Real Power Loss Reduction and Voltage Stability Enhancement by Stock Exchange, Product Demand-Availability, Affluent and Penurious Algorithms

Kanagasabai Lenin

Department of EEE, Prasad V. Potluri Siddhartha Institute of Technology,
Kanuru, Vijayawada, Andhra Pradesh, 520007, India
E-mail: gklenin@gmail.com

Abstract. In this paper, the Stock Exchange Algorithm (SEA), the Product Demand-Availability (PDA) algorithm, and the Affluent and Penurious (AP) algorithm are proposed to solve the power loss reduction problem. In the SEA approach, selling and buying shares in the stock exchange was imitated to design the algorithm. Stockholders are classified as Privileged, Average or Weak based on their fitness value. The PDA optimization algorithm is based on the consumer demand and availability of a product in the market. The Affluent and Penurious algorithm mimics the social behavior of people. The gap parameter (G) is defined to indicate the growing gap between affluent and penurious people when affluent people increase their wealth. The proposed Stock Exchange Algorithm, Product Demand-Availability optimization algorithm and the Affluent and Penurious optimization algorithm were tested in the IEEE 30 bus system. Real power loss minimization, voltage deviation minimization, and voltage stability index enhancement were successfully attained.

Keywords: *affluent and penurious; availability-requirement; gap parameter; optimal reactive power; product; shares; stock exchange; transmission loss.*

1 Introduction

Real power loss minimization, voltage stability enhancement and voltage deviation minimization were the main objectives of this work. Many conventional [1,2] numerical methods, called deterministic methods, such as gradient search (GS) [3], Newton method (NM) [4], interior point method (IPM) [5-7], linear program (LP) [8-10], dynamic programming method (DPM)[11], quadratic programming method (QPM) [12,13], and Lagrangian method (LM) [14], can find optimal solutions with adequate quality, but these methods have several disadvantages, such as high time consumption, high number of iterations, large number of computations, incapability of handling non-differentiable constraints and easily falling into a local optimum solution zone. In recent times, metaheuristic methods inspired by natural phenomena such as animal behavior have been more widely and successfully applied for solving problems such as the

Received July 2nd, 2020, Revised November 11th, 2020, Accepted for publication November 21st, 2020.
 ISSN: 2337-5787, DOI: 10.5614/itbj.ict.res.appl.2020.14.2.5

optimal reactive power dispatch (ORPD) problem. Many methods have been developed, creating large families of methods, such as variants of genetic algorithms (GA) [15-19], variants of differential evolution (DE) algorithms [20-24], variants of particle swarm optimization (PSO) algorithms [25-31], variants of gravitational search algorithms (GSA) [32-35], and many other new standard methods [36-48].

In this work, three algorithm were designed to solve the ORPD problem. Firstly, the Stock Exchange Algorithm, where the ORPD is equated with a person acting in the stock exchange. People buy shares of any company with reference to the market and their financial conditions. The variables of the reactive power problem are represented by shares; a person buying shares initiates the variables. Secondly, the Product Demand-Availability optimization algorithm was designed, which is based on the consumer demand and availability of a product in the market. When the product is introduced in the market it initially faces oscillation between demand and availability, but after some time it reaches a stable point. Finally, the Affluent and Penurious (AP) optimization algorithm was designed, which emulates the social behavior of people. Two groups of people are created: Affluent and Penurious. The gap parameter (G) in the proposed algorithm indicates the status of each person. The proposed Stock Exchange Algorithm, the Product Demand-Availability (PDA) optimization algorithm, the Affluent and Penurious (AP) optimization algorithm were tested in the IEEE 30 bus system and the IEEE 14, 30, 57, 118, 300 bus test systems without considering the voltage stability index. The proposed algorithms reduced the power loss effectively and the control variables were within the limits.

2 Problem Formulation

Power loss minimization is defined by:

$$Min\ \widetilde{OBF}(\underline{r},\underline{u}) \tag{1}$$

Subject to:

$$L(\underline{r},\underline{u}) = 0 \tag{2}$$

$$M(\underline{r},\underline{u}) = 0 \tag{3}$$

$$r = [VLG_1, .., VLG_{Ng}; QC_1, .., QC_{Nc}; T_1, .., T_{N_T}] \tag{4}$$

$$u = [PG_{slack}; VL_1, .., VL_{N_{Load}}; QG_1, .., QG_{Ng}; SL_1, .., SL_{N_T}] \tag{5}$$

$$F_1 = P_{Minimize} = Minimize\left[\sum_m^{NTL} G_m\left[V_i^2 + V_j^2 - 2 * V_i V_j cos\emptyset_{ij}\right]\right] \tag{6}$$

$$F_2 = Minimize\left[\sum_{i=1}^{N_{LB}}\left|V_{Lk} - V_{Lk}^{desired}\right|\left|V_{Lk} - V_{Lk}^{desired}\right|^2 + \sum_{i=1}^{Ng}\left|Q_{GK} - Q_{KG}^{Lim}\right|^2\right] \tag{7}$$

$$F_3 = Minimize\ L_{MaxImum} \tag{8}$$

$$L_{Maximum} = Maximum\left[L_j\right]; j = 1; N_{LB} \tag{9}$$

and $$\{L_j = 1 - \sum_{i=1}^{NPV} F_{ji}\frac{V_i}{V_j}\ F_{ji} = -[Y_1]^1[Y_2] \tag{10}$$

$$L_{Maximum} = Maximum\left[1 - [Y_1]^{-1}[Y_2] \times \frac{V_i}{V_j}\right] \tag{11}$$

Equality constraints:

$$0 = PG_i - PD_i - V_i\sum_{j\in N_B} V_j\left[G_{ij}cos\left[\emptyset_i - \emptyset_j\right] + B_{ij}sin\left[\emptyset_i - \emptyset_j\right]\right] \tag{12}$$

$$0 = QG_i - QD_i - V_i\sum_{j\in N_B} V_j\left[G_{ij}sin\left[\emptyset_i - \emptyset_j\right] + B_{ij}cos\left[\emptyset_i - \emptyset_j\right]\right] \tag{13}$$

Inequality constraints:

$$P_{gslack}^{minimum} \leq P_{gslack} \leq P_{gslack}^{maximum} \tag{14}$$

$$Q_{gi}^{minimum} \leq Q_{gi} \leq Q_{gi}^{maximum}, i \in N_g \tag{15}$$

$$VL_i^{minimum} \leq VL_i \leq VL_i^{maximum}, i \in NL \tag{16}$$

$$T_i^{minimum} \leq T_i \leq T_i^{maximum}, i \in N_T \tag{17}$$

$$Q_c^{minimum} \leq Q_c \leq Q_C^{maximum}, i \in N_C \tag{18}$$

$$\left|SL_i\right| \leq S_{L_i}^{maximum}, i \in N_{TL} \tag{19}$$

$$VG_i^{minimum} \leq VG_i \leq VG_i^{maximum}, i \in N_g \tag{20}$$

The multi-objective fitness (MOF) function is defined by:

$$MOF = F_1 + r_iF_2 + uF_3 = F_1 + \left[\sum_{i=1}^{NL} x_v\left[VL_i - VL_i^{min}\right]^2 + \sum_{i=1}^{NG} r_g\left[QG_i - QG_i^{min}\right]^2\right] + r_f F_3 \tag{21}$$

$$VL_i^{minimum} = \{VL_i^{max}, VL_i > VL_i^{max}\ VL_i^{min}, VL_i < VL_i^{min} \tag{22}$$

$$QG_i^{minimum} = \{QG_i^{max}, QG_i > QG_i^{max}\ QG_i^{min}, QG_i < QG_i^{min} \tag{23}$$

3 Stock Exchange Algorithm

The Stock Exchange Algorithm is based on the process of the stock market. Stockholders who fall in the highest class (Privileged) hold on to their shares to enjoy their gains; they form 10% to 30% of the total population. Stockholders that fall in the Average class form 20% to 50% of the total population. The difference between both classes can be evaluated with respect to their stock or share value:

$$\begin{aligned} &stock\ holder\ population_j^{category\ (B)} \\ &\quad = random\ number \\ &\quad \times stock\ holder\ population_{A,i}^{category\ (A)} \\ &\quad + (1 - random\ number) \\ &\quad \times Stock\ holder\ population_{B,i}^{category\ (A)}\ ; i \\ &\quad = 1,2,3, \ldots, n_i\ ; j = 1,2.., n_j \end{aligned} \tag{24}$$

Stockholders who possess the lowest fitness value fall in the Weak class. They form 30% to 50% of the population and they exchange stock or shares to attain gains depending on the conditions.

$$\begin{aligned} &change\ in\ share\ (AS) \\ &\quad = 2 \times random_1 \\ &\quad \times \Big(stock\ holder\ population_{i,A}^{category\ (A)} \\ &\quad - Stock\ holder\ population_k^{category\ (C)}\Big) + 2 \\ &\quad \times random_2 \\ &\quad \times \Big(Stock\ holder\ population_{i,B}^{category\ (A)} \\ &\quad - Stock\ holder\ population_k^{category\ (C)}\Big) \end{aligned} \tag{25}$$

$$\begin{aligned} &stock\ holder\ population_k^{category\ (C),new} \\ &\quad = stock\ holder\ population_k^{category\ (C)} + 0.799 \\ &\quad \times Altering\ in\ share\ (AS) \end{aligned} \tag{26}$$

During fluctuating conditions stockholders who fall in the Privileged class have an excellent solution to the problem; they form 10% to 30% of the total population.

In the initial conditions the value of the stock or shares is increased.

$$number\ of\ shares\ (\ \Delta S_{t\ A}) = S_{t\ A} - (\ stock\ exchange\ information(\delta) + (2 \times random\ number \times constant\ co\ effiecient\ (\mu) \times stage\ of\ risk\ (\ \tau_A)) \quad (27)$$

$$\mu = \frac{t - th\ person\ in\ the\ stock\ exchange}{last\ person\ in\ the\ stock\ exchange} \quad (28)$$

$$S_{t\ A} = \sum_{y=1}^{n} \quad stock\ or\ shares\ of\ the\ t - th\ person_y\ \ ; y = 1,2,3, .. , n \quad (29)$$

$$\tau_A = S_{t\ A} \times stock\ exchange\ risk\ (ser_A) \quad (30)$$

$$ser_A^k = ser_{A,maximum} - \frac{ser_{A,maximum} - ser_{A,minimum}}{iteration_{maximum}} \times k \quad (31)$$

When the stockholder possesses no information about the stock exchange conditions then $stock\ exchange\ information(\delta)$ is equal to each person's total stock value in stable conditions.

$$number\ of\ shares\ (\ \Delta S_{t\ B}) = S_{t\ B} - stock\ exchange\ information(\delta) \quad (32)$$

During fluctuating conditions in the stock exchange few people sell stock or shares. Some buy shares but the total market stock or share value stays the same. Stockholders who fall in the Weak class will exchange their stocks to obtain the best cost value. They will try to reach the best stock composition by buying and selling stocks.

$$number\ of\ shares\ (\ \Delta S_{t\ C}) = (4\ \times\ random\ number([-0.5,0.5])\ \times\ stock\ exchange\ information(\delta)\ \times\ stage\ of\ risk\ (\ \tau_B)\ ) \quad (33)$$

$$random\ number = 0.5 - random \quad (34)$$

$$\tau_B = S_{t\ B} \times stock\ exchange\ risk\ (ser_B) \quad (35)$$

1. Begin
2. Choose the initial values
3. Find the quality of the stock with reference to the initial stockholders
4. Calculate the total cost of the stockholders
5. Calculate the stockholder ranking
6. Group the stockholders in the classes Privileged, Average and Weak

   --- Stable condition ---
7. For the Average class of stock holders, a change in stock value and stock exchange balance conditions is analyzed by

$$stock\ holder\ population_j^{category\ (B)} = random\ number \times stock\ holder\ population_{A,i}^{category\ (A)} + \ (1 - random\ number) \times Stock\ holder\ population_{B,i}^{category\ (A)}\ \ ; i = 1,2,3, \dots, n_i\ ; j = 1,2.. , n_j$$

8. For the Weak class of stockholders, the change in stock value and stock exchange balance conditions is analyzed by

$$change\ in\ share\ (AS) = 2 \times random_1 \times \left(stock\ holder\ population_{i,A}^{category\ (A)} - stock\ holder\ population_{k}^{category\ (C)}\right) + 2 \times random_2 \times \left(Stock\ holder\ population_{i,B}^{category\ (A)} - Stock\ holder\ population_{k}^{category\ (C)}\right)$$

9. Calculate the total cost of the stockholders
10. Calculate the stockholder ranking
11. Group the stockholders in the classes Privileged, Average and Weak

--- Fluctuating condition ---

12. In the Average class of stockholders, the change in stock value and stock exchange balance conditions is analyzed by

$$number\ of\ shares\ (\ \Delta S_{t\ A}) = S_{t\ A} - (stock\ exchange\ information(\delta) + (2 + random\ number \times constant\ co\ effiecient\ (\mu) \times stage\ of\ risk\ (\ \tau_A))$$

13. In the Weak class of stockholders, the change in stock value and stock exchange balance conditions is analyzed by

$$number\ of\ shares\ (\ \Delta S_{t\ C}) = (4 \times random\ number([-0.5{,}0.5]).\ stock\ exchange\ information(\delta) \times stage\ of\ risk\ (\ \tau_B)\ )$$

14. If the end condition is reached then stop, else go to step ‘d’
15. End

4 Product Demand-Availability Algorithm

The Product Demand-Availability (ARP) optimization algorithm is based on consumer demand and availability of a product in the market. If the demand of that particular product increases, then the producer will increase its production so that it will be available to more consumers. The availability and demand mechanism was imitated in the design of the algorithm. Any product has a current price that reflects the market conditions, denoted by Pp_i, and the availability of the product in running time and is denoted by A_{t+1}. The availability in the market and consumer demand varies and can be written as the following linear function:

$$A_{t+1} = f(Pp_i) \tag{36}$$

The price of product Pp_{t+1} at a later stage is determined with respect to product availability A_{t+1} and demand (*D*) of the product in the market.

$$Pp_{t+1} = D(A_{t+1}) \tag{37}$$

When there is a high rise in availability of the product in the market then the price may fall steeply, so that D is a decreasing function. The price stability (ST_o) and product stability (SU_o) intersect at point $P(ST_o, SU_o)$.

Function f is described as:

$$A_{t+1} - A_o = c(Pp_i - Pp_o) \tag{38}$$

Function D is defined by:

$$Pp_{t+1} - Pp_o = -d(A_{t+1} - A_o) \tag{39}$$

At a particular instant, when linear coefficients $|c, d| < 1$ with respect to the demand (D), function f will have a steep value. There will be fluctuations between demand and availability but after some iterations they both reach equilibrium P (ST_o, SU_o). When $|c, d| > 1$, the demand (D) has a steep value with reference to function f. Then the fluctuations between demand and availability will increase and stability point P (ST_o, SU_o) is deviated.

The price and demand of the product are defined by the following matrix:

$$T = [T_1\ T_2 \ldots T_n\ ] = \left[T_1^1\ \cdots\ T_1^d\ \vdots \ddots \vdots\ T_n^1\ \cdots\ T_n^d\ \right] \tag{40}$$

$$U = [U_1\ U_2 \ldots U_n\ ] = \left[U_1^1\ \cdots\ U_1^d\ \vdots \ddots \vdots\ U_n^1\ \cdots\ U_n^d\ \right] \tag{41}$$

In n markets the fitness values of the price and requirement of the product are denoted by:

$$Ft = [Ft_1, Ft_2, .., Ft_n]^T \tag{42}$$

$$Fu = [Fu_1, Fu_2, .., Fu_n]^T \tag{43}$$

The fluctuation and stability of the PDA algorithm is utilized to do exploration and exploitation. First, the iteration's values of price stability (ST_o) and product stability (SU_o) are determined. The product stability vector is defined by:

$$G_i = \left|Fu_i - \frac{1}{n}\sum_{i=1}^{n} Fu_i\right| \tag{44}$$

$$roulette\ wheel\ selection\ (W)\ = \frac{G}{\sum_{i=1}^{n} u_i} \tag{45}$$

$$u_o = u_w \tag{46}$$

The price stability vector of the product is defined by:

$$H_i = \left|Ft_i - \frac{1}{n}\sum_{i=1}^{n} Ft_i\right| \tag{47}$$

$$roulette\ wheel\ selection\ (W)\ = \frac{H}{\sum_{i=1}^{n} H_i} \tag{48}$$

$$\{random_1 . \frac{\sum_1^n t_i}{1} \ if \ random < 0.5 \ t_w \qquad if \ random \geq 0.5 \qquad (49)$$

With respect to price stability (ST_o) and product stability (SU_o), availability and demand are defined by:

$$u_i(t+1) = u_o + \ weight \ factor \ (\alpha) \cdot (u_i(t) - u_o) \qquad (50)$$

$$t_i(t+1) = t_o - \ weight \ factor \ (\beta) \cdot (t_i(t+1) - t_o) \qquad (51)$$

With respect to product availability, the demand equation can be written as follows:

$$t_i(t+1) = t_o - \ weight \ factors \ (\alpha\beta) \cdot (t_i(t) - t_o) \qquad (52)$$

$$\alpha = \frac{2\cdot(max.iter - iter + 1)}{max.iter} \cdot sin \ sin \ (2\pi r) \ ; \ r = [0,1] \qquad (53)$$

$$\beta = 2. cos(2\pi r); \ r = [0,1] \qquad (54)$$

$$\alpha\beta = \frac{4\cdot(max.iter - iter + 1)}{max.iter} \cdot sin \ sin \ (2\pi r) \ cos(2\pi r); \ r = [0,1] \qquad (55)$$

1. Start
2. Initialization of population and weights
3. Price stability (ST_o) and product stability (SU_o) are arbitrarily initialized
4. Calculate the fitness values
5. Replace the values by the best found values
6. As long as the stop criterion is not satisfied do:
7. For each product market $i = (1,2..,n)$
8. Define the stability of product price, availability and requirement with:

$G_i = \left| Fu_i - \frac{1}{n} \sum_{i=1}^n Fu_i \right|$

9. $roulette \ wheel \ selection \ (W) \ = \frac{G}{\sum_{i=1}^n u_i}$

$u_o = u_w$

$H_i = \left| Ft_i - \frac{1}{n} \sum_{i=1}^n Ft_i \right|$

10. $roulette \ wheel \ selection \ (W) \ = \frac{H}{\sum_{i=1}^n H_i}$

$\{random_1 . \frac{\sum_1^n t_i}{1} \ if \ random < 0.5 \ t_w \qquad if \ random \geq 0.5$

11. Upgrade the quantities by:

$u_i(t+1) = u_o + \ weight \ factor \ (\alpha) \cdot (u_i(t) - u_o)$

$t_i(t+1) = t_o - \ weight \ factor \ (\beta) \cdot (t_i(t+1) - t_o)$

12. Calculate the fitness value of $Ft \ and \ Fu$

13. When $Fu\ is\ better\ than\ Ft$ then replace $Ft\ by\ Fu$
14. End if
15. End for
16. Update the optimal solution
17. End while
18. Return the optimal solution

5 Affluent and Penurious Algorithm

The Affluent and Penurious (AP) optimization algorithm generates a population with a lower bound and an upper bound as the exploration space. Two subpopulations are distinguished, Affluent and Penurious:

$$population_{N\,(main)} = population_{N(Affluent)} + population_{N(Penurious)} \quad (56)$$

The algorithm imitates social behavior, i.e. the affluent dominate and have a better position than the penurious. Referring to this, Eq. (25) below was designed:

$$value_1 < value_2 < value_3 < \cdots < value_m < value_{m+1} < value_{m+2} \ldots < value_n \quad (57)$$

Position changes of members in the Affluent group are defined by:

$$\overrightarrow{Y_{Affluent,i}^{New}} = \overrightarrow{Y_{Affluent,i}^{Old}} + gap\ parameter\ (G)\left[\overrightarrow{Y_{Affluent,i}^{Old}} - \overrightarrow{Y_{Penurious\,,best}^{Old}}\right] \quad (58)$$

$\overrightarrow{Y_{Affluent,i}^{New}}$ represents the *i*-th position of the new value of the affluent population, $\overrightarrow{Y_{Affluent,i}^{Old}}$ represents the current value. The top most member in the penurious group is indicated by $\overrightarrow{Y_{Penurious\,,best}^{Old}}$. Gap parameter G represents the distance between the Affluent and the Penurious classes. G indicates the positions of the people in the population relative to each other. The value of *G* is a random number [0, 1]. Movement in the position of members in the Penurious class is defined by:

$$\overrightarrow{Y_{Penurious,i}^{New}} = \overrightarrow{Y_{Penurious,i}^{Old}} + \left[G(UP) - \overrightarrow{Y_{Penurious\,,i}^{Old}}\right] \quad (59)$$

$\overrightarrow{Y_{Penurious,i}^{New}}$ represents the *i*-th position of the new value of the Penurious population, $\overrightarrow{Y_{Penurious,i}^{Old}}$ represents the current value, $G(UP)$ is the upgraded affluent parameter ([0,1]).

$$UP = \frac{\overrightarrow{Y^{Old}_{Affluent,best}} + \overrightarrow{Y^{Old}_{Affluent,mean}} + \overrightarrow{Y^{Old}_{Affluent,low}}}{3} \tag{60}$$

$\overrightarrow{Y^{Old}_{Affluent,best}}$ represents the position of the best member in the Affluent population, $\overrightarrow{Y^{Old}_{Affluent,mean}}$ represents the position of an average member in the Affluent population, $\overrightarrow{Y^{Old}_{Affluent,low}}$ indicates the position of the lowest member in the Affluent population. The value of UP is fixed in each iteration and then *G* determines the level of enhancement, which leads to an increase of $\overrightarrow{Y^{Old}_{Penurious,i}}$. There will be an increase of the upgrading parameter when the value of *G* is 0. When $\overrightarrow{Y^{Old}_{Penurious,i}}$ possesses a value that is greater than the value of *G*, then $\overrightarrow{Y^{Old}_{Penurious\ ,i}}$ will have a large increase and vice versa. Variation in the value of *G* creates strong competition in the Penurious population. This means that when the value of *G* is small then there will be a large increase of *UP*. In the AP algorithm, 0 means a normal distribution and 1 means variance. These values are used as mutation for the Affluent and the Penurious populations respectively. The mutation of Affluent and Penurious is defined as follows:

$$if\ random < probability\ of\ mutation\ \ then;\overrightarrow{Y^{New}_{Affluent,\iota}} = \overrightarrow{Y^{New}_{Affluent,\iota}}\, random(nd) \tag{61}$$

$$if\ random < probability\ of\ mutation\ \ then;\overrightarrow{Y^{New}_{Penurious,\iota}} = \overrightarrow{Y^{New}_{Penurious,\iota}}\, random(nd) \tag{62}$$

The random value is between 0 and 1; $random(nd)$ is the normalized distribution value and is obtained from the normal distribution of mean (0) and variance (1).

1. Start
2. Initialization of the population
3. Classification of the population {Affluent, Penurious}

// Affluent population //

4. Choose an Affluent individual
5. Choose the best Penurious individual
6. Update the population

$\overrightarrow{Y^{New}_{Affluent,i}} = \overrightarrow{Y^{Old}_{Affluent,i}} +\ gap\ parameter\ (G) \left[\overrightarrow{Y^{Old}_{Affluent,i}} - \overrightarrow{Y^{Old}_{Penurious\ ,best}}\right]$

7. Apply mutation

$$if\ random < probability\ of\ mutation, then\ \overrightarrow{Y_{Affluent,i}^{New}} = \overrightarrow{Y_{Affluent,i}^{New}}\ random(nd)$$

8. Calculate the value of the Affluent individual
9. Are there any other Affluent individuals? If yes go to step d
10. Else combine the population and classify on the basis of new, old Affluent and Penurious.

// Penurious population //

11. Choose a Penurious individual
12. Calculate the upgraded parameter UP

$$UP = \frac{\overrightarrow{Y_{Affluent,best}^{Old}} + \overrightarrow{Y_{Affluent,mean}^{Old}} + \overrightarrow{Y_{Affluent,low}^{Old}}}{3}$$

13. Update the population

$$\overrightarrow{Y_{Penurious,i}^{New}} = \overrightarrow{Y_{Penurious,i}^{Old}} + \left[G(UP) - \overrightarrow{Y_{Penurious\ ,i}^{Old}}\right]$$

14. Apply mutation

$$if\ random < probability\ of\ mutation, then\ \overrightarrow{Y_{Penurious,i}^{New}} = \overrightarrow{Y_{Penurious,i}^{New}}\ random(nd)$$

15. Calculate the value of Penurious
16. Are there any other Penurious individuals? If yes then go to step k
17. Else combine the population and classify on the basis of new, old Affluent and Penurious.

// Affluent, Penurious //

18. Separate Affluent population
19. Separate penurious population
20. Is the end criterion satisfied?
21. If yes then pick the best Affluent individual
22. Else go to step d
23. End

6 Simulation Results

The proposed Stock Exchange Algorithm, the Product Demand-Availability algorithm, the Affluent and Penurious algorithm were verified in the standard IEEE 30 bus system [49]. Table 1 and 2 show the variables and limits. Table 3 to 6 give a comparison of the real power loss. Then the validity of the proposed Stock Exchange Algorithm, the Product Demand-Availability algorithm, and the Affluent and Penurious algorithm was tested without considering the voltage stability index in the IEEE 14, 30, 57, 118, 300 bus test systems. Table 7-11 shows a comparison of the power loss.

Table 1 Constraints of control variables.

	Minimum (PU)	Maximum (PU)
Generator voltage	0.9500	1.100
Transformer tap	0.9000	1.100
VAR source	0.0000	5.00 (MVAR)

Table 2 System parameters.

Power loss (base case) MW	5.66000
Base case for VD (PU)	0.58217

Table 3 Comparison of real power loss with different metaheuristic algorithms.

	Differential Evolution (DE) [50]	Gravitational Search Algorithm (GSA) [51]	APO-PSO [52]	SEA	ARP	AP
Power Loss in MW	4.5550	4.51430	4.39800	4.241	4.235	4.229
VD in PU	1.95890	0.875220	1.04700	1.032	1.039	1.032
L-index in PU	0.55130	0.141090	0.12670	0.1211	0.1229	0.1219

Table 4 Comparison of different algorithms with reference to voltage stability improvement.

	Differential Evolution (DE) [50]	Gravitational Search Algorithm (GSA) [51]	APO-PSO [52]	SEA	ARP	AP
Power Loss in MW	6.475500	6.91170	5.6980	5.419	5.413	5.401
VD in PU	0.091100	0.06760	0.0870	0.080	0.075	0.082
L-index in PU	0.143520	0.13490	0.13770	0.1321	0.1329	0.1326

Table 5 Comparison with reference to voltage deviation minimization.

	Differential Evolution (DE) [50]	Gravitational Search Algorithm (GSA) [51]	APO-PSO [52]	SEA	ARP	AP
Power Loss in MW	7.073300	4.975200	4.478000	4.236	4.232	4.227
VD in PU	1.419000	0.2157900	1.857900	1.8212	1.8210	1.8209
L-index in PU	0.124600	0.1368400	0.122700	0.1181	0.1184	0.1189

Table 6 Comparison of values with reference to multi-objective formulation.

	APO-PSO [52]	SEA	ARP	AP
Power Loss in MW	4.84200	4.741	4.735	4.739
VD in PU	1.00900	1.006	1.001	1.001
L-index in PU	0.11920	0.1193	0.1195	0.1196

Table 7 Comparison of loss with respect to IEEE 14 bus system.

	Value of Base case [56]	Modified PSO (MPSO) [56]	Basic PSO (PSO) [55]	Standard EP [54]	SAR-GA [54]	SEA	ARP	AP
Percentage of reduction in power loss	0.000	9.200	9.100	1.50	2.50	18.14	16.08	16.30
Power loss (Mw)	13.550	12.293	12.315	13.346	13.216	11.091	11.370	11.340

Table 8 Comparison of power loss with respect to IEEE 30 bus system.

	Value of Base case [56]	Modified PSO (MPSO) [56]	Basic PSO (PSO) [55]	Standard EP [54]	SAR-GA [54]	SEA	ARP	AP
Percentage of reduction in power loss	0.000	8.400	7.400	6.600	8.300	25.86	22.62	22.56
Power loss (Mw)	17.550	16.070	16.250	16.380	16.090	13.01	13.58	13.59

Table 9 Comparison of power loss with respect to IEEE 57 bus system.

	Base case value [56]	Modified PSO (MPSO) [56]	Basic PSO (PSO) [55]	Canonical-GA [53]	Adaptive GA [53]	SEA	ARP	AP
Percentage of reduction in power loss	0.000	15.400	14.100	9.200	11.600	28.02	25.47	26.68
Power loss (Mw)	27.800	23.510	23.860	25.240	24.56 0	20.010	20.719	20.382

Table 10 Comparison of real power loss with respect to IEEE 118 bus system.

	Base case value [56]	Modified PSO (MPSO) [56]	Basic PSO [55]	I-PSO [55]	CL-PSO [53]	SEA	ARP	AP
Percentage of reduction in power loss	0.000	11.700	10.100	10.600	11.300	14.61	14.28	14.90
Power loss (Mw)	132.80	117.19	119.34	131.99	130.96	113.39	113.83	113.01

Table 11 Comparison of real power loss with respect to IEEE 300 bus system.

	Enhanced GA (EGA) [58]	Enhanced FA (EFA) [58]	Cuckoo search algorithm [57]	SEA	ARP	AP
Power loss (MW)	646.29980	650.60270	635.89420	610.0099	610.8135	610.2129

Real power loss reduction was attained and the percentage of power loss reduction was improved. A comparison was made with a number of standard algorithms: Modified Particle Swarm Optimization algorithm, Basic Particle Swarm Optimization algorithm, Adaptive Genetic algorithm, Canonical Genetic algorithm, and the Comprehensive Learning Particle Swarm Optimization algorithm. All three proposed algorithms performed well in terms of power loss reduction.

7 Conclusion

In this work, the Stock Exchange Algorithm (SEA), the Product Demand-Availability (PDA) algorithm and the Affluent and Penurious (AP) algorithm were designed to solve the reactive power dispatch problem. In the SEA, the ORPD problem is equated to persons' actions in the stock exchange, where the variables of the reactive power problem are represented by shares; a person buying shares initiates the variables of the ORDP problem. In the PDA algorithm, demand and price are treated as solutions, both being updated throughout the iterations. Finally, the AP algorithm generates a population with a lower bound and upper bound as the exploration space and two subpopulations are created, Affluent and Penurious.

The three proposed algorithms were verified in the standard IEEE 30 bus system. They were then evaluated in the IEEE 14, 30, 57, 118, 300 bus test systems without L-index. The real power loss obtained by SEA, PDA and AP was 4.241

(MW), 4.235 (MW) and 4.229 (MW), respectively. The percentage of real power loss reduction obtained by SEA, PDA and AP for the IEEE 14, 30, 57, 118, 300 bus test systems was: 18.14%, 16.08%, 16.30%; 25.86%, 22.62%, 22.56%; 28.02%, 25.47%, 26.68%; 14.61%, 14.28%, 14.90%. Real power loss minimization, voltage deviation minimization, and voltage stability index enhancement were attained. The percentage of power loss reduction was improved. In the future, the SEA, the PDA algorithm, and the AP algorithm will be applied to larger systems. The proposed algorithms can also be applied to other power system problems in both online and offline mode.

References

[1] Lee, K., Park, Y., Ortiz, J., *A United Approach to Optimal Real and Reactive Power Dispatch*. IEEE Transactions Power Apparatus System, Vol. PAS **104**, pp. 1147-1153, 1985.

[2] Domel, H.W. & Tinney, W.F., *Optimal Power Flow Solutions*, IEEE Transactions, **PAS-87**, pp. 1866-1876, 1968.

[3] Yu, D.C., Fagan, J.E., Foote, B. & Aly, A.A., *An Optimal Load Flow Study by The Generalized Reduced Gradient Approach*, Electric Power Systems Research, **10**(1), pp. 47-53, 1986.

[4] Sun, D.I., Ashley, B., Brewer, B., Hughes, A. & Tinney, W.F., *Optimal Power Flow by Newton Approach*, IEEE Transactions on Power Apparatus and Systems, **103**(10), pp. 2864-2880, 1984.

[5] Granville, S., *Optimal Reactive Dispatch Through Interior Point Methods*, IEEE Transactions on Power Systems, **9**(1), pp. 136-146, 1994.

[6] Rezania, E. & Shahidehpour, S.M., *Real Power Loss Minimization Using Interior Point Method*, International Journal of Electrical Power & Energy Systems, **23**(1), pp. 45-56, 2001.

[7] Zhu, J. & Xiong, X., *Optimal Reactive Power Control Using Modified Interior Point Method*, Electric Power Systems Research, **66**(2), pp. 187-192, 2003.

[8] Alsac, O., Bright, J., Prais, M. & Stott, B., *Further Developments In LP-Based Optimal Power Flow*, IEEE Transactions on Power Systems, **5**(3), pp. 697-711, 1990.

[9] Stott, B. & Marinho, J.L., *Linear Programming for Power System Network Security Applications*, IEEE Transactions on Power Apparatus and Systems, **98**(3), pp. 837-848, 1979.

[10] Chebbo, A.M. & Irving, M.R., *Combined Active and Reactive Dispatch. I. Problem Formulation and Solution Algorithm*, IEE Proceedings-Generation, Transmission and Distribution, **142**(4), pp. 393-400, 1995.

[11] Lu, F.C. & Hsu, Y.Y., *Reactive Power/Voltage Control in A Distribution Substation Using Dynamic Programming*, IEE Proceedings Generation, Transmission and Distribution, **142**(6), pp. 639-645, 1995.

[12] Quintana, V.H. & Santos-Nieto, M., *Reactive-Power Dispatch by Successive Quadratic Programming*, IEEE Transactions on Energy Conversion, **4**(3), pp. 425-435, 1989.

[13] Grudinin, N., *Reactive Power Optimization Using Successive Quadratic Programming Method*, IEEE Transactions on Power Systems, **13**(4), pp. 1219-1225, 1998.

[14] de Sousa, V., Baptista, E. & da Costa, G., *Optimal Reactive Power Flow Via the Modified Barrier Lagrangian Function Approach*, Electric Power Systems Research, **84**(1), pp. 159-164, 2012.

[15] Cao, Y.J. & Wu, Q.H., *Optimal Reactive Power Dispatch Using an Adaptive Genetic Algorithm*, in Proceedings of the Second International Conference on Genetic Algorithms in Engineering Systems, pp. 117–122, Glasgow, UK, 1997.

[16] Khanmiri, D.T., Nasiri, N. & Abedinzadeh, T., *Optimal Reactive Power Dispatch Using an Improved Genetic Algorithm*, International Journal of Computer and Electrical Engineering, **4**(4), pp. 463-466, 2012.

[17] Pal, B.B., Biswas, P. & Mukhopadhyay, A., *GA Based FGP Approach for Optimal Reactive Power Dispatch*, Procedia Technology, **10**, pp. 464-473, 2013.

[18] Alam, M.S. & De, M., *Optimal Reactive Power Dispatch Using Hybrid Loop-Genetic Based Algorithm*, in Proceedings of the 2016 19th National Power Systems Conference (NPSC), pp. 1-6, Bhubaneswar, India, December 2016.

[19] Rayudu, K., Yesuratnam, G. & Jayalaxmi, A., *Improving Voltage Stability by Optimal Reactive Power Dispatch Based on Genetic Algorithm and Linear Programming Technique*, in Complexity 27 Proceedings of the 2016 International Conference on Electrical, Electronics, and Optimization Techniques, ICEEOT 2016, pp. 1357-1362, March 2016.

[20] Ela, A.A.A.E., Abido, M.A. & Spea, S.R., *Differential Evolution Algorithm for Optimal Reactive Power Dispatch*, Electric Power Systems Research, **81**(2), pp. 458-464, 2011.

[21] Singh, H. & Srivastava, L., *Modified Differential Evolution Algorithm for Multi-Objective VAR Management*, International Journal of Electrical Power & Energy Systems, **55**, pp. 731– 740, 2014.

[22] Basu, M., *Quasi-Oppositional Differential Evolution for Optimal Reactive Power Dispatch*, International Journal of Electrical Power & Energy Systems, **78**, pp. 29-40, 2016.

[23] Basu, M., *Multi-Objective Optimal Reactive Power Dispatch Using Multi-Objective Differential Evolution*, International Journal of Electrical Power & Energy Systems, **82**, pp. 213-224, 2016.

[24] Li, Y., Li, X. & Li, Z., *Reactive Power Optimization Using Hybrid CABC-DE Algorithm*, Electric Power Components and Systems, **45**(9), pp. 980-989, 2017.

[25] Kumari, M. & Sydulu, M., *Improved Particle Swarm Algorithm Applied to Optimal Reactive Power Control*, in Proceedings of the 2006 IEEE International Conference on Industrial Technology, pp. 1873-1878, Mumbai, India, December 2006.

[26] Cai, G., Ren, Z. & Yu, T., *Optimal Reactive Power Dispatch Based on Modified Particle Swarm Optimization Considering Voltage Stability*, in Proceedings of the 2007 IEEE Power Engineering Society General Meeting, pp. 1-5, Tampa, FL, USA, June 2007.

[27] Mahadevan, K. & Kannan, P.S., *Comprehensive Learning Particle Swarm Optimization for Reactive Power Dispatch* Applied Sof Computing, **10**(2), pp. 641-652, 2010.

[28] Polprasert, J., Ongsakul, W. & Dieu, V.N., *Optimal Reactive Power Dispatch Using Improved Pseudo-Gradient Search Particle Swarm Optimization*, Electric Power Components and Systems, **44**(5), pp. 518-532, 2016.

[29] Mehdinejad, M., Mohammadi-Ivatloo, B., DadashzadehBonab, R. & Zare, K., *Solution of Optimal Reactive Power Dispatch of Power Systems Using Hybrid Particle Swarm Optimization and Imperialist Competitive Algorithms*, International Journal of Electrical Power & Energy Systems, **83**, pp. 104-116, 2016.

[30] Naderi, E., Narimani, H., Fathi, M. & Narimani, M.R., *A Novel Fuzzy Adaptive Configuration of Particle Swarm Optimization to Solve Large-Scale Optimal Reactive Power Dispatch*, Applied Soft Computing, **53**, pp. 441-456, 2017.

[31] Ding, J., Zhang, Q. & Ma, Y., *Optimal Reactive Power Dispatch Based on The CS-PSO Algorithm*, in Proceedings of the 13th Conference on Industrial Electronics and Applications (ICIEA), IEEE, Wuhan, China, 2018.

[32] Morsali, J., Zare, K., Tarafdar Hagh, M., *Performance Comparison of TCSC with TCPS And SSSC Controllers in AGC of Realistic Interconnected Multi-Sources Power System*, Ain Shams Engineering Journal, **7**(1), pp. 143-158, 2016.

[33] Arifoğlu, U. & Yalçin, F., *System Constrained Active Power Loss Minimization in Practical Multi-terminal HVDC Systems through GA*, Sakarya University Journal of Science, 10.16984/saufenbilder.421351, (1-1), 2018.

[34] Wei, H., Lin, C. & Wang, Y., *The Optimal Reactive Power Flow Model in Mixed Polar Form Based on Transformer Dummy Nodes*. IEEJ Trans Elec Electron Eng., **13**, pp. 411-416, 2018.

[35] Fang, S, Cheng, H, Xu, G, Zhou, Q, He, H. & Zeng, P., *Stochastic Optimal Reactive Power Reserve Dispatch Considering Voltage Control Areas*, Int. Trans. Electr. Energ. Syst. 2017.

[36] Ghazavi Dozein, M., Monsef, H., Ansari, J. & Kazemi, A., *An Effective Decentralized Scheme to Monitor and Control the Reactive Power Flow: A Holonic-Based Strategy*, Int. Trans. Electr. Energ. Syst., **26**, pp. 1184-1209, 2016.

[37] Du, Z., Nie, Y. & Liao, P., *PCPDIPM-Based Optimal Reactive Power Flow Model Using Augmented Rectangular Coordinates*, Int. Trans. Electr. Energ. Syst., **24**, pp. 597-608, 2014.

[38] Liu, B., Liu, F., Zhai, B. & Lan, H., *Investigating Continuous Power Flow Solutions of IEEE 14-Bus System*. IEEJ Trans Elec Electron Eng., **14**, pp. 157-159, 2019.

[39] Soodi, H.A. & Vural A.M., *STATCOM Estimation Using Backpropagation, PSO, Shuffled Frog Leap Algorithm, and Genetic Algorithm Based Neural Networks*, Comput Intell Neurosci, **2018**(1), pp.1-17, 2018. DOI: 10.1155/2018/6381610

[40] Gagliano A. & Nocera F., *Analysis of The Performances of Electric Energy Storage in Residential Applications*, International Journal of Heat and Technology, **35**(1), pp. S41-S48, 2017, DOI: 10.18280/ijht.35Sp0106.

[41] Caldera, M., Ungaro, P., Cammarata, G. & Puglisi G., *Survey-Based Analysis of The Electrical Energy Demand in Italian Households*, Mathematical Modelling of Engineering Problems, **5**(3), pp. 217-224. DOI: 10.18280/mmep.050313

[42] Do, D.T.T. & Lee, J., A *Modified Symbiotic Organisms Search (Msos) Algorithm for Optimization of Pin-Jointed Structures*, Appl. Soft Comput., **61**, pp. 683-699, 2017.

[43] Panda, A. & Pani, S., *A Symbiotic Organisms Search Algorithm with Adaptive Penalty Function to Solve Multi-Objective Constrained Optimization Problems*, Appl. Soft Comput., **46**, pp. 344-360, 2016.

[44] Shiva, C.K. & Mukherjee, V., *A Novel Quasi-Oppositional Harmony Search Algorithm for Automatic Generation Control of Power System*, Appl. Soft Comput., **35**, pp. 749-765, 2015.

[45] Bouchekara, H.R.E.H., Zellagui, M. & Abido, M.A., *Optimal Coordination of Directional Overcurrent Relays Using A Modified Electromagnetic Field Optimization Algorithm*, Applied Soft Computing, **54**, pp. 267-283, 2017.

[46] Talebi, B. & Dehkordi, M.N., *Sensitive Association Rules Hiding Using Electromagnetic Field Optimization Algorithm*, Expert Systems with Applications, **114**, pp. 155-172, 2018.

[47] Becerra. & Cooray, V., *A Simplified Physical Model to Determine the Lightning Upward Connecting Leader Inception*, IEEE Trans. Power Deliv., **21**, pp. 897-908, 2006.

[48] Nematollahi, F., Rahiminejad, A. & Vahidi, B., *A Novel Physical Based Meta-Heuristic Optimization Method Known as Lightning Attachment Procedure Optimization*, Applied Soft Computing, **59**, pp. 596-621, 2017, DOI: 10.1016/j.asoc.2017.06.033.

[49] *Illinois Center for a Smarter Electric Grid (ICSEG)*, Available online: https://icseg.iti.illinois.edu/ieee-30-bussystem/ (accessed on 25 February 2019).

[50] El Ela, A.A., Abido, M.A. & Spea, S.R., *Differential Evolution Algorithm for Optimal Reactive Power Dispatch*, Electr. Power Syst. Res., **81**, pp. 458-464, 2011.

[51] Duman, S., Sönmez, Y., Güvenç, U. & Yörükeren, N., *Optimal Reactive Power Dispatch Using a Gravitational Search Algorithm*, IET Gener. Transm. Distrib., **6**, pp. 563-576, 2012.

[52] Aljohani, T.M., Ebrahim, A.F. & Mohammed, O., *Single and Multiobjective Optimal Reactive Power Dispatch Based on Hybrid Artificial Physics–Particle Swarm Optimization*, Energies, **12**, 2333, 2019.

[53] Dai, C., Chen, W., Zhu, Y. & Zhang, X., *Seeker Optimization Algorithm for Optimal Reactive Power Dispatch*, IEEE T. Power Syst., **24**(3), pp. 1218-1231, 2009.

[54] Subbaraj, P. & Rajnarayan, P.N., *Optimal Reactive Power Dispatch Using Self-Adaptive Real Coded Genetic Algorithm*. Electr. Power Syst. Res., **79**(2), pp. 374-38, 2009.

[55] Pandya, S. & Roy, R., *Particle Swarm Optimization Based Optimal Reactive Power Dispatch*. Proceeding of the IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT), pp. 1-5.

[56] Hussain, A.N., Abdullah, A.A. & Neda, O.M., *Modified Particle Swarm Optimization for Solution of Reactive Power Dispatch*, Research Journal of Applied Sciences, Engineering and Technology, **15**(8), pp. 316-327, 2018 DOI:10.19026/rjaset.15.5917.

[57] Reddy, S.S., *Optimal Reactive Power Scheduling Using Cuckoo Search Algorithm*, International Journal of Electrical and Computer Engineering, **7**(5), pp. 2349-2356. 2017.

[58] Reddy, S.S., *Faster Evolutionary Algorithm Based Optimal Power Flow Using Incremental Variables*, Electrical Power and Energy Systems, **54**, pp. 198-210, 2014.