (SSA), Harris Hawk Optimization (HHO) and SMA have shown competitive performance compared to the existing algorithms like GWO and WOA [41] for other engineering design problems [48],[22],[46]. So the potential of these algorithms as a UCP optimizer should be explored.

Li et al. [46] proposed a newly developed nature inspired stochastic optimization technique simulating the behavior and morphological changes of acellular slime mould *Physarum polycephalum*. *P. polycephalum* can find the shortest path to a maze, solve complex puzzles, and most importantly, can make multi-objective foraging decisions [7], which resemble with UCP. Slime mould optimization algorithm involves continuous updating of slime mould position to locate the optimal food source, which in case of UCP, translated as optimal cost. However, continuous-natured decision making process of original SMA should be transformed into binary decision making, as on/off scheduling of UCP is represented with two binary values. So in this paper, sigmoid transformation [37] is used to binarize the continuous-natured SMA. Heuristic adjustment of [21] is used for handling the UCP constraints. Then the consequent BSMA optimizer is tested for solving UCP.

The rest of the thesis is organized as follows. Unit commitment problem formulation, its constraint and boundaries are described in Chapter 2, Slime Mould Algorithm (SMA) and its working phases are discussed in Chapter 3, Application of BSMA to solve UCP is described in Chapter 4, Performance tests for different test systems, comparisons, convergence and deviation characteristics are shown in Chapter 5. Lastly, the paper is concluded with a brief summary and future prospects of this topic in Chapter 6.

Chapter 2

Problem Formulation of Unit Commitment

This part is to introduce the numerical models of the constituents of the UC with the end goal that it very well may be defined as a streamlining issue, from presently alluded to as Thermal Unit Commitment Problem (TUCP) and be broke down from a reasonable viewpoint. At that point, since each choice in the UC spins around what to do (or not to do) with a bunch of thermal generating units (generally), a helpful beginning point for the difficult plan is to present the attributes of such units. Prime objective of Unit Commitment is to find the optimum generating schedule for available units which will lead us to the lowest possible operating cost and therefore maximum profit. The total operating cost of a generating unit is composed of three components, fuel cost, startup cost and shutdown cost.

2.1 Characteristics of Thermal Generating Units

Numerous intriguing issues with regards to the force framework designing field are included in the activity, control, and plan of these three segments (for example coordinated machine plan, security considers, and so on). Albeit a plenty of fascinating points are accessible, the investigation of the efficient activity of the force framework and explicitly that of the age planning measure of the UC is of need. Subsequently, principal to these examinations is the numerical connection between the information and the yield (I-O) of the TGU [74].

The input is defined in terms of fuel cost in \$/h (or heat input in MBTU/h) while the output is expressed as the net electrical power in MW, henceforth denoted as $P(k)$, that is available to the EPS. This I-O relation is commonly called the unit fuel consumption function or the operating cost function. The I-O curve is a smooth, convex, and quadratic approximation of the actual nonlinear mathematical relation between the unit's input and output and is widely used in power system operations studies.

2.2 Fuel Cost

The main idea behind the UC is to determine the ON/OFF status of a set of generating units (predominantly TGU's) such that an optimal generation schedule is obtained. This generation schedule must then satisfy every power system requirement and must consider the intrinsic physical limitations of the operating generating unit. Moreover,the meaning of optimal may also vary depending on the desired objective. The first step in the formulation of the UC as a TUCP is always to establish the objective to accomplish and thus to determine the objective function to optimize. Having defined the I-O characteristic of a TGU in terms of fuel cost and electric power output one possible objective for the TUCP is to minimize the total fuel cost of a set of N TGU's over a predetermined study period T

The fuel costs are calculated using the data of unit heat rate and fuel price information. It is a second order quadratic function of power output, of each generator at each hour determined by Economic Dispatch (ED) [3]. Fuel cost can be expressed as:

$$F^i_{cost} = a^i + b^i P^i_G + c^i {P^i_G}^2 \tag{2.1}$$

where F^i_{cost} is the fuel cost function of i^{th} unit, P^i_G is the power generated by the i^{th} unit and a^i, b^i, c^i are the fuel cost coefficients of i^{th} unit. A sinusoidal term is added with Equation (2.1) due to valve point loading [71] in multi-valve steam turbines [12]. So fuel cost function with valve-point loading effect can be expressed as:

$$F^i_{cost} = a^i + b^i P^i_G + c^i {P^i_G}^2 + |d^i sin(e^i * (P^i_{G\_min} - P^i_G))| \tag{2.2}$$

where d^i and e^i are the valve-point coefficients of i^{th} unit and $P^i_{G\_min}$ is the minimum generation limit of i^{th} unit. Total fuel cost of overall generation can be expressed as:

$$\text{Total Fuel Cost} = \sum_{t=1}^{H} \sum_{i=1}^{N} F^i_{cost} P^i_G * \delta^i_t \tag{2.3}$$

$$\forall\ t \in H;\ i \in N;\ \delta^i_t \in \{0, 1\}$$

where H is the total time period of load demand considered for unit commitment i.e. total scheduling hours, N is the total number of generating units and δ_t^i is the generating status bit of i^{th} unit at t^{th} hour.

These coefficients are determined either experimentally through historical or statistical data of unit efficiency and operation or may also be included in the design data as provided by the unit's manufacturer. Also, notice that the c co-efficient is independent of the TGU power output and thus is equivalent to the fuel cost incurred by operating the TGU with no power output. Furthermore, it can be observed that the output power that can be produced by the TGU is bounded by a minimum output power and a maximum. Evidently, this is due to the physical limitations of the TGU. For instance, the minimum output power limitation is mainly influenced by the regenerative cycle of the steam turbine and the combustion stability of the fuel input into the boiler. On the other hand, the maximum output power is usually equivalent to the design capacity of the turbine, boiler, or generator.

2.3 Startup Cost and Shutdown Cost

To bring a TGU on-line, the temperature and pressure in the boiler B must first build slowly. Thus, a determined amount of time and fuel must first be invested to bring the unit to the required operating temperature state.

Startup cost is required to bring a de-committed thermal generating unit back to committed state. Startup cost is warmth dependent [31], and therefore can vary depending upon the de-committed time period of inactive generating unit. Startup cost can be expressed as:

$$SU_{cost_t}^{\ i} = \begin{cases} SU^i_{cost\_hot}, & \text{for } T^i_{mu} \leq T^i_{off} \leq (T^i_{md} + T^i_{cold}) \\ SU^i_{cost\_cold}, & \text{for } T^i_{off} \geq (T^i_{md} + T^i_{cold}) \end{cases} \tag{2.4}$$

where $SU_{cost_t}^{\ i}$ is the startup cost of i^{th} unit at t^{th} hour, $SU^i_{cost\_hot}$ and $SU^i_{cost\_cold}$ are the hot and cold startup cost of i^{th} unit respectively, T^i_{off} is the consecutive hours of de-committed state of i^{th} unit, T^i_{mu} and T^i_{md} are the minimum up and down time of i^{th} unit, respectively. And lastly, T^i_{cold} is the cold start hours of i^{th} unit.

The value of SC is also influenced by the amount of time the TGU has been off before being brought online. The value of SC can vary from an upper "cold-start" cost to a much lower "hot-start" cost if the unit has been turned off recently and is close to the operating temperature state.

The cold start and hot start costs can then be modeled by considering the exponential decrease in temperature after the TGU (with some thermal time constant γ has been turn OFF such that the best decision on when to turn ON again the TGU is made.

Shutdown costs refers to a fixed amount of cost to maintain a de-committed generating unit and it is independent of the de-committed period. Shutdown costs are neglected in this paper in accordance with the other approaches in [35], [56] and [26]. So, the total cost of overall generation can be represented as:

$$\text{Total Cost} = \sum_{t=1}^{H} \sum_{i=1}^{N} F^i_{cost} P^i_G * \delta^i_t + {SU_{cost}}^i_t * (1 - \delta^i_t)\delta^i_t \tag{2.5}$$

2.4 Generation Regulating Constraints

In practical cases, plant operators face numerous generation regulating constraints. For example, a generating unit has its minimum and maximum generation limits which change over time. Also, we've to consider the time required to bring a de-committed unit online. Three types of constraints are considered in this research as follows: 1) System constraints, 2) Unit constraints and 3) Time-dependent constraints. Implication of such constraints to UCP is stated below:

2.4.1 Maximum and Minimum Generation Limits

The power generation of a particular committed unit should be within it's maximum and minimum generation limits.

$$P^i_{G\_min}\delta^i_t \leq P^i_G\delta^i_t \leq P^i_{G\_max}\delta^i_t \tag{2.6}$$

where $P^i_{G\_min}$ and $P^i_{G\_max}$ are the minimum and maximum generation limit of i^{th} unit respectively.

2.4.2 Balancing Load Demand and Power Supply

As per load demand and power supply balance constraint, the summation of power generation of all committed unit must be greater than or equal to the load demand at time t.

$$\sum_i^N P_{G}{}^i_t * \delta^i_t \geq P_{Dt} \tag{2.7}$$

where P_{Dt} is the load demand at t^{th} hour.

2.4.3 Spinning Reserve

An extra generation capacity should be reserved to continue satisfying load demand in the cases of a sudden excessive load or any kind of generation failure. It makes the system more reliable and immune to failures. This excess reserve capacity is known as spinning reserve. Spinning reserve is added with the load demand and the summation is considered as the generation requirement. So Equation (2.7) can be re-written as:

$$\sum_i^N P_{G}{}^i_t * \delta^i_t \geq P_{Dt} + SR_t \tag{2.8}$$

where SR_t is the spinning reserve required at t^{th} hour.

2.4.4 Minimum Up/Down Time

A generating unit should be brought online after a certain time interval from decommitted status based on its thermal cooling characteristics. Similarly, the unit must run for a certain period of time before it is shut down. These constraints are termed as Minimum Up and Minimum Down Time, and can be expressed as:

$$\delta_t^i = \begin{cases} 1, & \text{for } 1 \leq {T_{on}}_{t-1}^i \leq T_{mu}^i \\ 0, & \text{for } 1 \leq {T_{off}}_{t-1}^i \leq T_{md}^i \end{cases} \tag{2.9}$$

2.4.5 Initial Status

The terminal status of every unit from previous scheduling time period should be assessed to avoid disrupting the minimum up/down time cycle of the generating units.

To summarize the unit commitment problem formulation, objective function of Equation (2.5) is a minimization problem. Equation (2.6), (2.8) and (2.9) are inequality constraints. This constrained minimization problem of unit commitment is going to be solved using BSMA.

Chapter 3

Slime Mould Algorithm (SMA)

SMA is a recently proposed nature based optimization algorithm [46] which has been adopted to solve many engineering optimization problems. The foundation and detailed working procedure of SMA is presented in this chapter.

3.1 Foundation of Slime Mould Algorithm

Eukaryoric Slime mould occupies cold, moist places. The prime healthy stage of a slime mould is Plasmodium, the active and dynamic stage of slime mould, and the fundamental exploration phase of this research. In this stage, the natural matter in slime mould looks for food, encompasses it, and secretes proteins to process it. During the migration process, the front end extends into a fan-shaped, followed by an interconnected venous network that allows cytoplasm to flow inside [38], as shown in Figure 3.1. Due to their novel example and trademark, they can utilize various food sources simultaneously to shape a venous organization interfacing them. If there is enough food in the environment, slime mould can even grow to more than 900 square centimeters [38].

Owing to the feature of slime mould can be easily cultured on agar and oatmeal [10], they were broadly utilized as model organisms. Kamiya and his colleagues [33] were the primary group to consider the definite interaction of the cytoplasmic progression of slime mould. Their work is of extraordinary assistance to our resulting comprehension of the manner in which slime mould move and associates food in the climate. We now cognize that when a vein approaches a food source, the bio-oscillator produces a propagating wave [50] that increases the cytoplasmic flow through the vein, and the faster the cytoplasm flows, the thicker the vein. Through this mix of positive-negative feedback, the slime can build up the ideal way to associate food in a generally prevalent manner. Subsequently, slime mould was additionally numerically demonstrated and applied in diagram hypothesis and path networks.

The venous construction of slime creates alongside the stage distinction of the constriction mode [50], so there are three connections between the morphological changes of the venous design and the withdrawal method of slime mould.

- Thick veins structure generally along the span when the compression frequencies fluctuate from outside to inside.

- At the point when the compression mode is unsteady, anisotropy starts to show up.

- When the contraction pattern of slime mould is no longer ordered with time and space, the venous construction is not, at this point present.

Therefore, the connection between venous design and compression example of slime mould is predictable with the state of normally framed cells. Physarum solver flow feedback determines the thickness of each vein. [8]. The raise in the progression of cytoplasm leads to an increment in the measurement of veins. As the stream diminishes, the veins contract as a result of the reduction of the breadth.

Slime mould can fabricate a more grounded course where food fixation is higher, along these lines guaranteeing that they get the greatest convergence of supplements. Late investigations have likewise uncovered that sludge form have the skill of making scrounging plans dependent on improvement hypothesis [42]. At the point when the nature of different food sources is unique, slime mould can pick the food source with the most elevated fixation. In any case, slime mould additionally needs to gauge speed and danger in scavenging. For example, slime mould needs to settle on quicker choices to keep away from natural harm to them. Tests have shown that the faster the dynamic speed is, the prospects of slime mould to locate the excellent food source is more modest [43]. Consequently, when choosing the wellspring of food, slime mould clearly needs to gauge the speed and exactness.

Slime mould need to choose when to leave this territory and search in another region when scavenging. When lacking total data, the most ideal path for an slime mould to

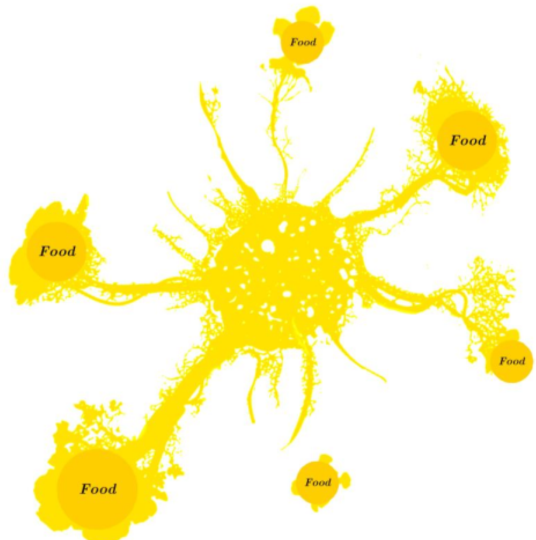

Figure 3.1: Foraging morphology of slime mould.

assess when to leave the current position is to receive heuristic or exact principles dependent on the deficient data presently accessible. Experience has shown that when slime mould experience excellent food, the likelihood of leaving the zone is diminished [44]. Nonetheless, because of its exceptional natural attributes, slime mould can use an assortment of food sources simultaneously. Consequently, regardless of whether the slime mould has discovered a superior wellspring of food, it can in any case isolate a segment of the biomass to misuse the two assets all the while when better food is found [8].

Slime mould can likewise powerfully change their hunt designs as per the nature of staple provenience. At the point when the nature of food sources is high, the slime mould will utilize the locale restricted pursuit strategy [34], along these lines zeroing in the inquiry on the food sources that have been found. On the off chance that the thickness of the food provenience at first discovered is low, the slime mould will leave the food source to investigate other elective food sources in the district. This versatile inquiry procedure can be more reflected when diverse quality food blocks are scattered in a locale. A portion of the systems and qualities of the slime mould referenced above will be numerically demonstrated in the ensuing segments.

3.2 Working Principle of Slime Mould Algorithm

The working principle of SMA mimics the food searching phenomenon of slime mould. Depending upon the odor of the food, slime mould proceeds to migrate through its vegetative growth. It approaches the food with continuous course correction from its cytoplasmic feedback. A slime mould can search for multiple food sources simultaneously, looking for a higher concentration of food. Feedback dependent propagation wave from bio-oscillator controls the cytoplasmic flow, hence the thickness of vein towards a food source. Figure 3.2 of [2] shows an exemplary visualization of the process.

The higher the concentration of food contacted by the vein, the stronger the wave generated by the bio-oscillator, the faster the cytoplasm flows, and the thicker the vein [46]. For choosing the optimal, most concentrated food source, slime mould also controls it's weight distribution in the search area. Upon realizing a lesser food concentration in an area, slime mould shifts its weight to inspect the other regions. The whole food searching procedure of slime mould can be summarized as:

- The odor of food at any position in search area will trigger the propagation wave of bio-oscillator, and hence, control the course of cytoplasmic veins.
- The diameter of the veins will be thicker towards highly concentrated areas and thinner towards the barely concentrated areas.
- The weight of the slime mould will be shifted mostly towards the higher concentrated regions.

3.2.1 Exploration phase

Exploration phase of SMA involves locating the food and approaching towards the location in the search area. This process as in [46], can mathematically be expressed as such:

$$\vec{X}(k+1) = \begin{cases} \vec{X}_b(k) + \vec{vb} * (\vec{W}\vec{X}_A(k) - \vec{X}_B(k)), & \text{for } r < p \\ \vec{vc} * \vec{X}(k), & \text{for } r \geq p \end{cases} \tag{3.1}$$

(a) (b)

Figure 3.2: Slime mould exploits the nutrient source (oat flakes), (a) Exploiting oat flakes by a network of cytoplasmic veins [2], (b) Scheme of the cytoplasmic propagation, circles are nutrient source (oat flakes) and star mark is the primary slime mould position, arrows are cytoplasmic veins [2]

.

$$\forall\ \vec{vb} \in [-a, a];\ \vec{vc} \in [-1, 1];\ r \in \{0, 1\}$$

where k represents the current iteration and r is a random number between $\{0, 1\}$. $\vec{X}(k)$ represents the current slime mould location and $\vec{X}(k+1)$ refers to the next location. $\vec{X}_b(k)$ is the best location with the strongest odor found so far. $\vec{X}_A(k)$ and $\vec{X}_B(k)$ are two randomly selected slime mould positions. $\vec{vb}$ and $\vec{vc}$ oscillates in-between their limits and refers to the decision making of slime mould, whether to reach the food source or search for other higher quality sources (Figure 3.3). Both the variables reaches zero as the number of iterations increases, because by then the slime mould finds it's optimal food source. The limits of $\vec{vb}$ can be expressed as:

$$a = arctanh(-(\frac{k}{max\_k}) + 1) \tag{3.2}$$

where max_k denotes the maximum number of iteration.

p is a function formulated as:

$$p = tanh(|S(v) - DF|) \tag{3.3}$$

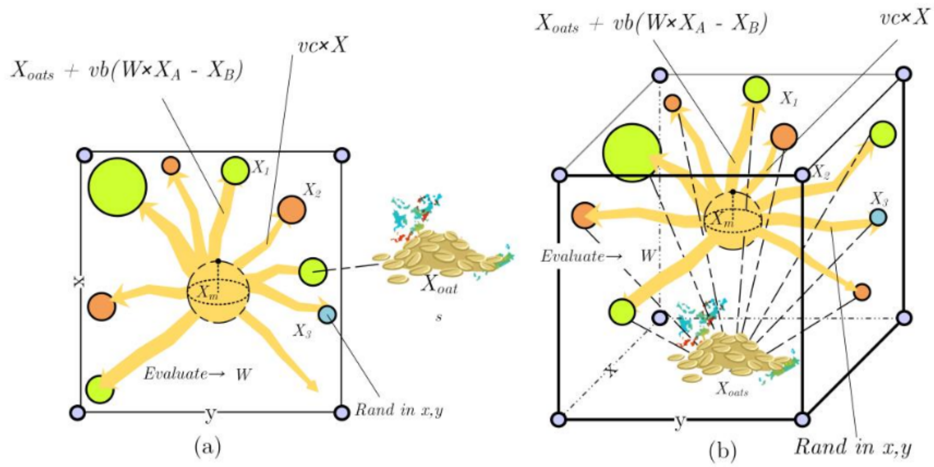

Figure 3.3: Exploration phase of slime mould: Possible positions in 2D and 3D.

$$\forall\ v \in \{1, 2, 3, ..., n\}$$

where v is the number of slime mould veins, $S(v)$ is the current fitness value of $\vec{X}$, and DF is the best fitness among all the iterations. And lastly, $\vec{W}$ represents the frequency of oscillation which determines the thickness of the veins. In simpler terms, $\vec{W}$ is the weight of the slime mould. $\vec{W}$ is mathematically expressed as:

$$\vec{W}(SmellIdx(v)) = \begin{cases} 1 + rlog(\frac{bF - S(v)}{bF - wF} + 1), & \text{for } condition \\ 1 - rlog(\frac{bF - S(v)}{bF - wF} + 1), & \text{for } others \end{cases} \tag{3.4}$$

where *condition* indicates that $S(v)$ ranks first half of the population, bF and wF are the best and worst fitness of the ongoing iteration routine respectively (Figure 3.4), and $SmellIdx(v)$ refers to a sequence of fitness value sorted in ascending order.

Figure 3.3 shows the effects of Equation 3.1. The position of individual slime mould $\vec{X}$ can be updated according to the best location $\vec{X_b}$ currently obtained, and the fine-tuning of parameters $\vec{vb}$, $\vec{vc}$ and $\vec{W}$ can change the position of the individual slime mould. Figure 3.3 is also used to illustrate the location change of the slime mould in 3D space. *rand* in the equation can make people structure search vectors at any point, that is, search arrangement space toward any path, so the calculation has the chance of finding the ideal arrangement. Therefore, Equation 3.1 empowers the looking through

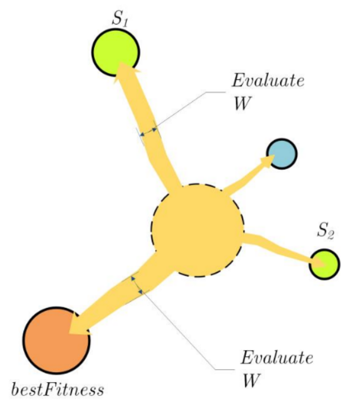

Figure 3.4: Exploration phase of slime mould: Assessment of fitness.

individual to look on the whole potential bearings close to the ideal arrangement, in this manner recreating the round area design of slime mould when moving toward food. It is additionally pertinent to expand this idea to Hyper-dimensional space.

3.2.2 Exploitation phase

In the exploitation phase, SMA optimizer continuously updates the slime mould position depending upon the feedback from the exploration phase. As in [46] this can be expressed as follows:

$$\vec{X^*} = \begin{cases} rand * (UB - LB) + LB, & \text{for } rand < z \\ \vec{X_b}(k) + \vec{vb} * (W * \vec{X_A}(k) - \vec{X_B}(k)), & \text{for } r < p \\ \vec{vc} * \vec{X}(k), & \text{for } r \geq p \end{cases} \tag{3.5}$$

$$\forall\, rand \in [0, 1];\, r \in [0, 1]$$

where LB and UB are the lower and upper bounds of search area respectively. $rand$ and r are random values between 0 and 1, and lastly the value of parameter z is taken as 0.03, as the probability maintains a proper balance between exploration and exploitation at this constant z value [46].

This part mimics the withdrawal method of venous tissue construction of slime mould numerically while looking. The higher the convergence of food reached by the vein, the more grounded the wave created by the bio-oscillator, the quicker the cytoplasm streams, and the thicker the vein. Equation 3.4 numerically mimicked the positive and negative criticism between the vein width of the slime mould and the food fixation that was investigated. The component r in Equation 3.4 recreates the vulnerability of venous constriction mode. log is utilized to ease the change pace of mathematical worth so the estimation of withdrawal recurrence doesn't change excessively. condition reenacts the slime mould to change their inquiry designs as indicated by the nature of food. At the point when the food focus is content, the greater the load close to the locale is; the point at which the food fixation is low, the heaviness of the district will be decreased, hence going to investigate different areas. Figure 3.4 shows the process of evaluating fitness values for slime mould.

Slime mould mostly relies upon the spread wave created by the natural oscillator to change the cytoplasmic stream in veins, so they will in general be in a superior situation of food fixation. On the motivation behind recreating the varieties of venous width of slime mould, we used $\vec{W}$, $\vec{vb}$ and $\vec{vc}$ to realize the variations. $\vec{W}$ numerically recreates the wavering recurrence of slime mould almost one at various food focus, so that slime mould can move toward food all the more immediately when they discover great food, while approach food all the more gradually when the food fixation is lower in individual position, hence improving the proficiency of slime mould in picking the ideal food source. The value of $\vec{vb}$ wavers arbitrarily among $[-a, a]$ and slowly moves toward zero as the addition of cycles. The value of $\vec{vc}$ oscillates between $[-1, 1]$ and tends to zero anyway.

Synergistic interaction between $\vec{vb}$ and $\vec{vc}$ mimics the selective behavior of slime mould. To locate a superior wellspring of food, regardless of whether slime mould has discovered a superior wellspring of food, it will in any case isolate some natural matter for investigating different regions trying to locate a more excellent wellspring of food, instead of putting every last bit of it in one source.

Chapter 4

Formulation of Binary Slime Mould Algorithm to solve UCP

In this section, SMA binarization method, priority list of units, solving Economic Load Dispatch (ELD) with lambda iteration method and constraint handling processes are discussed. Then the use of BSMA to solve constraint minimization problem of unit commitment is described in detail with necessary diagrams.

4.1 Population Structure and Binary Mapping

An illustration of BSMA population structure is shown in Figure 4.1. In the figure, the commitment status of i^{th} unit of s^{th} slime mould at k^{th} iteration and t^{th} hour is represented as δ_{ts}^{ik}. However, SMA itself is non-discrete in nature, meaning a particular slime mould of the population can be assigned continuous values. Sigmoid transform inspired from BPSO [37] is used to limit δ_{ts}^{ik} to only binary values. Sigmoid transform equation is expressed as:

$$S_f(X_s^k) = \frac{1}{1 + e^{-X_s^k}} \tag{4.1}$$

$$X_s^{k+1} = \begin{cases} 1, & \text{for } S_f(X_s^k) > r \\ 0, & \text{otherwise} \end{cases} \tag{4.2}$$

where S_f denotes the transfer function, X_s^k is the position for s^{th} slime mould at k^{th} iteration and finally r is a random number in [0,1].

Value "1" is assigned to δ_{ts}^{ik} if i^{th} unit is committed at t^{th} hour, and vice versa for de-committed unit. An N by H matrix is assigned to each δ_s^k where N is the maximum number of units, and $i \in \{1, 2, 3, ..., N\}$ and H represents the total time horizon, and $t \in \{1, 2, 3, ..., H\}$. NP denotes the overall population of slime mould and

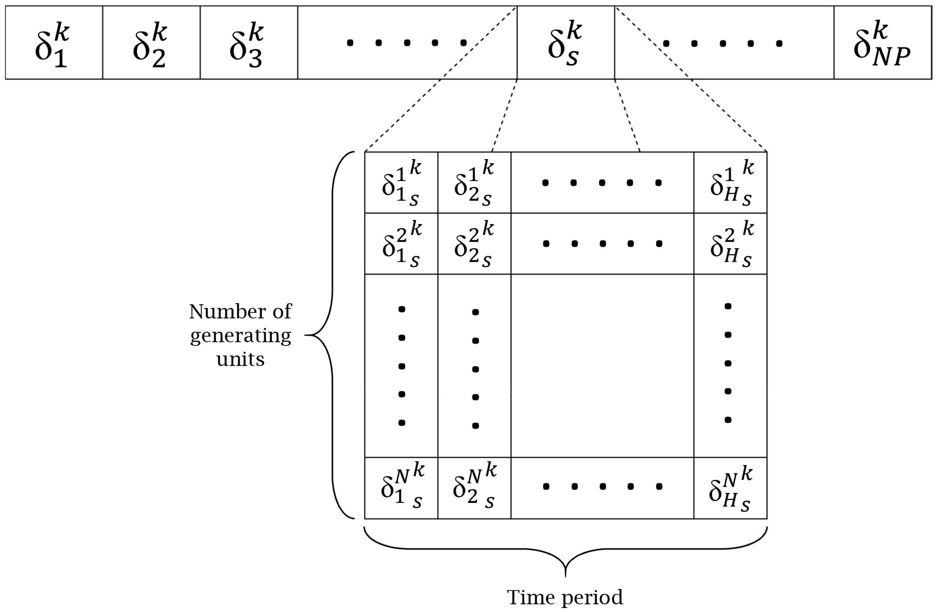

Figure 4.1: Population structure of unit commitment with BSMA.

$s \in \{1, 2, 3, ..., NP\}$.

4.2 Priority List of generating units

Not all the units have the same running cost as the cost parameters of a generator change at a great extent over its lifetime. A priority list is formed according to the objective function of fuel cost in Equation (2.1). Fuel cost, F^i_{cost} of i^{th} unit depends greatly on it's fuel co-efficients a^i, b^i and c^i. The lesser the value of F^i_{cost} of a generating unit, the higher it is placed in the priority list, and consequently, kept committed for a longer period of time.

4.3 Economic Load Dispatch using Lambda Iteration method

The purpose of Economic load dispatch (ELD) is to distribute generation demand within the available units so that the total generation and operation cost is minimized and

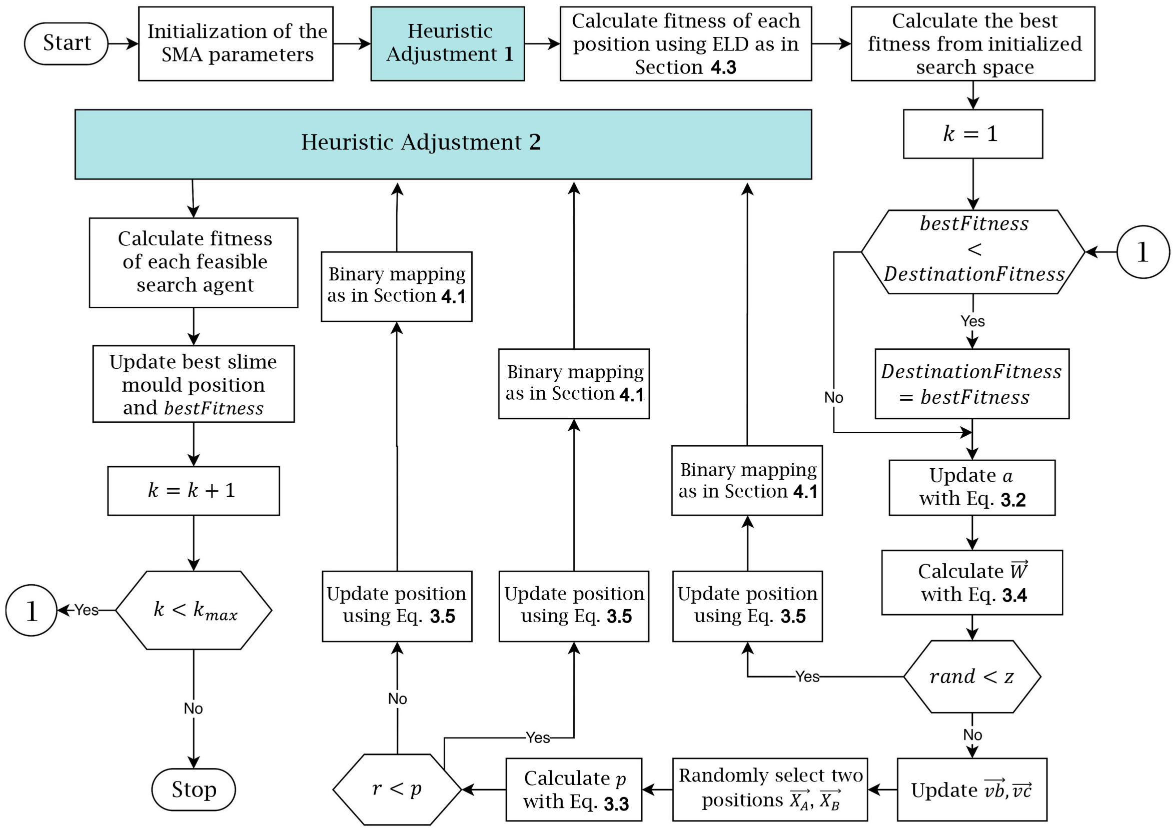

Figure 4.2: Flowchart of UCP with BSMA.

none of the system operating constraints is violated. After determining a feasible unit schedule, the generation schedule is obtained on tolerance basis. The margin specifies the difference between the generation and load demand to a specified limited value. After finding the optimal value, conventional cost calculation takes place. ELD using a piecewise quadratic cost functions cannot be easily solved by conventional numerical methods [66]. Therefore, an enhanced lambda iteration method [62] is used to execute economic load dispatch.

4.4 Constraint handling and repairment

In order to eliminate the infeasible solutions from the search space, a heuristic approach of [21] is adopted in this paper. Handling constraints like minimum up/down time

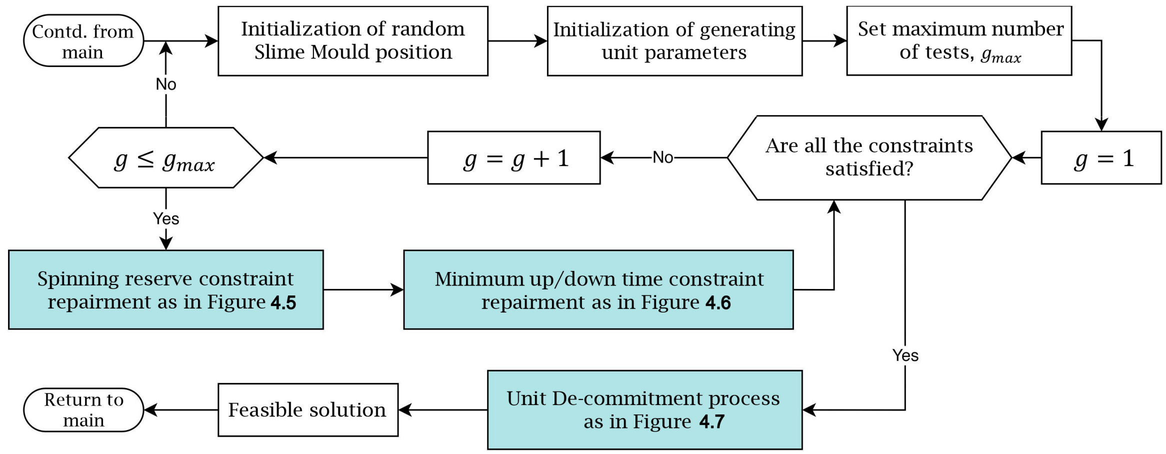

Figure 4.3: Flowchart of Heuristic adjustment 1.

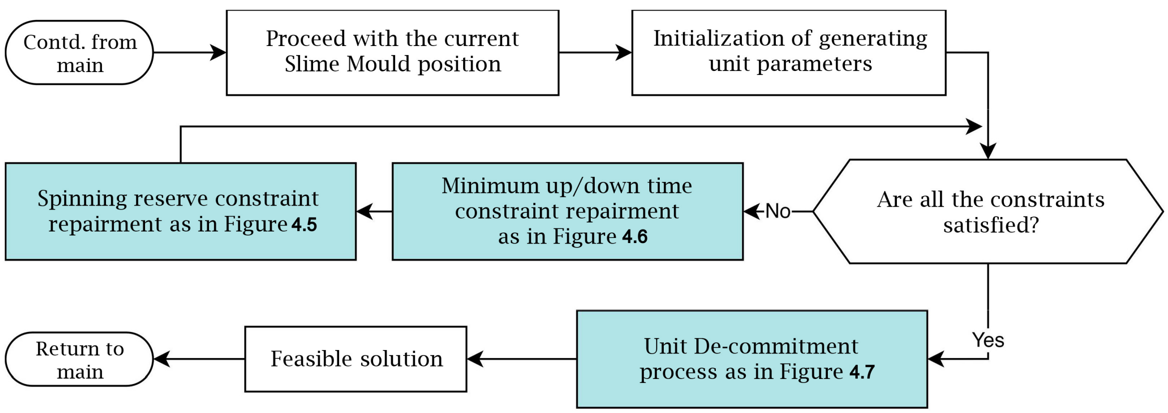

Figure 4.4: Flowchart of Heuristic adjustment 2.

improves the solution quality and reduces the possibility of failure significantly. On the other hand, extra reserve capacity and unnecessary committed units can increase the running cost by a wide margin. Therefore, the constraint handling heuristic approach is used for each and every slime mould of the population.

4.4.1 Spinning reserve constraint repairment

Spinning reserve constraint is to be satisfied for system reliability. The solution in not feasible as long as the spinning reserve constraint is violated, so the required number of units are made committed until the requisite spinning reserve capacity along with

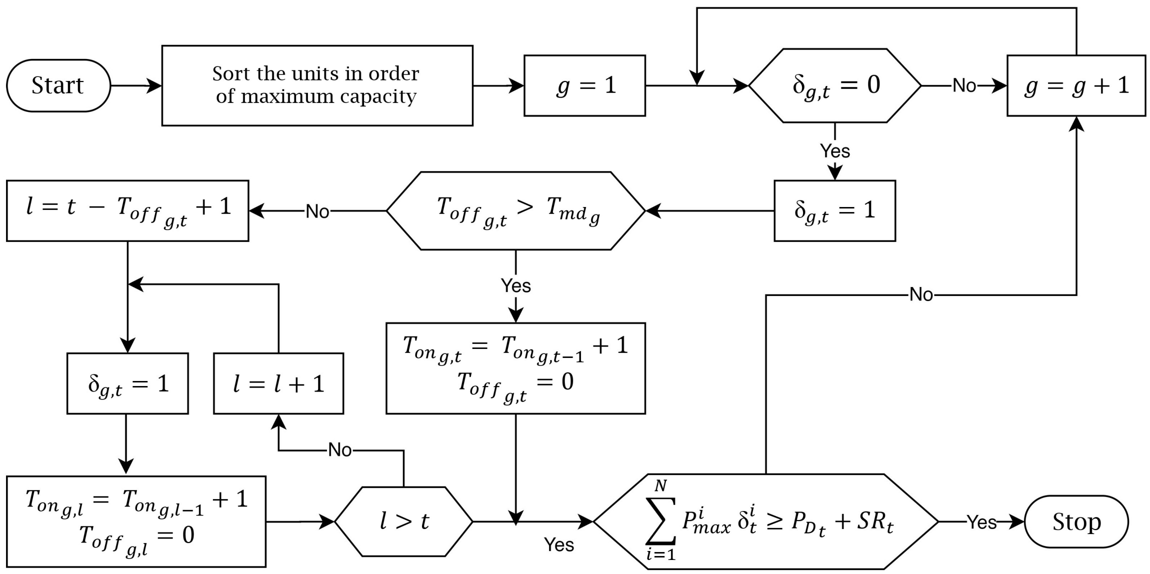

Figure 4.5: Flowchart of spinning reserve constraint handling for constraint repairment.

the load demand is satisfied. The spinning reserve constraint repairment is depicted in detail in Figure 4.5.

4.4.2 Minimum up/down time constraint repairment

All the units must follow the minimum up/down time as a prerequisite before being committed or de-committed. The heuristic adjustment process for any minimum up/down time constraint violation is shown in Figure 4.6.

4.4.3 Unit de-commitment process

While satisfying the spinning reserve constraint and minimum up/down time constraint, some extra thermal units might be committed, resulting in an unnecessary increase in operational cost. De-commitment process described in Figure 4.7 is applied to de-commit those inessential units.

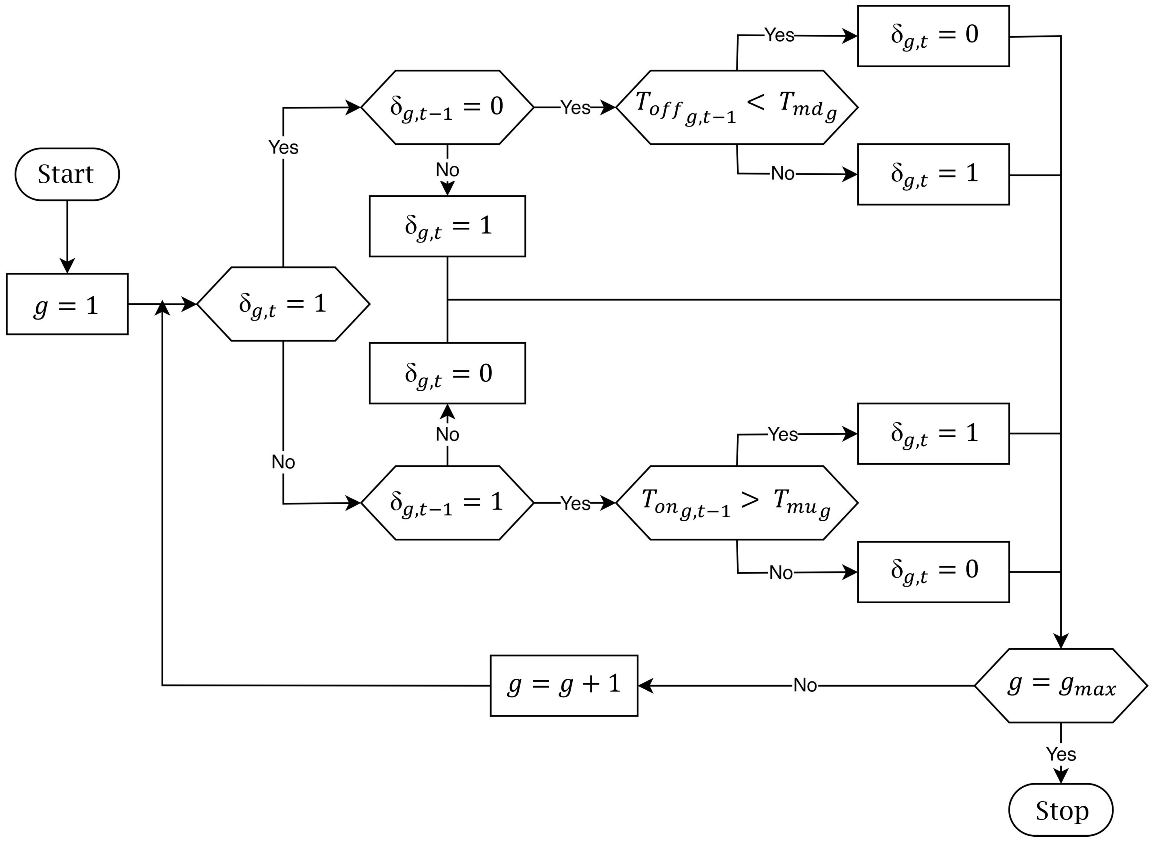

Figure 4.6: Flowchart of minimum up time, minimum down time constraint repairment.

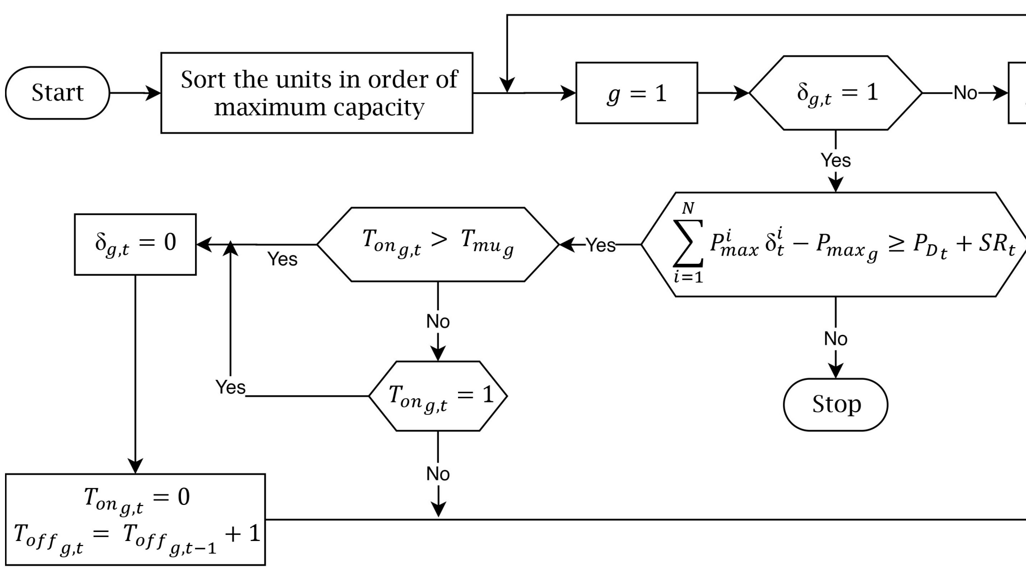

Figure 4.7: Flowchart of de-commitment process to avoid excess spinning reserve.

4.5 Fundamental steps of UCP with BSMA

A detailed illustration of UCP with BSMA is represented as flowcharts in Figure 4.2, 4.3 and 4.4. The fundamental steps are described below:

Step 1. Initialize the search agent population according to Section 4.1.

Step 2. Formation of priority list according to Section 4.2.

Step 3. Adjust unit status of each search agent to satisfy spinning reserve constraint according to Section 4.4.1 and Figure 4.5.

Step 4. Modify the search agents to meet the minimum up/down time constraint as in Section 4.4.2 and Figure 4.6.

Step 5. De-commit the redundant units following the procedure in Section 4.4.3 and Figure 4.7.

Step 6. Solve Economic Load Dispatch (ELD) with lambda iteration method as in Section 4.3

Step 7. Initialization of the BSMA parameters.

Step 8. Calculate the fitness value of each feasible search agent with the UCP objective function Equation (2.5).

Step 9. Determine the $bestFitness$ among the search agents by comparing individual fitness values.

Step 10. Evaluate weight of the slime mould with the Equation (3.4).

Step 11. Update the values of p, $\vec{vb}$ and $\vec{vc}$ for each search agents.

Step 12. Reform the slime mould positions with Equation (3.5).

Step 13. Perform sigmoid transform for updated positions found in Step 12. according to Equations (4.1) and (4.2).

Step 14. If current iteration is less than the maximum number of iterations, then go to Step 3. Otherwise continue.

Step 15. Obtain the values of the search agent with $bestFitness$ as the optimal solution.

Chapter 5

Performance Assessment of BSMA

In order to test its effectiveness in solving UCP, BSMA is modeled for three different test cases. The first test case incorporates standard benchmark setups ranging from 10 to 100 units and the cost characteristics are considered quadratic in nature. The second test case is a single set of 10 unit system with valve-point loading effect included in the cost function. And the third test case is a standard IEEE 118-bus 54-unit system. A MATLAB 2016b environment with INTEL core i3, 4gb RAM and Windows operating system is used to simulate the performance tests. The outcomes are then put in comparison with other established methods to demonstrate the efficacy of BSMA on solving unit commitment problem.

The population of simulations is determined after studying the effect it had on the results. Effect of population size on optimal cost and execution time is depicted in Figure 5.1, for 40 unit system and 100 iterations. 40 unit system is considered as a trade-off unit value between small, medium and large scale systems [56]. The study suggests, population greater than 25 produces similar results with greater execution time, hence, 25 is selected as slime mould population size for performance testing.

5.1 Case-I: 10-100 unit systems without valve-point loading effect

These benchmark unit systems are further classified into three categories, Small Scale, Medium Scale and Large Scale Systems. In all three cases, 10% spinning reserve is considered and transmission losses are neglected.

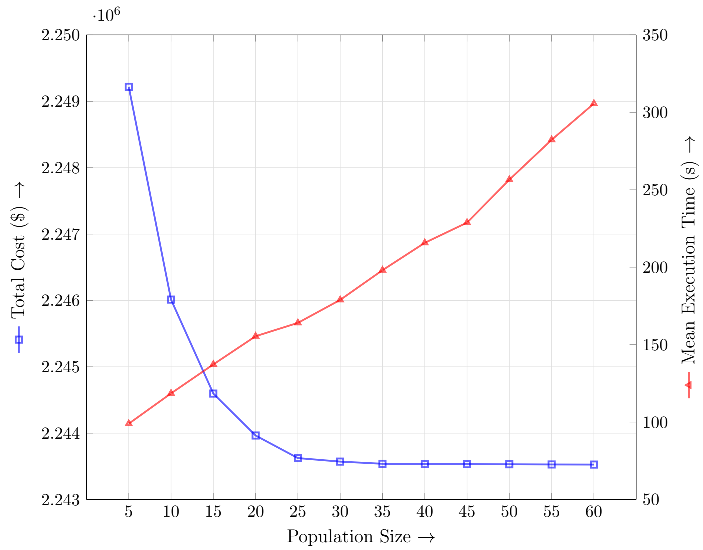

Figure 5.1: Variation of total cost and mean execution time with population size (for 40 unit system, 100 iterations).

5.1.1 Small scale test systems

Two types of test systems, 10 and 20 unit systems are studied as small scale generation systems for performance testing. 24 hour load demand data used in 10-unit system is shown in Table 5.1 and illustrated in Figure 5.2. Generating unit data used on this occasion are shown in Table 5.2. The commitment schedule and demand distribution found after simulation is shown in Table 5.3. From the scheduling in Table 5.3, the generation cost is obtained $559866.23 and startup cost is found $4070. Performance comparison with 20 other renowned approaches is shown in Table 5.4 for 50 trials. The table shows the average cost of BSMA is on par with hybridized approach QIBGWO [63], both with $563936, better than Binary grey wolf algorithm [56], Hybrid harmony search/random search algorithm [32], Hybrid genetic imperialistic competitive algorithm [60] and all the other classical, evolutionary and hybridized methods. The best cost found for BSMA is $563662.12, which is significantly better than any other approach. Convergence and deviation characteristics for 1000 iterations and 50 trials are shown in Figure 5.5a and Figure 5.6a respectively. Standard deviation, σ is 0.0293% for 10-unit system, which is significantly better than other pure, non-hybridized metaheuristic UCP optimization approaches. Mean execution time is shown in Figure 5.4.

Unit characteristics of 10-unit system are replicated and load demand is doubled for

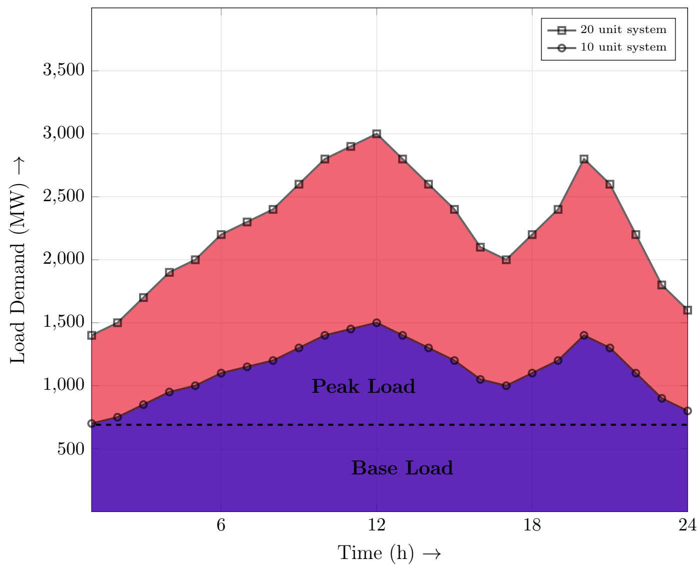

Figure 5.2: Test load demand curve for 10 and 20 unit systems.

Table 5.1: Test load demand data for 10 generating unit system

Time (h)	1	2	3	4	5	6	7	8	9	10	11	12
Load (MW)	700	750	850	950	1000	1100	1150	1200	1300	1400	1450	1500
Time (h)	13	14	15	16	17	18	19	20	21	22	23	24
Load (MW)	1400	1300	1200	1050	1000	1100	1200	1400	1300	1100	900	800

each instance as shown in Figure 5.2 for 20-unit system, in accordance with the other renowned approaches in [25], [56] and [63]. The generation scheduling and demand distribution for a random trial is shown in Table 5.5. From Table 5.5, the total generation cost is found as \$1115010.63 and the startup cost for this load demand is obtained as \$8690. Standard deviation, σ is 0.0283%. Convergence and deviation attributes for 1000 iterations and 50 trials are illustrated in Figure 5.5b and 5.6b respectively. Mean execution time is shown in Figure 5.4. Table 5.4 reveals the performance comparison of BSMA with other approaches. In this case, the best cost recorded for 50 trials is \$1123204.99, and it is marginally better than QIBGWO [63], Ring crossover genetic algorithm [9] and Quantum-inspired binary PSO [26]. Average cost \$1123700.63 is however, better than all the other mentioned approaches except QIBGWO [63].

Table 5.2: Test data for 10 generating unit system.

Unit No.	$P^i_{G\_max}$	$P^i_{G\_min}$	a^i	b^i	c^i	T^i_{mu}	T^i_{md}	$SU^i_{cost\_hot}$	$SU^i_{cost\_cold}$	T^i_{cold}	Initial Status
U1	455	150	1000	16.19	0.00048	8	8	4500	9000	5	8
U2	455	150	970	17.26	0.00031	8	8	5000	10000	5	8
U3	130	20	700	16.6	0.002	5	5	550	1100	4	-5
U4	130	20	680	16.5	0.00211	5	5	560	1120	4	-5
U5	162	25	450	19.7	0.00398	6	6	900	1800	4	-6
U6	80	20	370	22.26	0.00712	3	3	170	340	2	-3
U7	85	25	480	27.74	0.00079	3	3	260	520	2	-3
U8	55	10	660	25.92	0.00413	1	1	30	60	0	-1
U9	55	10	665	27.27	0.00222	1	1	30	60	0	-1
U10	55	10	670	27.79	0.00173	1	1	30	60	0	-1

5.1.2 Medium scale test systems

Similar to the approach taken for 20-unit system, 10-unit system and demand data are replicated to produce inputs for all the medium and large scale UCP test systems. 40 and 60-unit systems are considered as medium scale test systems. Performance of BSMA against 40-unit system is considered for showing the variation of cost and mean execution time over number of population in Figure 5.1, as a trade-off between small, medium and large scale systems. Also, the convergence comparison with QEA [21], BGWO [56] and QIBGWO [63] is demonstrated in Figure 5.3, which reveals a competitive convergence with QIBGWO and quicker convergence than the other two. Cost comparison shown in Table 5.6 reveals the superiority of BSMA for both 40 and 60-unit system over most of the approaches. Best, average and worst in case of convergences are shown in Figure 5.5c and 5.5d for 40 and 60-unit system respectively. Deviation characteristics are shown in Figure 5.6c and 5.6d with standard deviation σ. Declining value of σ suggests BSMA is more effective for larger systems. And lastly, mean

Table 5.3: Commitment and Generation schedule for 10 generating unit system trial. (with 10% spinning reserve)

Time (h)	Commitment Schedule for 10 generating unit system										Generation Schedule for 10 generating unit system									
	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10
1	1	1	0	0	0	0	0	0	0	0	455	245	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	455	295	0	0	0	0	0	0	0	0
3	1	1	1	0	0	0	0	0	0	0	455	265	130	0	0	0	0	0	0	0
4	1	1	1	1	0	0	0	0	0	0	455	235	130	130	0	0	0	0	0	0
5	1	1	1	1	0	0	0	0	0	0	455	285	130	130	0	0	0	0	0	0
6	1	1	1	1	1	0	0	0	0	0	455	360	130	130	25	0	0	0	0	0
7	1	1	1	1	1	0	0	0	0	0	455	410	130	130	25	0	0	0	0	0
8	1	1	1	1	1	0	0	0	0	0	455	455	130	130	30	0	0	0	0	0
9	1	1	1	1	1	1	0	1	0	0	455	455	130	130	100	20	0	10	0	0
10	1	1	1	1	1	1	1	0	0	1	455	455	130	130	162	33	25	0	0	10
11	1	1	1	1	1	1	1	1	0	1	455	455	130	130	162	73	25	10	0	10
12	1	1	1	1	1	1	1	1	1	1	455	455	130	130	162	80	25	43	10	10
13	1	1	1	1	1	1	1	0	0	1	455	455	130	130	162	33	25	0	0	10
14	1	1	1	1	1	1	0	0	1	0	455	455	130	130	100	20	0	0	10	0
15	1	1	1	1	1	0	0	0	0	0	455	455	130	130	30	0	0	0	0	0
16	1	1	1	1	1	0	0	0	0	0	455	310	130	130	25	0	0	0	0	0
17	1	1	1	1	1	0	0	0	0	0	455	260	130	130	25	0	0	0	0	0
18	1	1	1	1	1	0	0	0	0	0	455	360	130	130	25	0	0	0	0	0
19	1	1	1	1	1	0	0	0	0	0	455	455	130	130	30	0	0	0	0	0
20	1	1	1	1	1	1	1	1	0	0	455	455	130	130	162	33	25	10	0	0
21	1	1	1	1	1	1	1	0	0	0	455	455	130	130	85	20	25	0	0	0
22	1	1	0	0	1	1	1	0	0	0	455	455	0	0	145	20	25	0	0	0
23	1	1	0	0	1	0	0	0	0	0	455	420	0	0	25	0	0	0	0	0
24	1	1	0	0	0	0	0	0	0	0	455	345	0	0	0	0	0	0	0	0

execution time for both the units is shown in Figure 5.4.

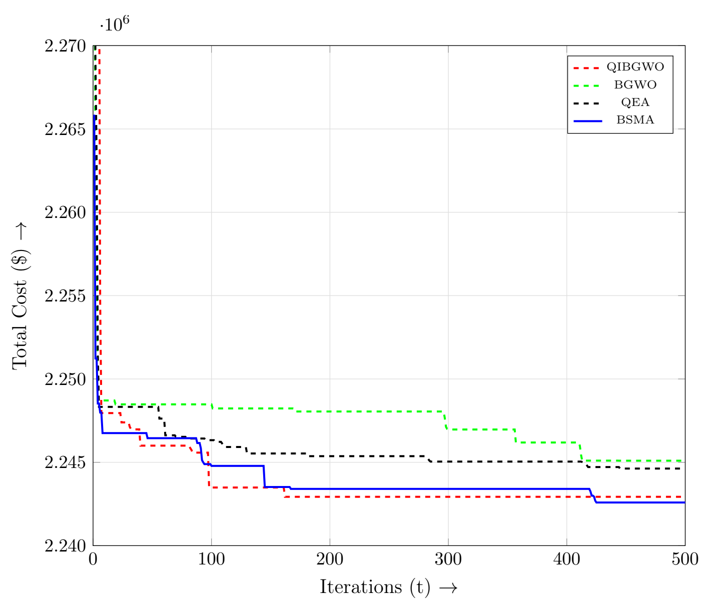

Figure 5.3: Comparison of convergence characteristics with recent approaches for 40 unit system.

Table 5.4: Comparison of results for 10 and 20 unit systems. (with 10% spinning reserve)

	10 unit system			20 unit system		
Approach	Best Cost ($)	Average Cost ($)	Worst Cost ($)	Best Cost ($)	Average Cost ($)	Worst Cost ($)
GA [35]	563977	-	566606	1126243	-	1132059
EP [29]	564551	565352	566231	1126494	1127257	1129793
QEA [21]	563938	563969	564672	1123607	1124689	1125715
LR [54]	566107	566493	566817	1128362	1128395	1128444
ESA [61]	565828	565988	566260	1126254	1127955	1129112
PSO [79]	564212	565783	565103	1125983	-	1131054
IPSO [79]	563954	564579	564162	1125279	-	1127643
BDE [24]	563997	563997	563997	1126998	1127374	1127927
BPSO [25]	563977	563977	563977	1128192	1128213	1128403
IBPSO [76]	563977	564155	565312	1196029	-	-
QBPSO [26]	563977	563977	563977	1123297	1123981	1124294
LCA-PSO [72]	570006	-	-	1139005	-	-
IQEA [15]	563977	563977	563977	1123890	1124320	1124504
BFWA [55]	563977	564018	564855	1124658	1124941	1125087
HGICA [60]	563935.31	563937	563938	1124565	1124933	1125147
HHSRSA [32]	563937.6	563965.31	563995.33	1124889	1124912.8	1124951.5
RCGA [9]	563937	564019	564219	1123297	1123851	1124537
QIBGWO [63]	563936.3	563936.3	563936.3	1123294	1123459	1123526
BBSA [52]	563937.3	564568.85	565205.72	1124720.76	1125598.56	1126985.16
BGWA [56]	563976	564378	565518	1125546	1126126	1127393
Proposed BSMA	563662.12	563936.66	564254.05	1123204.99	1123700.629	1124294.43

5.1.3 Large scale test systems

80 and 100-unit systems are studied as large scale systems. In these occasions, standard deviation is found to be significantly less compared to smaller systems and BSMA converges in lesser iterations. Comparison of total cost with other renowned approaches is shown in Table 5.7. For the 80-unit test system, SMA has shown the most optimal cost of $4483601.9, where QIBGWA [63] being the second best and RCGA [9] next on the list. Similarly for 100-unit system, BSMA is leading with $5602705.28, better than all the other approaches including QBPSO [26]. Table 5.6 and 5.7 indicate that the BSMA proposed in this paper has shown better performance for the larger test systems

Table 5.5: Commitment and Generation schedule for 20 generating unit system trial. (with 10% spinning reserve)

Time (h)	Commitment Schedule (U1-U10)										Generation Schedule (U1-U10)									
	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10
1	1	1	0	1	1	0	1	1	0	1	455	150	0	120	25	0	25	10	0	10
2	1	1	0	1	1	0	1	1	1	0	455	195	0	130	25	0	25	10	10	0
3	1	1	0	1	1	0	1	0	1	1	455	295	0	130	25	0	25	0	10	10
4	1	1	0	1	1	0	0	0	0	0	455	417.5	0	130	25	0	0	0	0	0
5	1	1	0	1	1	0	0	1	0	1	455	455	0	130	30	0	0	10	0	10
6	1	1	0	1	1	0	0	0	0	1	455	432.5	0	130	25	0	0	0	0	10
7	1	1	0	1	1	1	0	1	0	0	455	455	0	130	35	20	0	10	0	0
8	1	1	0	1	1	1	0	1	0	0	455	455	0	130	125	20	0	10	0	0
9	1	1	0	1	1	1	0	1	1	1	455	455	0	130	157.5	20	0	10	10	10
10	1	1	1	1	1	1	1	1	0	1	455	455	130	130	162	33	25	10	0	10
11	1	1	1	1	1	1	1	1	1	0	455	455	130	130	162	73	25	10	10	0
12	1	1	1	1	1	1	1	1	1	1	455	455	130	130	162	80	25	43	10	10
13	1	1	1	1	1	1	0	1	1	1	455	455	130	130	162	43	0	10	10	10
14	1	1	1	1	1	0	0	1	0	1	455	455	130	130	110	0	0	10	0	10
15	1	1	1	0	1	0	0	1	1	1	455	455	130	0	135	0	0	10	10	10
16	1	1	1	0	1	0	0	0	0	1	455	435	130	0	25	0	0	0	0	10
17	1	1	1	0	1	1	1	0	0	1	455	362.5	130	0	25	20	25	0	0	10
18	1	1	1	0	1	1	1	0	0	0	455	455	130	0	37.5	20	25	0	0	0
19	1	1	1	0	1	1	1	0	1	0	455	455	130	0	122.5	20	25	0	10	0
20	1	1	1	1	1	1	1	0	1	1	455	455	130	130	162	33	25	0	10	10
21	1	1	1	1	1	1	0	1	1	1	455	455	130	130	82.5	20	0	10	10	10
22	1	1	0	1	1	1	0	0	0	0	455	397.5	0	130	25	20	0	0	0	0
23	1	1	0	1	0	1	0	1	0	0	455	287.5	0	130	0	20	0	10	0	0
24	1	1	0	1	0	1	1	0	0	1	455	375	0	130	0	20	25	0	0	10

Time (h)	Commitment Schedule (U11-U20)										Generation Schedule (U11-U20)									
	U11	U12	U13	U14	U15	U16	U17	U18	U19	U20	U11	U12	U13	U14	U15	U16	U17	U18	U19	U20
1	1	1	0	0	0	0	0	0	0	0	455	150	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	455	195	0	0	0	0	0	0	0	0
3	1	1	0	0	0	0	0	0	0	0	455	295	0	0	0	0	0	0	0	0
4	1	1	0	0	0	0	0	0	0	0	455	417.5	0	0	0	0	0	0	0	0
5	1	1	0	0	0	0	0	0	0	0	455	455	0	0	0	0	0	0	0	0
6	1	1	1	1	0	0	0	0	0	0	455	432.5	130	130	0	0	0	0	0	0
7	1	1	1	1	0	0	1	0	0	0	455	455	130	130	0	0	25	0	0	0
8	1	1	1	1	0	0	1	1	0	0	455	455	130	130	0	0	25	10	0	0
9	1	1	1	1	1	0	1	0	0	0	455	455	130	130	157.5	0	25	0	0	0
10	1	1	1	1	1	1	1	0	0	0	455	455	130	130	162	33	25	0	0	0
11	1	1	1	1	1	1	1	1	1	0	455	455	130	130	162	73	25	10	10	0
12	1	1	1	1	1	1	1	1	1	1	455	455	130	130	162	80	25	43	10	10
13	1	1	1	1	1	1	0	0	1	1	455	455	130	130	162	43	0	0	10	10
14	1	1	1	1	1	0	0	1	1	0	455	455	130	130	110	0	0	10	10	0
15	1	1	1	0	1	0	0	1	1	0	455	455	130	0	135	0	0	10	10	0
16	1	1	1	0	1	0	0	0	0	0	455	435	130	0	25	0	0	0	0	0
17	1	1	1	0	1	0	0	0	0	0	455	362.5	130	0	25	0	0	0	0	0
18	1	1	1	0	1	0	0	0	0	0	455	455	130	0	37.5	0	0	0	0	0
19	1	1	1	0	1	1	0	0	0	0	455	455	130	0	122.5	20	0	0	0	0
20	1	1	1	1	1	1	1	0	0	0	455	455	130	130	162	33	25	0	0	0
21	1	1	1	1	1	1	1	0	0	0	455	455	130	130	82.5	20	25	0	0	0
22	1	1	1	1	1	0	1	1	0	0	455	397.5	130	130	25	0	25	10	0	0
23	1	1	0	1	1	0	0	0	0	0	455	287.5	0	130	25	0	0	0	0	0
24	1	0	0	1	0	0	0	0	0	0	455	0	0	130	0	0	0	0	0	0

than the smaller ones. The convergence and deviation characteristics of these two test systems are shown in Figure 5.5e, 5.5f, 5.6e and 5.6f respectively. Figure 5.4 shows the mean execution time of 80 and 100-unit systems.

Table 5.6: Comparison of results for 40 and 60 unit systems. (with 10% spinning reserve)

	40 unit system			60 unit system		
Approach	Best Cost ($)	Average Cost ($)	Worst Cost ($)	Best Cost ($)	Average Cost ($)	Worst Cost ($)
GA [35]	2252909	-	2269282	3376625	-	3384252
EP [29]	2249093	2252612	2256085	3371611	3376255	3381012
QEA [21]	2245557	2246728	2248296	3366676	3368220	3372007
LR [54]	2258503	2258503	2258503	3394066	3394066	3394066
ESA [61]	2250012	2252125	2254539	-	-	-
PSO [79]	2250012	-	2257146	3374174	-	3382921
IPSO [79]	2248163	-	2252117	3370979	-	3379125
BDE [24]	2245700	2246600	2247284	3367066	3367405	3367783
BPSO [25]	2243210	2244634	2245982	3363649	3365301	3367171
IBPSO [76]	2243728	-	-	3367865	3368278	3368779
QBPSO [26]	2242957	2244657	2245941	3361980	3367550	3367755
LCA-PSO [72]	2277396	-	-	3420438	-	-
IQEA [15]	2245151	2246026	2246701	3365003	3365667	3366223
BFWA [55]	2248228	2248572	2248645	3367445	3367828	3367974
HGICA [60]	2239186	2242612	2246085	-	-	-
HHSRSA [32]	2248508	2248652.7	2248757	-	-	-
RCGA [9]	2242887	2243569	2244117	3365337	3366052	3366873
QIBGWO [63]	2242947	2244071	2244279	3361766	3364280	3364873
BBSA [52]	2248259.85	2251980	2266356.1	3367650.89	3369425.55	3389428.94
BGWA [56]	2252475	2257866	2263333	3367276	3367550	3367755
Proposed BSMA	2242303.6	2243301.7	2244683.4	3363491.58	3364887.915	3366097.72

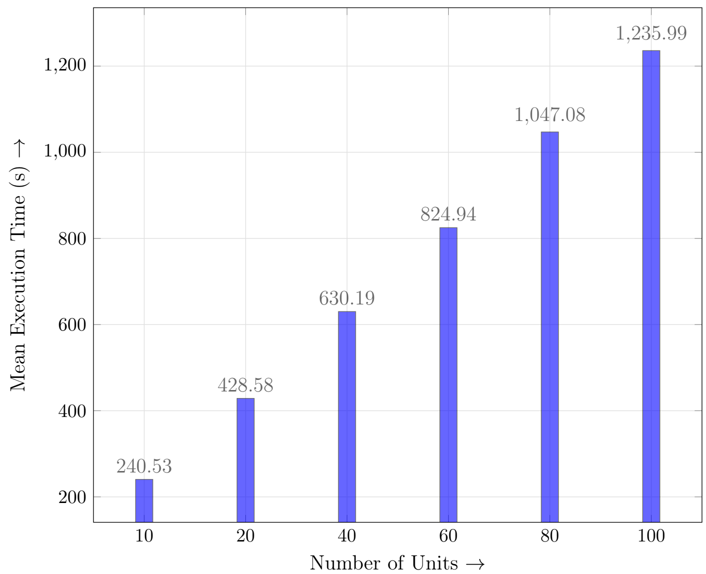

Figure 5.4: Mean execution time of test systems for BSMA.

Table 5.7: Comparison of results for 80 and 100 unit systems. (with 10% spinning reserve)

	80 unit system			100 unit system		
Approach	Best Cost ($)	Average Cost ($)	Worst Cost ($)	Best Cost ($)	Average Cost ($)	Worst Cost ($)
GA [35]	4507692	-	4552982	5626361	-	5690086
EP [29]	4498479	4505536	4512739	5626885	5633800	5639148
QEA [21]	4488470	4490128	4492839	5609550	5611797	5613220
LR [54]	4526022	4526022	4526022	5657277	5657277	5657277
ESA [61]	4498076	4501156	4503987	5617876	5624301	5628506
PSO [79]	4501538	-	4513725	5625376	-	5641378
IPSO [79]	4495032	-	4508943	5619284	-	5633021
BDE [24]	4489022	4490456	4491262	5609341	5609984	5610608
BPSO [25]	4491083	4491681	4492686	5610293	5611181	5612265
IBPSO [76]	4488351	-	-	5608792	-	-
QBPSO [26]	4482085	4485410	4487168	5602486	5604275	5606178
LCA-PSO [72]	4554346	-	-	5706201	-	-
IQEA [15]	4486963	4487985	4489286	5606022	5607561	5608525
BFWA [55]	4491284	4492550	4493036	5610954	5612422	5612790
HGICA [60]	4485936	4487958	4489283	5604022	5608561	5613260
RCGA [9]	4486991	4487476	4487949	5606663	5607088	5607850
QIBGWO [63]	4481925	4486761	4487935	5602365	5605773	5606974
BBSA [52]	4491248.13	4492746.8	4492278.5	5611769	5612450.6	5613759.7
BGWA [56]	4495173	4506362	4513026	5628975	5637699	5643899
Proposed BSMA	4482619.33	4483601.9	4485743.1	5601253.43	5602705.284	5604741.04

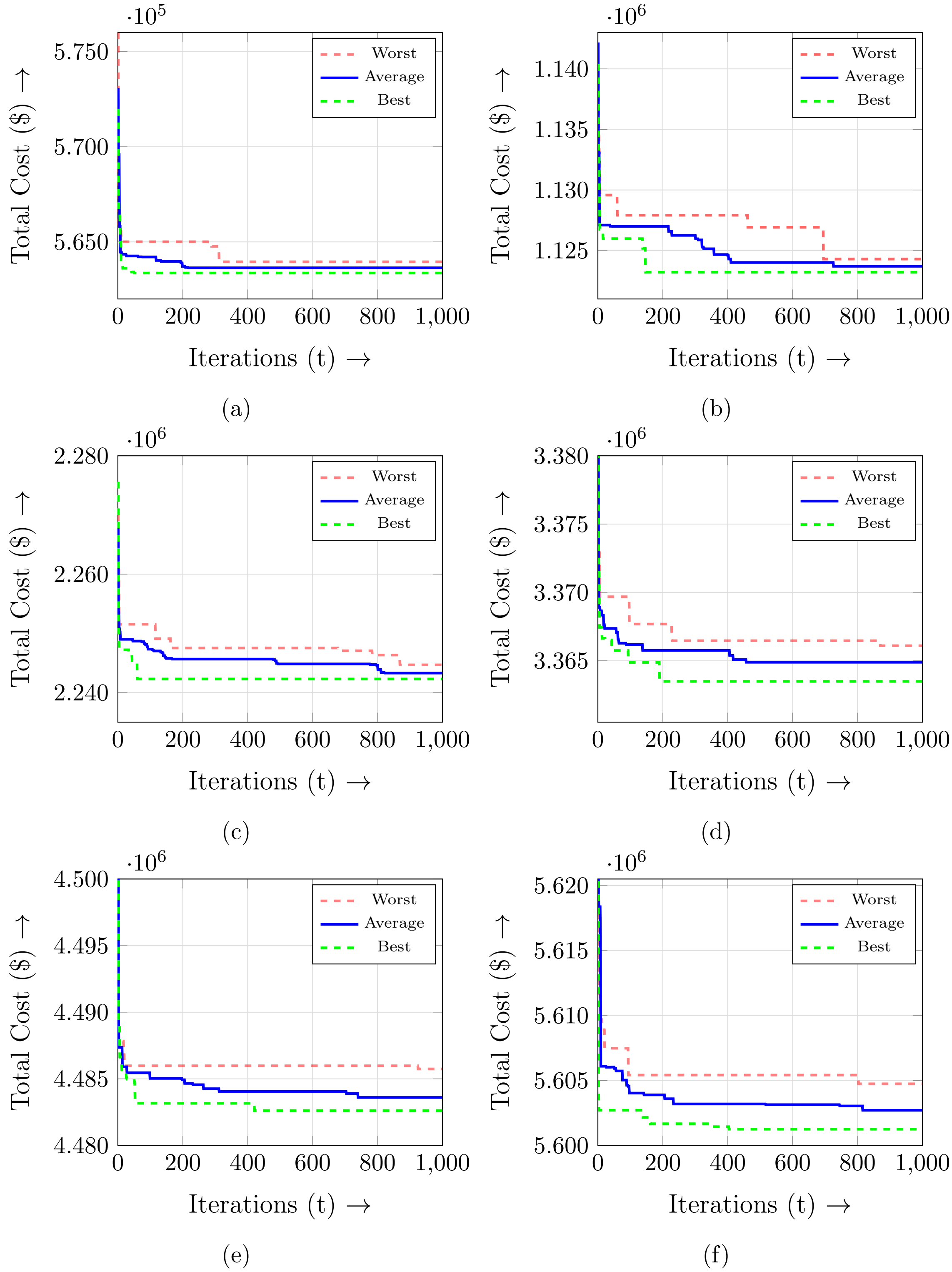

Figure 5.5: Convergence curves of different test systems for BSMA: (a) 10 units, (b) 20 units, (c) 40 units, (d) 60 units, (e) 80 units, (f) 100 units.

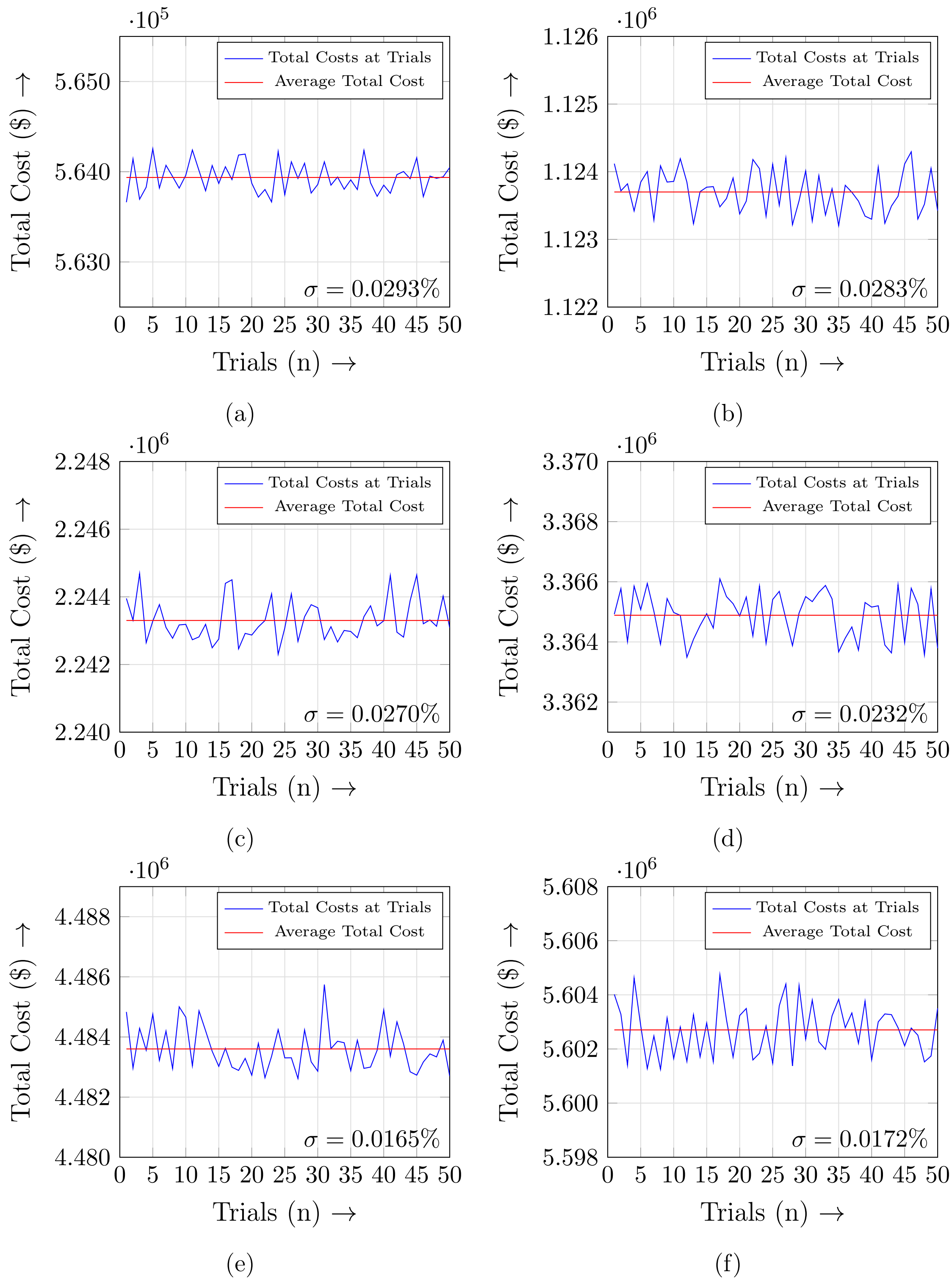

Figure 5.6: Deviation of results of BSMA on independent trials: (a) 10 units, (b) 20 units, (c) 40 units, (d) 60 units, (e) 80 units, (f) 100 units.

5.2 Case-II: 10-unit system with valve-point loading effect

In this second test case, a single set of 10-unit system is considered including the sinusoidal term of valve-point loading effect in fuel cost function. Generating unit data and Load demand are obtained from [6]. Spinning reserve is taken as 6% of the load demand and transmission losses are neglected. Generation schedule is shown in Table 5.8. As in Table 5.9, average total generating cost of BSMA is better than NSGA-II [6], BRABC [12] and TLBO [59]. Only QTLBO [59] has shown better cost than Proposed BSMA in this test case.

5.3 Case-III: IEEE 118-bus test system

IEEE 118-bus test system consisting of 54 generating units is also considered for BSMA performance test. Quadratic cost function is used in this test case. Spinning reserve is taken as 10% of load demand and transmission losses are neglected. Comparison is shown in Table 5.10 with SDP [4], BRABC [12], BRCFF [13] and FFA with multiple workers [40]. The comparison shows the superiority of BSMA, with the best operation cost for IEEE 118-bus system over the other 4 approaches. Load demand and generating unit data used for this case are shown in 5.11 and 5.12.

Table 5.8: Generation Schedule for 10-unit system with valve-point loading effect

Time (h)	Generation Schedule										F_{cost} (\$/h)	SU_{cost} (\$/h)
	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10		
1	456.5	395.5	184.0	0	0	0	0	0	0	0	25181.2	550
2	379.9	395.5	334.5	0	0	0	0	0	0	0	26912.0	0
3	379.9	395.5	308.5	0	174.1	0	0	0	0	0	30457.9	900
4	456.5	308.2	297.1	120.4	223.9	0	0	0	0	0	34379.0	560
5	456.5	395.5	296.6	107.6	223.9	0	0	0	0	0	36100.5	0
6	456.5	395.5	310.8	241.3	223.9	0	0	0	0	0	39580.5	0
7	456.5	395.5	326.1	300.0	223.9	0	0	0	0	0	41461.7	0
8	456.5	395.5	298.8	241.4	222.6	159.9	0	0	0	0	43000.3	340
9	456.5	395.7	296.2	299.7	222.5	122.4	130.9	0	0	0	46092.1	520
10	456.5	395.5	321.1	299.8	222.8	160.0	131.0	85.4	0	0	50444.9	60
11	456.6	395.7	307.5	299.9	223.4	159.9	131.0	120.0	52.0	0	52793.3	60
12	456.5	458.6	300.7	299.9	222.6	160.0	130.9	85.3	52.0	53.4	55354.8	60
13	456.5	395.4	321.6	299.9	222.6	159.7	131.0	85.3	0	0	50446.3	0
14	456.5	395.5	296.6	299.5	222.6	122.4	130.9	0	0	0	46086.8	0
15	456.5	395.6	298.8	241.3	224.0	159.9	0	0	0	0	43000.1	0
16	456.5	395.5	297.3	180.8	224.0	0	0	0	0	0	37714.4	0
17	456.5	395.5	284.7	119.4	224.0	0	0	0	0	0	36092.6	0
18	456.5	395.5	302.1	120.4	222.6	0	130.9	0	0	0	39077.9	260
19	456.5	395.5	315.5	180.8	172.8	123.9	130.9	0	0	0	42687.3	170
20	456.5	395.5	339.8	299.8	222.5	159.9	130.9	47.0	20.1	0	50904.3	120
21	456.7	458.6	340.0	185.3	222.7	129.9	130.9	0	0	0	46628.0	0
22	456.6	395.5	325.0	299.8	0	0	130.9	0	20.0	0	39671.6	30
23	456.5	395.5	302.1	0	0	0	130.9	0	47.0	0	32131.4	0
24	456.5	395.5	331.9	0	0	0	0	0	0	0	28525.3	0

Table 5.9: Comparison of results for 10-unit system with valve-point loading effect

Approach	Average Cost ($)
Binary/Real Coded Artificial Bee Colony [12]	982949.55
Teaching-learning Based Algorithm [59]	978426.82
Quasi-oppositional Teaching Learning Based Algorithm [59]	976827.17
Proposed BSMA	978354.24

Table 5.10: Comparison of results for 54-unit 118-bus system

Approach	Average Cost ($)
Semi-definite Programming [4]	1645444.98
Binary/Real Coded Artificial Bee Colony [12]	1644269.71
Binary/Real Coded Firefly Algorithm [13]	1644141
Firefly Algorithm with Multiple Workers [40]	1644134
Proposed BSMA	1644122.68

Table 5.11: Test load demand data for 54-unit system 118-bus system

Time (h)	1	2	3	4	5	6	7	8	9	10	11	12
Load (MW)	4200	3960	3480	2400	3000	3600	4200	4680	4920	5280	5340	5040
Time (h)	13	14	15	16	17	18	19	20	21	22	23	24
Load (MW)	4800	4560	5280	5400	5100	5340	5640	5880	6000	5400	5220	4920

Table 5.12: Test data for 54-unit 118-bus system.

Unit No.	$P^i_{G\_max}$	$P^i_{G\_min}$	a^i	b^i	c^i	T^i_{mu}	T^i_{md}	$SU^i_{cost\_hot}$	$SU^i_{cost\_cold}$	T^i_{cold}	Initial Status
U1	30	5	31.67	26.24	0.0697	1	1	40	120	2	1
U2	30	5	31.67	26.24	0.0697	1	1	40	120	2	1
U3	30	5	31.67	26.24	0.0697	1	1	40	120	2	1
U4	300	150	6.78	12.89	0.0109	8	8	440	1320	7	8
U5	300	100	6.78	12.89	0.0109	8	8	110	330	7	8
U6	30	10	31.67	26.24	0.0697	1	1	40	120	2	1
U7	100	25	10.15	17.82	0.0128	5	5	50	150	4	5
U8	30	5	31.67	26.24	0.0697	1	1	40	120	2	1
U9	30	5	31.67	26.24	0.0697	1	1	40	120	2	1
U10	300	100	6.78	12.89	0.0109	8	8	100	300	7	8
U11	350	100	32.96	10.76	0.003	8	8	100	300	7	8
U12	30	8	31.67	26.24	0.0697	1	1	40	120	2	1
U13	30	8	31.67	26.24	0.0697	1	1	40	120	2	1
U14	100	25	10.15	17.82	0.0128	5	5	50	150	4	5
U15	30	8	31.67	26.24	0.0697	1	1	40	120	2	1
U16	100	25	10.15	17.82	0.0128	5	5	50	150	4	5
U17	30	8	31.67	26.24	0.0697	1	1	40	120	2	1
U18	30	8	31.67	26.24	0.0697	1	1	40	120	2	1
U19	100	25	10.15	17.82	0.0128	5	5	59	177	4	5
U20	250	50	28	12.33	0.0024	8	8	100	300	7	8
U21	250	50	28	12.33	0.0024	8	8	100	300	7	8
U22	100	25	10.15	17.82	0.0128	5	5	50	150	4	5
U23	100	25	10.15	17.82	0.0128	5	5	50	150	4	5
U24	200	50	39	13.29	0.0044	8	8	100	300	7	10
U25	200	50	39	13.29	0.0044	8	8	100	300	7	10
U26	100	25	10.15	17.82	0.0128	5	5	50	150	4	5
U27	420	100	64.16	8.34	0.0106	10	10	250	750	8	10
U28	420	100	64.16	8.34	0.0106	10	10	250	750	8	10
U29	300	80	6.78	12.89	0.0109	8	8	100	300	7	10
U30	80	30	74.33	15.47	0.0459	4	4	45	135	4	4
U31	30	10	31.67	26.24	0.0697	1	1	40	120	2	1
U32	30	5	31.67	26.24	0.0697	1	1	40	120	2	1
U33	20	5	17.95	37.70	0.0283	1	1	30	90	2	1
U34	100	25	10.15	17.82	0.0128	5	5	50	150	4	5
U35	100	25	10.15	17.82	0.0128	5	5	50	150	4	5
U36	300	150	6.78	12.89	0.0109	8	8	440	1320	7	10
U37	100	25	10.15	17.82	0.0128	5	5	50	150	4	5
U38	30	10	31.67	26.24	0.0697	1	1	40	120	2	1
U39	300	100	32.96	10.76	0.003	8	8	440	1320	7	10
U40	200	50	6.78	12.89	0.0109	8	8	400	1200	7	10
U41	20	8	17.95	37.70	0.0283	1	1	30	90	2	1
U42	50	20	58.81	22.94	0.0098	1	1	45	135	2	1
U43	300	100	6.78	12.89	0.0109	8	8	100	300	7	8
U44	300	100	6.78	12.89	0.0109	8	8	100	300	7	8
U45	300	100	6.78	12.89	0.0109	8	8	110	330	7	8
U46	20	8	17.95	37.70	0.0283	1	1	30	90	2	1
U47	100	25	10.15	17.82	0.0128	5	5	50	150	4	5
U48	100	25	10.15	17.82	0.0128	5	5	50	150	4	5
U49	20	8	17.95	37.70	0.0283	1	1	30	90	2	1
U50	50	25	58.81	22.94	0.0098	2	2	45	135	2	2
U51	100	25	10.15	17.82	0.0128	5	5	50	150	4	5
U52	100	25	10.15	17.82	0.0128	5	5	50	150	4	5
U53	100	25	10.15	17.82	0.0128	5	5	50	150	4	5
U54	50	25	58.81	22.94	0.0098	2	2	45	135	2	2

Chapter 6

Conclusion and future prospect

This paper presents a binarized slime mould algorithm to solve single-objective thermal Unit Commitment Problem. Foraging behavior of slime mould and searching procedure for nutrition sources is mimicked in SMA, which is then translated to make it compatible with UCP parameters. Original SMA is made confined to discrete values with sigmoid transform, fitness values are determined with lambda iteration based Economic Load Dispatch and complex UCP constraints for all the search agents are handled by heuristic adjustments. BSMA is tested for 10, 20, 40, 60, 80 and 100 unit test systems and then compared with 20 renowned classical, heuristic-metaheuristic and hybridized approaches. Performance assessment tests has shown that BSMA achieved least production cost in a reasonable computational time compared to the other approaches for benchmark test systems. The convergence is found to be much quicker as the optimizer reaches it's final solution in lesser iterations. Comparatively lesser standard deviation for medium and large-scale test systems shows the potential of BSMA as a suitable optimizer for larger systems. For IEEE 118-bus system, BSMA outperforms the competing algorithms with the most cost effective solution. Therefore in future, proposed SMA can be modelled to solve other power system problems like binary natured profit-based Unit Commitment Problem (PBUCP) [58], [19], multi-objective Unit Commitment Problem (MOUCP), where MOUCP deals with emission reduction, reliability maximization and various uncertain gen- eration environments [56], [68], scheduling of wind-powered, hydro-powered and other clean energy sources with or without secutiry constraints [28], [73] and micro-grid associated stochastic optimization problems [51], [45]. Real coded hybridization with classical methods like GA, LR, EP, evolutionary methods like PSO, GWO, ACO and Quantum-inspired variant of SMA can also be developed to solve problems of power generation and operation domain like Combine Economic and Emission Dispatch (CEED) [57], [17] in future.

Bibliography

[1] Saleh Y Abujarad, MW Mustafa, and JJ Jamian. Recent approaches of unit commitment in the presence of intermittent renewable energy resources: A review. *Renewable and Sustainable Energy Reviews*, 70:215–223, 2017.

[2] Andrew Adamatzky. Simulating strange attraction of acellular slime mould physarum polycephaum to herbal tablets. *Mathematical and Computer Modelling*, 55(3-4):884–900, 2012.

[3] J Mary Anita and I Jacob Raglend. Solution of unit commitment problem using shuffled frog leaping algorithm. In *2012 International Conference on Computing, Electronics and Electrical Technologies (ICCEET)*, pages 109–115. IEEE, 2012.

[4] X Bai and H Wei. Semi-definite programming-based method for security-constrained unit commitment with operational and optimal power flow constraints. *IET Generation, Transmission & Distribution*, 3(2):182–197, 2009.

[5] Huseyin Hakan Balci and Jorge F Valenzuela. Scheduling electric power generators using particle swarm optimization combined with the lagrangian relaxation method. *International Journal of Applied Mathematics and Computer Science*, 14:411–421, 2004.

[6] M Basu. Dynamic economic emission dispatch using nondominated sorting genetic algorithm-ii. *International Journal of Electrical Power & Energy Systems*, 30(2):140–149, 2008.

[7] Madeleine Beekman and Tanya Latty. Brainless but multi-headed: decision making by the acellular slime mould physarum polycephalum. *Journal of molecular biology*, 427(23):3734–3743, 2015.

[8] Madeleine Beekman and Tanya Latty. Brainless but multi-headed: decision making by the acellular slime mould physarum polycephalum. *Journal of molecular biology*, 427(23):3734–3743, 2015.

[9] Syed Basit Ali Bukhari, Aftab Ahmad, Syed Auon Raza, and Muhammad Noman Siddique. A ring crossover genetic algorithm for the unit commitment problem. *Turkish Journal of Electrical Engineering & Computer Sciences*, 24(5):3862–3876, 2016.

[10] W. G. Camp. A method of cultivating myxomycete plasmodia. *Bulletin of the Torrey Botanical Club*, 63(4):205–210, 1936.

[11] K Chandram, N Subrahmanyam, and M Sydulu. Unit commitment by improved pre-prepared power demand table and muller method. *International Journal of Electrical Power & Energy Systems*, 33(1):106–114, 2011.

[12] K Chandrasekaran, S Hemamalini, Sishaj P Simon, and Narayana Prasad Padhy. Thermal unit commitment using binary/real coded artificial bee colony algorithm. *Electric Power Systems Research*, 84(1):109–119, 2012.

[13] K Chandrasekaran and Sishaj P Simon. Optimal deviation based firefly algorithm tuned fuzzy design for multi-objective ucp. *IEEE Transactions on power systems*, 28(1):460–471, 2012.

[14] Chuan-Ping Cheng, Chih-Wen Liu, and Chun-Chang Liu. Unit commitment by lagrangian relaxation and genetic algorithms. *IEEE transactions on power systems*, 15(2):707–714, 2000.

[15] CY Chung, Han Yu, and Kit Po Wong. An advanced quantum-inspired evolutionary algorithm for unit commitment. *IEEE Transactions on Power Systems*, 26(2):847–854, 2010.

[16] Arthur I Cohen and Miki Yoshimura. A branch-and-bound algorithm for unit commitment. *IEEE Transactions on Power Apparatus and Systems*, (2):444–451, 1983.

[17] A Lakshmi Devi and O Vamsi Krishna. Combined economic and emission dispatch using evolutionary algorithms-a case study. *ARPN Journal of engineering and applied sciences*, 3(6):28–35, 2008.

[18] Javad Ebrahimi, Seyed Hossein Hosseinian, and Gevorg B Gharehpetian. Unit commitment problem solution using shuffled frog leaping algorithm. *IEEE Transactions on Power Systems*, 26(2):573–581, 2010.

[19] M Jabbari Ghadi, Alfred Baghramian, and M Hosseini Imani. An ica based approach for solving profit based unit commitment problem market. *Applied Soft Computing*, 38:487–500, 2016.

[20] Moosa Moghimi Hadji and Behrooz Vahidi. A solution to the unit commitment problem using imperialistic competition algorithm. *IEEE Transactions on Power Systems*, 27(1):117–124, 2011.

[21] Kuk-Hyun Han and Jong-Hwan Kim. Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. *Trans. Evol. Comp*, 6(6):580–593, December 2002.

[22] Ali Asghar Heidari, Seyedali Mirjalili, Hossam Faris, Ibrahim Aljarah, Majdi Mafarja, and Huiling Chen. Harris hawks optimization: Algorithm and applications. *Future Generation Computer Systems*, 97:849–872, 2019.

[23] RA Jabr. Rank-constrained semidefinite program for unit commitment. *International Journal of Electrical Power & Energy Systems*, 47:13–20, 2013.

[24] Yun-Won Jeong, Woo-Nam Lee, Hyun-Houng Kim, Jong-Bae Park, and Joong-Rin Shin. Thermal unit commitment using binary differential evolution. *Journal of Electrical Engineering and Technology*, 4(3):323–329, 2009.

[25] Yun-Won Jeong, Jong-Bae Park, Se-Hwan Jang, and Kwang Y Lee. A new quantum-inspired binary pso for thermal unit commitment problems. In *2009 15th International Conference on Intelligent System Applications to Power Systems*, pages 1–6. IEEE, 2009.

[26] Yun-Won Jeong, Jong-Bae Park, Se-Hwan Jang, and Kwang Y Lee. A new quantum-inspired binary pso: application to unit commitment problems for power systems. *IEEE Transactions on Power Systems*, 25(3):1486–1495, 2010.

[27] Bin Ji, Xiaohui Yuan, Xianshan Li, Yuehua Huang, and Wenwu Li. Application of quantum-inspired binary gravitational search algorithm for thermal unit commitment with wind power integration. *Energy conversion and management*, 87:589–598, 2014.

[28] Ruiwei Jiang, Jianhui Wang, and Yongpei Guan. Robust unit commitment with wind power and pumped storage hydro. *IEEE Transactions on Power Systems*, 27(2):800–810, 2011.

[29] K. A. Juste, H. Kita, E. Tanaka, and J. Hasegawa. An evolutionary programming solution to the unit commitment problem. *IEEE Transactions on Power Systems*, 14(4):1452–1459, 1999.

[30] Vikram Kumar Kamboj. A novel hybrid pso–gwo approach for unit commitment problem. *Neural Computing and Applications*, 27(6):1643–1655, 2016.

[31] Vikram Kumar Kamboj. A novel hybrid pso–gwo approach for unit commitment problem. *Neural Computing and Applications*, 27(6):1643–1655, 2016.

[32] Vikram Kumar Kamboj, SK Bath, and JS Dhillon. Implementation of hybrid harmony search/random search algorithm for single area unit commitment problem. *International Journal of Electrical Power & Energy Systems*, 77:228–249, 2016.

[33] Noburo Kamiya. The control of protoplasmic streaming. *Science*, 92(2394):462–463, 1940.

[34] Peter Kareiva and Garrett Odell. Swarms of predators exhibit" preytaxis" if individual predators use area-restricted search. *The American Naturalist*, 130(2):233–270, 1987.

[35] S. A. Kazarlis, A. G. Bakirtzis, and V. Petridis. A genetic algorithm solution to the unit commitment problem. *IEEE Transactions on Power Systems*, 11(1):83–92, 1996.

[36] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95 - International Conference on Neural Networks*, volume 4, pages 1942–1948 vol.4, 1995.

[37] James Kennedy and Russell C Eberhart. A discrete binary version of the particle swarm algorithm. In *1997 IEEE International conference on systems, man, and cybernetics. Computational cybernetics and simulation*, volume 5, pages 4104–4108. IEEE, 1997.

[38] Dietrich Kessler. Plasmodial structure and motility. *Cell biology of Physarum and Didymium/edited by Henry C. Aldrich, John W. Daniel*, 1982.

[39] S Khanmohammadi, M Amiri, and M Tarafdar Haque. A new three-stage method for solving unit commitment problem. *Energy*, 35(7):3072–3080, 2010.

[40] Banumalar Koodalsamy, Manikandan Bairavan Veerayan, Chandrasekaran Koodalsamy, and Sishaj Pulikottil Simon. Firefly algorithm with multiple workers for the power system unit commitment problem. *Turkish Journal of Electrical Engineering & Computer Sciences*, 24(6):4773–4789, 2016.

[41] Vijay Kumar and Dinesh Kumar. Binary whale optimization algorithm and its application to unit commitment problem. *Neural Computing and Applications*, pages 1–29, 2018.

[42] Tanya Latty and Madeleine Beekman. Food quality and the risk of light exposure affect patch-choice decisions in the slime mold physarum polycephalum. *Ecology*, 91(1):22–27, 2010.

[43] Tanya Latty and Madeleine Beekman. Speed–accuracy trade-offs during foraging decisions in the acellular slime mould physarum polycephalum. *Proceedings of the Royal Society B: Biological Sciences*, 278(1705):539–545, 2011.

[44] Tanya Latty and Madeleine Beekman. Slime moulds use heuristics based on within-patch experience to decide when to leave. *Journal of Experimental Biology*, 218(8):1175–1179, 2015.

[45] Bei Li, Robin Roche, and Abdellatif Miraoui. Microgrid sizing with combined evolutionary algorithm and milp unit commitment. *Applied energy*, 188:547–562, 2017.

[46] Shimin Li, Huiling Chen, Mingjing Wang, Ali Asghar Heidari, and Seyedali Mirjalili. Slime mould algorithm: A new method for stochastic optimization. *Future Generation Computer Systems*, 2020.

[47] Rammohan Mallipeddi and Ponnuthurai Nagaratnam Suganthan. Unit commitment - a survey and comparison of conventional and nature inspired algorithms. *Int. J. Bio-Inspired Comput.*, 6(2):71–90, April 2014.

[48] Seyedali Mirjalili, Amir H Gandomi, Seyedeh Zahra Mirjalili, Shahrzad Saremi, Hossam Faris, and Seyed Mohammad Mirjalili. Salp swarm algorithm: A bio-inspired optimizer for engineering design problems. *Advances in Engineering Software*, 114:163–191, 2017.

[49] Seyedali Mirjalili, Seyed Mohammad Mirjalili, and Andrew Lewis. Grey wolf optimizer. *Advances in engineering software*, 69:46–61, 2014.

[50] Toshiyuki Nakagaki, Hiroyasu Yamada, and Tetsuo Ueda. Interaction between cell shape and contraction pattern in the physarum plasmodium. *Biophysical chemistry*, 84(3):195–204, 2000.

[51] Mohsen Nemati, Martin Braun, and Stefan Tenbohlen. Optimization of unit commitment and economic dispatch in microgrids based on genetic algorithm and mixed integer linear programming. *Applied energy*, 210:944–963, 2018.

[52] Nidhi, S. Reddy, R. Kumar, and B. K. Panigrahi. Binary bat search algorithm for unit commitment problem in power system. In *2017 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)*, pages 121–124, 2017.

[53] Taher Niknam, Amin Khodaei, and Farhad Fallahi. A new decomposition approach for the thermal unit commitment problem. *Applied Energy*, 86(9):1667–1674, 2009.

[54] Weerakorn Ongsakul and Nit Petcharaks. Unit commitment by enhanced adaptive lagrangian relaxation. *IEEE Transactions on Power Systems*, 19(1):620–628, 2004.

[55] Lokesh Kumar Panwar, Srikanth Reddy, and Rajesh Kumar. Binary fireworks algorithm based thermal unit commitment. *International Journal of Swarm Intelligence Research (IJSIR)*, 6(2):87–101, 2015.

[56] Lokesh Kumar Panwar, Srikanth Reddy, Ashu Verma, Bijaya K Panigrahi, and Rajesh Kumar. Binary grey wolf optimizer for large scale unit commitment problem. *Swarm and Evolutionary Computation*, 38:251–266, 2018.

[57] I Jacob Raglend and Narayana Prasad Padhy. Solutions to practical unit commitment problems with operational, power flow and environmental constraints. In *2006 IEEE Power Engineering Society General Meeting*, pages 8–pp. IEEE, 2006.

[58] I Jacob Raglend, C Raghuveer, G Rakesh Avinash, Narayana Prasad Padhy, and DP Kothari. Solution to profit based unit commitment problem using particle swarm optimization. *Applied Soft Computing*, 10(4):1247–1256, 2010.

[59] Provas Kumar Roy and Ranadhir Sarkar. Solution of unit commitment problem using quasi-oppositional teaching learning based algorithm. *International Journal of Electrical Power & Energy Systems*, 60:96–106, 2014.

[60] Navid Abdolhoseyni Saber, Mahdi Salimi, and Davar Mirabbasi. A priority list based approach for solving thermal unit commitment problem with novel hybrid genetic-imperialist competitive algorithm. *Energy*, 117:272–280, 2016.

[61] Dimitris N Simopoulos, Stavroula D Kavatza, and Costas D Vournas. Unit commitment by an enhanced simulated annealing algorithm. *IEEE Transactions on Power Systems*, 21(1):68–76, 2006.

[62] P. K. Singhal, R. Naresh, V. Sharma, and Goutham Kumar N. Enhanced lambda iteration algorithm for the solution of large scale economic dispatch problem. In *International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014)*, pages 1–6, 2014.

[63] K Srikanth, Lokesh Kumar Panwar, Bijaya K Panigrahi, Enrique Herrera-Viedma, Arun Kumar Sangaiah, and Gai-Ge Wang. Meta-heuristic framework: quantum inspired binary grey wolf optimizer for unit commitment problem. *Computers & Electrical Engineering*, 70:243–260, 2018.

[64] D. Srinivasan and J. Chazelas. A priority list-based evolutionary algorithm to solve large scale unit commitment problem. In *2004 International Conference on Power System Technology, 2004. PowerCon 2004.*, volume 2, pages 1746–1751 Vol.2, 2004.

[65] Chung-Ching Su and Yuan-Yih Hsu. Fuzzy dynamic programming: an application to unit commitment. *IEEE transactions on power systems*, 6(3):1231–1237, 1991.

[66] Yufei Tang, Chao Luo, Jun Yang, and Haibo He. A chance constrained optimal reserve scheduling approach for economic dispatch considering wind penetration. *IEEE/CAA Journal of Automatica Sinica*, 4(2):186–194, 2017.

[67] TO Ting, MVC Rao, and CK Loo. A novel approach for unit commitment problem via an effective hybrid particle swarm optimization. *IEEE Transactions on power systems*, 21(1):411–418, 2006.

[68] Anupam Trivedi, Dipti Srinivasan, Kunal Pal, and Thomas Reindl. A moea/d with non-uniform weight vector distribution strategy for solving the unit commitment problem in uncertain environment. In *Australasian Conference on Artificial Life and Computational Intelligence*, pages 378–390. Springer, 2017.

[69] K Vaisakh and LR Srinivas. Evolving ant colony optimization based unit commitment. *Applied Soft Computing*, 11(2):2863–2870, 2011.

[70] B Venkatesh, T Jamtsho, and HB Gooi. Unit commitment–a fuzzy mixed integer linear programming solution. *IET Generation, Transmission & Distribution*, 1(5):836–846, 2007.

[71] David C Walters and Gerald B Sheble. Genetic algorithm solution of economic dispatch with valve point loading. *IEEE transactions on Power Systems*, 8(3):1325–1332, 1993.

[72] Bo Wang, You Li, and Junzo Watada. Re-scheduling the unit commitment problem in fuzzy environment. In *2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*, pages 1090–1095. IEEE, 2011.

[73] Jianhui Wang, Mohammad Shahidehpour, and Zuyi Li. Security-constrained unit commitment with volatile wind power generation. *IEEE Transactions on Power Systems*, 23(3):1319–1327, 2008.

[74] Allen J Wood, Bruce F Wollenberg, and Gerald B Sheblé. *Power generation, operation, and control.* John Wiley & Sons, 2013.

[75] Xiaohui Yuan, Bin Ji, Shuangquan Zhang, Hao Tian, and Yanhong Hou. A new approach for unit commitment problem via binary gravitational search algorithm. *Applied Soft Computing*, 22:249–260, 2014.

[76] Xiaohui Yuan, Hao Nie, Anjun Su, Liang Wang, and Yanbin Yuan. An improved binary particle swarm optimization for unit commitment problem. *Expert Systems with applications*, 36(4):8049–8055, 2009.

[77] Xiaohui Yuan, Anjun Su, Hao Nie, Yanbin Yuan, and Liang Wang. Unit commitment problem using enhanced particle swarm optimization algorithm. *Soft Computing*, 15(1):139–148, 2011.

[78] Xiaohui Yuan, Hao Tian, Shuangquan Zhang, Bin Ji, and Yanhong Hou. Second-order cone programming for solving unit commitment strategy of thermal generators. *Energy conversion and management*, 76:20–25, 2013.

[79] B Zhao, CX Guo, BR Bai, and YJ Cao. An improved particle swarm optimization algorithm for unit commitment. *International Journal of Electrical Power & Energy Systems*, 28(7):482–490, 2006.

[80] Dexuan Zou, Steven Li, Xiangyong Kong, Haibin Ouyang, and Zongyan Li. Solving the dynamic economic dispatch by a memory-based global differential evolution and a repair technique of constraint handling. *Energy*, 147:59–80, 2018.

[81] Dexuan Zou, Steven Li, Xiangyong Kong, Haibin Ouyang, and Zongyan Li. Solving the combined heat and power economic dispatch problems by an improved genetic algorithm and a new constraint handling strategy. *Applied energy*, 237:646–670, 2019.