American Journal of Engineering and Technology Management
2026, Vol. 11, No. 3, pp. 31–41
https://doi.org/10.11648/j.ajetm.20261103.11

Research Article

Improved Driving Training-Based Optimization Algorithm Using Levy Flight and Crowding Distance Techniques for Solving Optimal Power Flow Problem

Edmond Randriamora*, Olivier Mickael Ranarison, Rivo Mahandrisoa Randriamaroson

Doctoral School of Science and Engineering Techniques and Innovation, University of Antananarivo, Antananarivo, Madagascar

Abstract

Driving Training-Based Optimization (DTBO) algorithm is a metaheuristic algorithm based on the simulation of driving training process. Improved version of the DTBO is proposed in this paper for solving Optimal Power Flow (OPF) problem. The Improved Driving Training-Based Optimization (IDTBO) algorithm includes the Crowding Distance Technique for more diverse driver and learner selection and incorporates the Levy Flight distribution for better exploration and local optima avoidance. OPF is considered as one of the most difficult optimization problems and is very important for the control of electrical network. The objective of this study is finding the best control variables while minimizing the total generation fuel cost and considering equality and inequality constraints of the system. The standard IEEE 30-bus network is used for evaluating the performance of the IDTBO algorithm for solving OPF problem. For solving conventional power flow equation, Newton Raphson algorithm is considered. Compared to Modified Driving Training-Based Optimization (MDTBO), Teaching Learning-Based Optimization (TLBO) and Particle Swarm Optimization (PSO) algorithms, the proposed method is more accurate and is better in convergence speed. The performance of the IDTBO is very useful for finding the most secure operating point of any electric power system and its convergence speed contributes to improving the dynamic management of a smart electricity grid.

Keywords

Electrical Network, Optimal Power Flow, Improved Driving Training-Based Optimization, Levy Flight, Crowding Distance, Fuel Cost, Newton Raphson Method

1. Introduction

Power flow study consists of measuring the active and reactive power at each bus of the network, while the goal of OPF problem is to optimize an objective function such as generation fuel cost by adjusting power system control variables which are the generator real powers, the generator bus voltages, the transformer tap settings and the reactive power of switchable VAR sources. The ideal power flow becomes a large-scale, highly constrained nonlinear and nonconvex optimization problem since each control variable has a limit.

In the literature, there are several methods for calculating

*Correspondence: Edmond Randriamora (edmondrandriamora@gmail.com)

Received: 20 April 2026; **Accepted:** 3 May 2026; **Published:** 14 May 2026

power flow such as Newton Raphson (1960), Gauss Seidel (1874), Brown (1971), Tensor (1972), Fast-decoupled (1974), Iwamoto (1978), quadratic Newton (1980), Integral State Estimation (1983), Runge-Kutta (1986), Generalized Minimal Residual (1993), continuous version of Newton (1997), Newton with partial regularization using Singular Value Decomposition (2007) and Levenberg Marquardt (2008) [1]. For the OPF problem, Mohamed Ebeed and al. (2018) identified 5 conventional optimization methods including linear programming, nonlinear programming, quadratic programming, Newton's Method and interior point method, and 89 recent optimization methods including 27 nature inspired algorithms, 11 human inspired algorithms, 6 physics inspired algorithms, 25 evolutionary inspired algorithms, 13 hybrid inspired algorithms and 7 artificial neural networks and fuzzy logic approach [2]. Emmanuel and al. (2021) also identified 39 swarm and bio-inspired optimization techniques, 11 human-inspired optimization techniques, 4 physic-inspired optimization techniques, 6 evolutionary-inspired optimization techniques, 5 artificial neural networks and fuzzy logic approach, and 12 hybrid optimization techniques [3]. PSO is proposed by M. Abido (2002) for solving OPF in IEEE 30-bus with as objective functions: fuel cost minimization, piecewise cost, voltage profile improvement and voltage stability enhancement [4]. Arul Ponnusamy and al. (2014) used Cuckoo Search Algorithm (CSA) for OPF solution in IEEE 62-Indian utility system with minimization of fuel cost function [5]. Bouchekara et al. (2014) present TLBO for IEEE 30-bus and 118-bus with quadratic fuel cost, piecewise quadratic cost, voltage stability, voltage profile, and active power transmission losses as objective functions in the OPF issue [6]. Recently, O. M. Ranarison (2025) proposed a Modified Driving Training Based Opitmization algorithm for solving OPF problem with five objective functions: minimization of fuel cost, voltage profile improvement, voltage stability enhancement, minimization of active and reactive power losses, applied in IEEE 30-bus system [7]. According to the survey of Levy Flight-Based Metaheuristics for Optimization by Juan Li and al. (2022), thirteen thousand Levy flight-related studies have been published in journals/dissertations/conferences up to 23 April 2022 since Levy flight was proposed in 1981. 13% of these relate to the field of engineering, but no precision for Electrical engineering [8]. However, the state of the art proves that there are some metaheuristic algorithms using Levy Flight for OPF solution. Edmond and al. (2017) proposed OPF by Cuckoo Search via Levy Flight algorithm on a standard IEEE 30-bus with fuel costs minimization, voltage profiles improvement and piecewise quadratic cost curve; results are compared to PSO and Differential Evolution (DE) algorithms [9]. An improved TLBO algorithm (ITLBO) using Levy mutation strategy for non-smooth OPF is also proposed by Ghasemi and al. (2015), tested on IEEE 30-bus and 57-bus systems with voltage stability, emission minimization, generation cost, quadratic cost with valve-point effect and piecewise quadratic cost, as objective functions [10]. K. Lenin (2018) used crowding distance based particle swarm optimization algorithm (CDPSO) for solving optimal reactive power dispatch problem with minimization of real power loss and minimization of voltage deviation, tested in IEEE 30-bus network [11].

In this paper, Improved version of Driving Training-Based Optimization algorithm is proposed, using Levy Flight and Crowding Distance techniques. Objective function is to minimize fuel cost generation while considering constraints of the IEEE 30-bus system. For the numerical calculation of power flow, the Newton-Raphson method is considered.

2. Power Flow

2.1. Mathematical Modeling of Power Flow

As shown in Figure 1, each bus of electrical network is characterized by 4 variables which are: P_i (active power injected), Q_i (reactive power injected), V_i (magnitude voltage), θ_i (voltage angle), "i" is the bus number. The powers generated at the bus i are denoted by $P_{G\,i}$ and $Q_{G\,i}$, and powers load by $P_{L\,i}$ and $Q_{L\,i}$. The active and reactive powers injected at the bus i are given by:

$$P_i = P_{G\,i} - P_{L\,i} + \sum_j P_{ij} \tag{1}$$

$$Q_i = Q_{G\,i} - Q_{L\,i} + \sum_j Q_{ij} \tag{2}$$

P_{ij} and Q_{ij} are power transmitted in the line ij and given by:

$$P_{ij} = Real\{\overline{S}_{ij}\} \tag{3}$$

$$Q_{ij} = -Imag\{\overline{S}_{ij}\} \tag{4}$$

Apparent power of line ij is:

$$\overline{S}_{ij} = \overline{V}_i\,\overline{I}_{ij}^{\,*} \tag{5}$$

With:

$$\overline{I}_{ij} = -(\overline{V}_i - \overline{V}_j)\overline{Ybus_{ik}} \tag{6}$$

$\overline{Ybus_{ik}}$ is the bus admittance of branch i and k.

Depending on the case, a bus can be of type PQ bus, PV bus or Slack bus. There is at least one slack bus per network where the most powerful power plant is associated. The description of each type of bus is given in Table 1. The general power equations of bus i are then given by:

$$P_i = Real\left\{\overline{S}_i^{\,*}\right\} \tag{7}$$

$$Q_i = -Imag\left\{\overline{S}_i^{\,*}\right\} \tag{8}$$

Where $\overline{S_i}^*$ is the conjugate apparent power of bus i, given by:

$$\overline{S_i}^* = P_i - jQ_i = \overline{V_i}^* \overline{I_i} = \overline{V_i}^* \sum_{k=1}^{n} \overline{Ybus_{ik}} \overline{V}_k \quad (9)$$

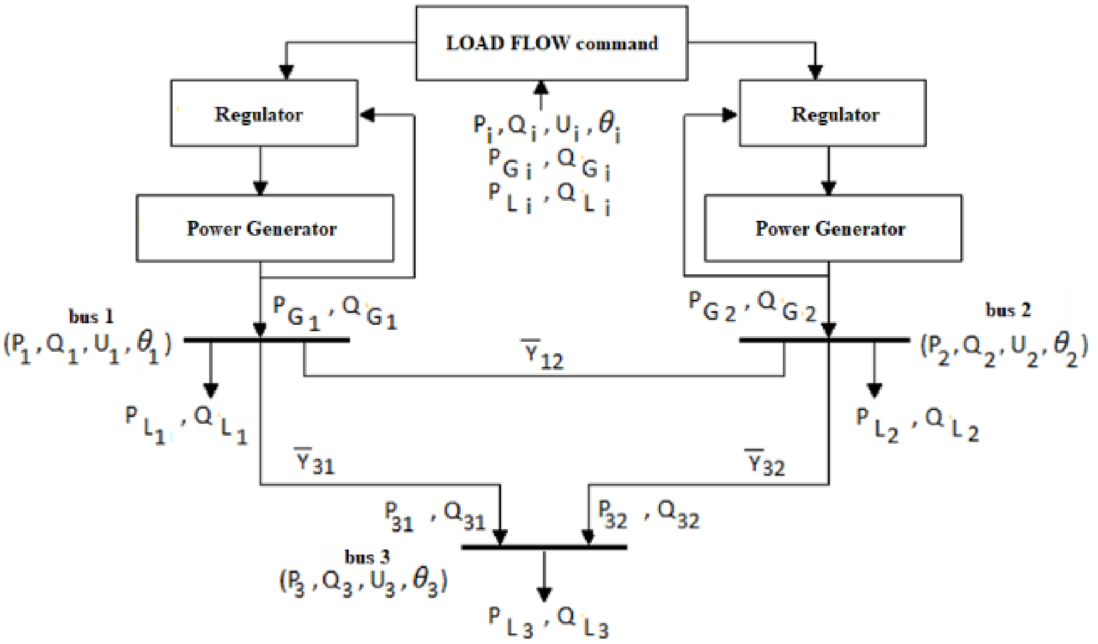

Figure 1. *Power control in an electrical network.*

Table 1. Bus classification.

Bus Type	Specified variables	Desired variables
PQ bus	$P,\ Q$	$V,\ \theta$
PV bus	$P,\ V$	$Q,\ \theta$
Slack bus	$V,\ \theta$	$P,\ Q$

2.2. Newton-Raphson's Method

Newton-Raphson's method is iterative and consists, at each step, of using the intersection between the x-axis and the tangent to the curve at the previous point (Figure 2).

$$x^{k+1} = x^k - \frac{f(x^k)}{f'(x^k)} \quad (10)$$

For n the total number of bus in the network, the power equations are:

$$\begin{cases} P_i = P_i(V_1, V_2, \dots, V_n, \theta_1, \theta_2, \dots, \theta_n) \\ Q_i = Q_i(V_1, V_2, \dots, V_n, \theta_1, \theta_2, \dots, \theta_n) \end{cases} \text{with } i = 1, \dots, n \quad (11)$$

The power residuals obtained according to Newton-Raphson's method are:

$$\begin{cases} \Delta P_i^k = \sum_{j=1}^{n} \left(\frac{\partial P_i^k}{\partial \theta_j} \Delta\theta_j^k + \frac{\partial P_i^k}{\partial V_j} \Delta V_j^k \right) \\ \Delta Q_i^k = \sum_{j=1}^{n} \left(\frac{\partial Q_i^k}{\partial \theta_j} \Delta\theta_j^k + \frac{\partial Q_i^k}{\partial V_j} \Delta V_j^k \right) \end{cases} \quad (12)$$

$$\begin{cases} \Delta P_i^k = P_i^{k+1} - P_i^k = P_i^{plan} - P_i^k \\ \Delta Q_i^k = Q_i^{k+1} - Q_i^k = Q_i^{plan} - Q_i^k \\ \Delta\theta_j^k = \theta_j^{k+1} - \theta_j^k \\ \Delta V_j^k = V_j^{k+1} - V_j^k \end{cases} \quad (13)$$

With P_i^{plan}, Q_i^{plan} are the planned values, and P_i^k, Q_i^k are the calculated values.

Hence the following matrix equation:

$$\begin{bmatrix} [\Delta P^k] \\ [\Delta Q^k] \end{bmatrix} = [J^k] \begin{bmatrix} [\Delta\theta^k] \\ [\Delta V^k] \end{bmatrix} \quad (14)$$

$$\begin{bmatrix} [\Delta\theta^k] \\ [\Delta V^k] \end{bmatrix} = [J^k]^{-1} \begin{bmatrix} [\Delta P^k] \\ [\Delta Q^k] \end{bmatrix} \quad (15)$$

With $[J^k]$ is the Jacobian given by:

$$[J^k] = \begin{bmatrix} \frac{\partial P_1^k}{\partial \theta_1} & \cdots & \frac{\partial P_1^k}{\partial \theta_n} & \frac{\partial P_1^k}{\partial V_1} & \cdots & \frac{\partial P_1^k}{\partial V_n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial P_n^k}{\partial \theta_1} & \cdots & \frac{\partial P_n^k}{\partial \theta_n} & \frac{\partial P_n^k}{\partial V_1} & \cdots & \frac{\partial P_n^k}{\partial V_n} \\ \frac{\partial Q_1^k}{\partial \theta_1} & \cdots & \frac{\partial Q_1^k}{\partial \theta_n} & \frac{\partial Q_1^k}{\partial V_1} & \cdots & \frac{\partial Q_1^k}{\partial V_n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial Q_n^k}{\partial \theta_1} & \cdots & \frac{\partial Q_n^k}{\partial \theta_n} & \frac{\partial Q_n^k}{\partial V_1} & \cdots & \frac{\partial Q_n^k}{\partial V_n} \end{bmatrix} \quad (16)$$

Finally, results are obtained by:

$$\begin{cases} \theta_i^{k+1} = \theta_i^k + \Delta\theta_i^k \\ V_i^{k+1} = V_i^k + \Delta V_i^k \end{cases} i = 1, \dots, n \quad (17)$$

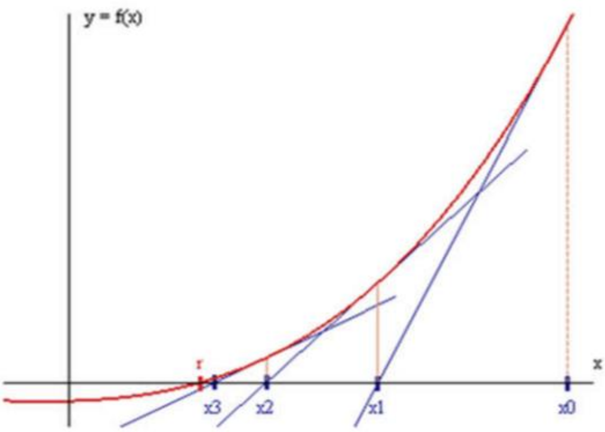

Figure 2. Iterative method of Newton-Raphson.

3. Optimal Power Flow

3.1. Mathematical Modeling of Optimal Power Flow

Let x be a vector of dependent variables and u a vector of control variables:

$$x = [P_{G1}, V_{L1} \dots V_{L,NPQ}, Q_{G1} \dots Q_{G,NG}, S_{TL,1} \dots S_{TL,NTL}] \quad (18)$$

$$u = [P_{G,2} \dots P_{G,NG}, V_{G,1} \dots V_{G,NG}, Q_{C,1} \dots Q_{C,NC}, T_1 \dots T_{NT}] \quad (19)$$

P_{G1}: active power output at slack bus;
V_L: voltage magnitude at PQ buses;
Q_G: reactive power output of all generator units;
S_{TL}: transmission line loading;
NPQ: number of load buses;
NG: number of generator units;
NTL: number of transmission lines
P_G: active power generation at the PV buses except at the slack bus;
V_G: voltage magnitude at PV buses;
Q_C: shunt VAR compensation;
T: tap settings of transformer;
NC: number of VAR compensators;
NT: number of regulating transformers

By minimizing an objective function F made up of x and u, the OPF problem can be modeled as:

$$Min\ F(x, u) \quad (20)$$

The objective function for minimizing the total generation fuel cost can be expressed as:

$$F_1 = \sum_{i=1}^{NG} F_i(P_{Gi}) = \sum_{i=1}^{NG} (a_i + b_i P_{Gi} + c_i P_{Gi}^2) \quad (21)$$

Where a_i, b_i, c_i are the cost coefficients of ith generator.

The equilibrium between generated and load powers is represented by the equality constraints:

$$P_{Gi} - P_{Li} = 0 \quad (22)$$

$$Q_{Gi} - Q_{Li} = 0 \quad (23)$$

Inequality constraints are security constraints that are closely related to system limitations.

$$P_{Gi}^{min} \le P_{Gi} \le P_{Gi}^{max}\ i = 1,2, \dots, NG \quad (24)$$

$$V_{Gi}^{min} \le V_{Gi} \le V_{Gi}^{max}\ i = 1,2, \dots, NG \quad (25)$$

$$Q_{Gi}^{min} \le Q_{Gi} \le Q_{Gi}^{max}\ i = 1,2, \dots, NG \quad (26)$$

$$T_i^{min} \le T_i \le T_i^{max}\ i = 1,2, \dots, NT \quad (27)$$

$$Q_{Ci}^{min} \le Q_{Ci} \le Q_{Ci}^{max}\ i = 1,2, \dots, NC \quad (28)$$

$$S_{Li} \le S_{Li}^{min}\ i = 1,2, \dots, NTL \quad (29)$$

$$V_{Li}^{min} \le V_{Li} \le V_{Li}^{max}\ i = 1,2, \dots, NPQ \quad (30)$$

3.2. Driving Training-Based Optimization Algorithm

DTBO is a human-based metaheuristic algorithm, for solving optimization problems on the base of simulation of driving training process, developed by Mohammad Dehghani and al. (2022) [12]. Main idea is presented in Figure 3 and explained as follow: *"Driving training is an intelligent process in which a learner driver is trained and acquires driving skills. Learner driver can choose from several instructors when attending driving school. The instructor then teaches the learner driver the instructions and skills. The learner driver tries to learn driving skills from the instructor and drive following the instructor. In addition, personal practice can improve the driver's skills of the learner.*

These interactions and activities have extraordinary potential for designing an optimizer".

Figure 3. *Main idea of DTBO.*

Driving students and instructors make up the DTBO population, where size N is manually chosen. The population matrix models the OPF problem, and DTBO members represent potential solutions:

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_i \\ \vdots \\ X_N \end{bmatrix}_{N\times 1} = \begin{bmatrix} x_{11} & \cdots & x_{1j} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i1} & \cdots & x_{ij} & \cdots & x_{im} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{Nj} & \cdots & x_{Nm} \end{bmatrix}_{N\times m} \quad (31)$$

m is the number of problem variables, X_i is the ith candidate solution, x_{ij} is the value of the jth variable determined by the ith candidate solution and is randomly initialized using following equation:

$$x_{ij} = lb_j + r \cdot (ub_j - lb_j),$$

$$i = 1,2,\dots,N, j = 1,2,\dots,m \quad (32)$$

lb_j and ub_j are respectively the lower and upper bounds of the jth variable; r is a random number from [0, 1].

Every candidate is put into the objective function, and the population's best member is the one with the best value (X_{best}).

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_N \end{bmatrix}_{N\times 1} = \begin{bmatrix} F(X_1) \\ \vdots \\ F(X_i) \\ \vdots \\ F(X_N) \end{bmatrix}_{N\times 1} \quad (33)$$

DTBO is modelled in three different phases:

Phase 1: Training by the driving instructor (exploration)

The learner driver selects the driving teacher in the first step, and the chosen instructor then instructs the learner driver in driving. It can be stated as:

$$DI = \begin{bmatrix} DI_{11} & \cdots & DI_{1j} & \cdots & DI_{1m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ DI_{i1} & \cdots & DI_{ij} & \cdots & DI_{im} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ DI_{N_{DI}1} & \cdots & DI_{N_{DI}j} & \cdots & DI_{N_{DI}m} \end{bmatrix}_{N_{DI}\times m} \quad (34)$$

The top participants are regarded as driving teachers in each iteration *t*, while the remaining members are classified as trainee drivers. Using the following formula, the number of driving instructors N_{DI} decreases with each iteration:

$$N_{DI} = 0.1 \cdot N \cdot (1 - t/T) \quad (35)$$

T is the maximum number of iterations.

The following formula is used to determine each member's new position:

$$x_{i,j}^{P1} = \begin{cases} x_{i,j} + r \cdot \left(DI_{k_{i,j}} - I \cdot x_{i,j}\right), F_{DI_{k_i}} < F_i \\ x_{i,j} + r \cdot \left(x_{i,j} - DI_{k_{i,j}}\right), otherwise. \end{cases} \quad (36)$$

I is a number randomly selected from the set [1, 2], r is a random number in the interval [0, 1].

If this new position increases the objective function's value, it replaces the old one.

$$X_i = \begin{cases} X_i^{P1}, si\ F_i^{P1} < F_i \\ X_i, otherwise \end{cases} \quad (37)$$

Phase 2: Patterning of the instructor skills of the student driver (exploration)

The learner driver mimics the instructor in the second phase. The DTBO's exploration power is increased by this process, which shifts the population to various locations throughout the search space.

$$x_{i,j}^{P2} = P \cdot x_{i,j} + (1 - P) \cdot DI_{k_{i,j}} \quad (38)$$

$$X_i = \begin{cases} X_i^{P2}, si\ F_i^{P2} < F_i \\ X_i, otherwise \end{cases} \quad (39)$$

P is the patterning index given by:

$$P = 0.001 + 0.9(1 - t/T) \quad (40)$$

Phase 3: Personal practice (exploitation)

This third stage is built on each student driver's individual practice to develop and strengthen their driving abilities.

$$x_{i,j}^{P3} = x_{i,j} + (1 - 2r) \cdot R \cdot (1 - t/T) \cdot x_{i,j} \quad (41)$$

$$X_i = \begin{cases} X_i^{P3}, si\ F_i^{P3} < F_i \\ X_i, otherwise \end{cases} \quad (42)$$

r is a random real number of the interval [0, 1], R is the

constant set to 0.05.

3.3. Improved Driving Training-Based Optimization Algorithm

IDTBO algorithm using Levy Flight and Crowding Distance Techniques are recently proposed by Daniel Kwegyir and al. (2024) [13]: "*The choice of learners and drivers in the original DTBO process can significantly impact the algorithm's accuracy. In the DTBO algorithm, drivers are the members used to produce new solutions in each iteration. If they are poorly chosen, the DTBO algorithm's chances of finding optimal solutions are minimal. Furthermore, if the learners are well chosen, there is a higher chance that the algorithm will converge to a reasonable solution within the search space*".

Implementation of Crowding Distance Technique:

A solution's similarity to its neighbors is measured by its crowding distance [14, 15]. The number of drivers chosen in equation (35) is always 10% of the entire population or less. Because many candidate members are left as learners and only a small number of the solution's fit members are chosen as drivers, this makes it difficult for the algorithm to produce high-quality solutions and avoid becoming trapped in local optima. The drivers and learners are chosen in crowding distance selection in order to maximize the average crowding distance of the chosen solutions.

First, arrange everyone in the population in order of best to worst solution. F_{max} and F_{min} are respectively the objective value of the best and worst solution.

$$F = \begin{bmatrix} F(X_{best}) \\ \vdots \\ F(X_i) \\ \vdots \\ F(X_{worst}) \end{bmatrix}_{N\times 1} = \begin{bmatrix} F_{max} \\ \vdots \\ F_i \\ \vdots \\ F_{min} \end{bmatrix}_{N\times 1} \quad (43)$$

Next, determine each member's crowding distance:

$$d_i = \frac{F_{i+1} - F_{i-1}}{F_{max} - F_{min}} \quad (44)$$

In the subsequent algorithmic iteration, choose the top half of the population with the greatest crowding distance to be drivers and the other half to be learners.

Implementation of Levy Flight distribution:

Equation (32) defines the random distribution that was frequently utilized to initiate the population in the original DTBO. However, the method can search outside the immediate search space of initial solutions and possibly uncover superior solutions if Levy Flight is used to initialize the solution [16-18]. Additionally, by avoiding local optima, Levy Flight prevents the algorithm from prematurely converging to subpar answers. The formula to produce an initial random solution is as follows, where A and D represent the number of search agents and the problem's dimension, respectively:

$$x_{ij} = lb_j + \left(ub_j - lb_j\right) \cdot rand(A, D) \quad (45)$$

Next, determine the Levy distribution's step size using:

$$step\ size = \frac{1}{\sqrt{D}} \quad (46)$$

Next, create a Cauchy number at random using the Cauchy distribution. The typical probability distribution function is applied, using scale value 1 and location parameter 0:

$$f(x) = \frac{1}{\pi(1+x^2)} \quad (47)$$

Levy number is given by:

$$levy\ number = step\ size\ \times Cauchy\ number \quad (48)$$

Lastly, the answer is scaled down using the Levy flight:

$$X_{ij} = x_{ij} + rand(A, D) \cdot levy(A) \quad (49)$$

Flowchart of IDTBO is presented in Figure 4.

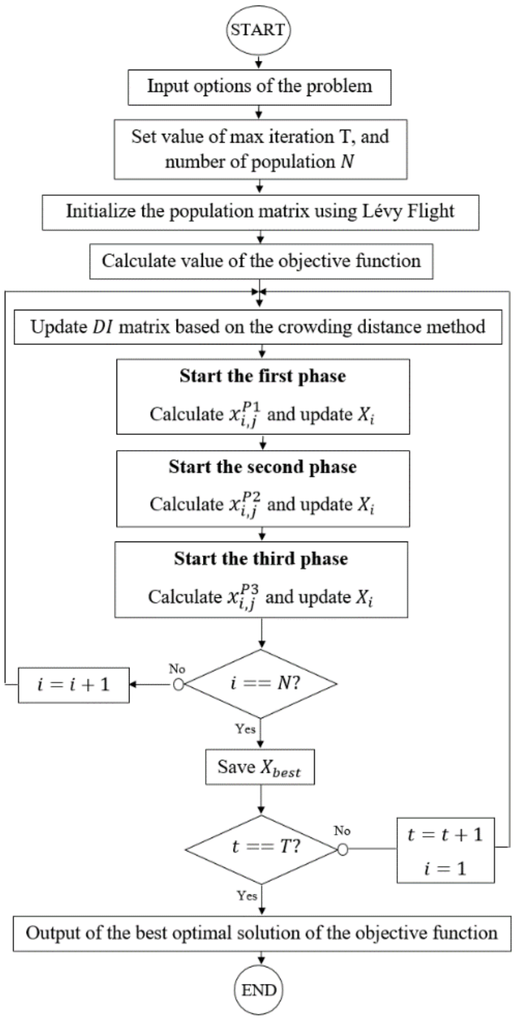

***Figure 4.** Flowchart of IDTBO.*

4. IEEE 30 Bus Network

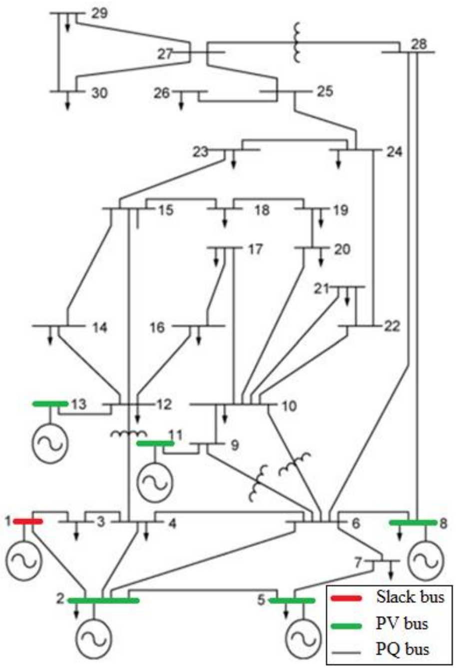

Figure 5. *IEEE 30-bus system.*

Six power generators (buses 1, 2, 5, 8, 11, and 13), four transformers with an off-nominal tap ratio (lines 6-9, 6-10, 4-12, and 28-27), and nine shunt VAR compensation buses (buses 10, 12, 15, 17, 20, 21, 23, 24, and 29) make up the IEEE 30-bus system (Figure 5). Cost coefficients, bus data, generator data, and line data are provided in [19, 20]. The control variable and line power transmission minimum and maximum limits are provided in [7].

Solution of OPF with IDTBO algorithm for minimizing fuel cost is given in Table 2. Results are compared with Particle Swarm Optimization (PSO), Teaching Learning-Based Optimization (TLBO) and Modified Driving Training-Based Optimization (MDTBO) algorithms.

The MATLAB code for IDTBO is executed with the same machine as the three others algorithms, with more interesting elapsed time which is equal to 149s (15% improved compared to 177s with MDTBO, and 50.9% improved compared to 305s with TLBO). The PSO's convergence speed remains the best, but with a less optimized result (799.5823$/h) compared to that obtained with IDTBO (799.2659$/h). Furthermore, compared to MDTBO (799.5753$/h), the cost is more attractive with the new method. In Figure 6, IDTBO also represents an interesting fuel cost variation.

Table 2. *Optimal settings of control variables with PSO, TLBO, MDTBO and IDTBO.*

Variables	Min	Max	Initial case	PSO	TLBO	MDTBO	IDTBO
P_1	50	200	99.2225	177.3006	177.0567	176.9336	176.6836
P_2	20	80	80	48.7654	48.6972	48.6810	48.7035
P_5	15	50	50	21.3155	21.3043	21.2255	21.2130
P_8	10	35	20	20.8126	21.0814	20.9396	20.9662
P_{11}	10	30	20	11.8358	11.8842	11.7209	12.0235
P_{13}	12	40	20	12.1574	12.000	12.6520	12.4609
V_1	0.95	1.1	1.0500	1.1000	1.1000	1.1000	1.1000
V_2	0.95	1.1	1.0400	1.0873	1.0879	1.0876	1.0882
V_5	0.95	1.1	1.0100	1.0613	1.0617	1.0608	1.0614
V_8	0.95	1.1	1.0100	1.0695	1.0694	1.0689	1.0701
V_{11}	0.95	1.1	1.0500	1.0999	1.1000	1.1000	1.0995
V_{13}	0.95	1.1	1.0500	1.0999	1.1000	1.1000	1.0998
T_{11}	0.9	1.1	1.0780	0.9902	1.0447	0.9497	0.9595
T_{12}	0.9	1.1	1.0690	1.0436	0.9000	1.0125	1.0261
T_{15}	0.9	1.1	1.0320	1.0999	0.9863	1.0177	1.0104
T_{36}	0.9	1.1	1.0680	1.0123	0.9657	0.9717	0.9733
Q_{10}	0.0	5.0	0	0.0018	5.000	2.2939	3.1194

Variables	Min	Max	Initial case	PSO	TLBO	MDTBO	IDTBO
Q_{12}	0.0	5.0	0	4.8882	5.000	1.5445	2.3485
Q_{15}	0.0	5.0	0	1.4461	5.000	2.1016	1.6498
Q_{17}	0.0	5.0	0	4.9987	5.000	2.4704	4.9998
Q_{20}	0.0	5.0	0	1.8570	5.000	0.8641	4.4746
Q_{21}	0.0	5.0	0	0.0004	5.000	3.6410	4.9998
Q_{23}	0.0	5.0	0	4.9983	3.8490	1.7289	4.8488
Q_{24}	0.0	5.0	0	3.1354	5.000	0.77854	2.3917
Q_{29}	0.0	5.0	0	4.9907	2.7434	1.0707	1.9906
Cost ($/h)			901.9501	799.5823	799.0680	799.5753	799.2659
P_{loss}(MW)			5.8225	8.7880	8.6245	8.7532	8.6514
Q_{loss}(MVAR)			-4.6063	3.2444	4.1827	5.0820	4.4967
V_D			1.1496	1.0757	1.8583	1.4079	1.5887
Lmax			0.1723	0.1270	0.1164	0.1268	0.1214
Elapsed time				107.9159	305.0494	177.6250	149.7006

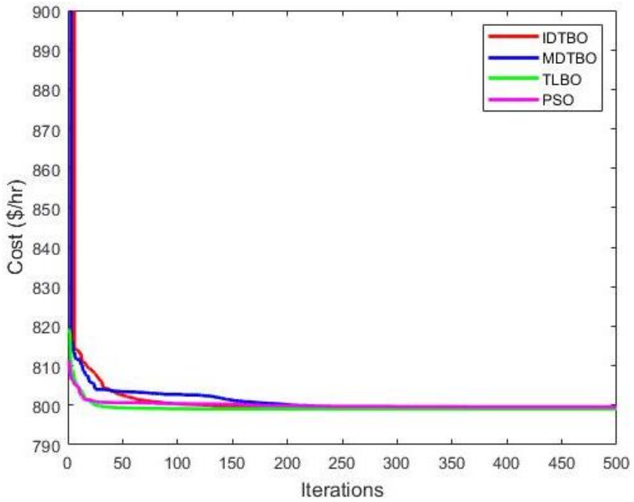

Figure 6. *Fuel cost variation with IDTBO, MDTBO, TLBO and PSO.*

In the first instance, there were some voltage breaches at buses 19 through 30 (below the minimum 0.95 p.u.), but the IDTBO results show that there are no longer any violations (Figure 7).

Active power flow and losses through transmission line is given in Figure 8. Active power losses are more interesting with IDTBO (a total of 8.6514 MW) compared to MDTBO (a total of 8.7532 MW). Similarly, the reactive power losses is greatly improved with IDTBO (a total of 4.4967 MVAR) compared to MDTBO (5.0820 MVAR). Reactive power flow and losses through transmission line is given in Figure 9.

Considering the imposed inequality constraints, including the limit on the power transmitted through the lines, solving OPF with IDTBO represents no violation with the considered objective function. As shown is Figure 10, there is still a significant margin before the power limits are reached.

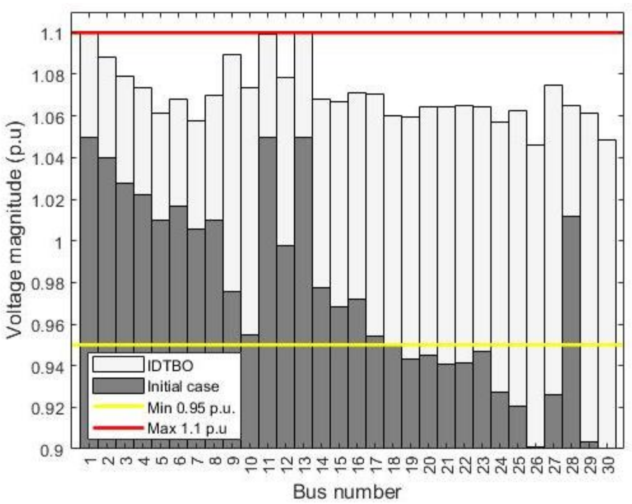

Figure 7. *System voltage profiles for initial case and IDTBO.*

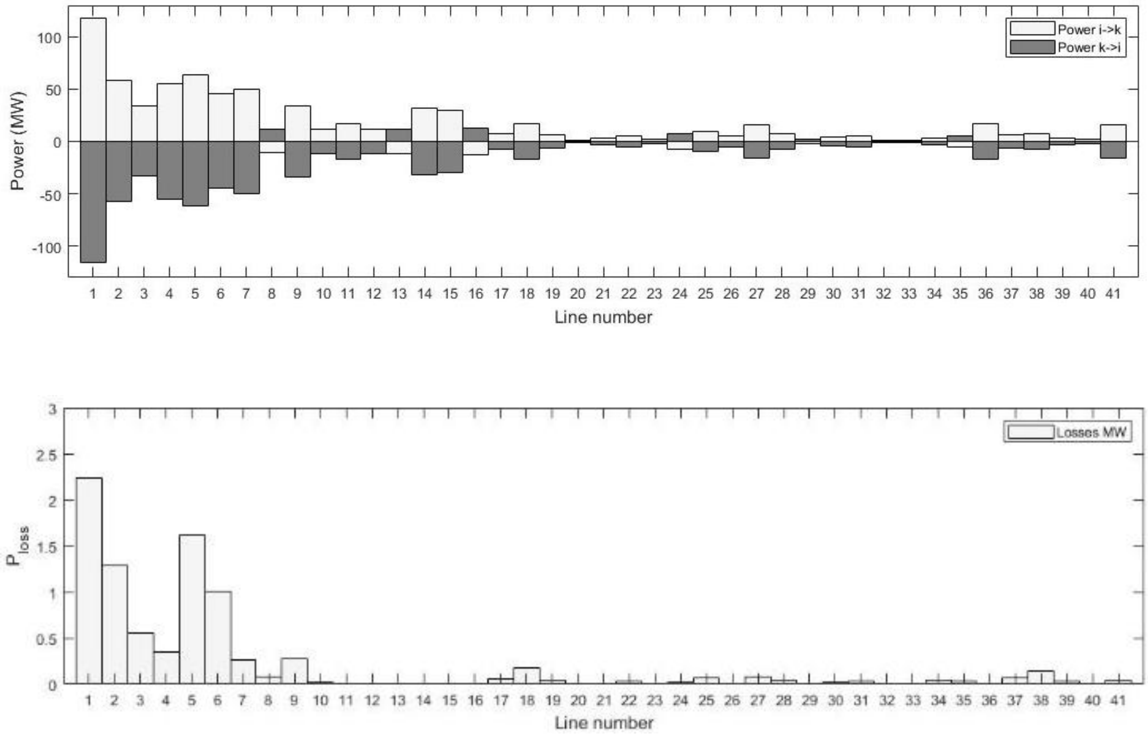

Figure 8. Active power flow and losses through transmission lines with IDTBO.

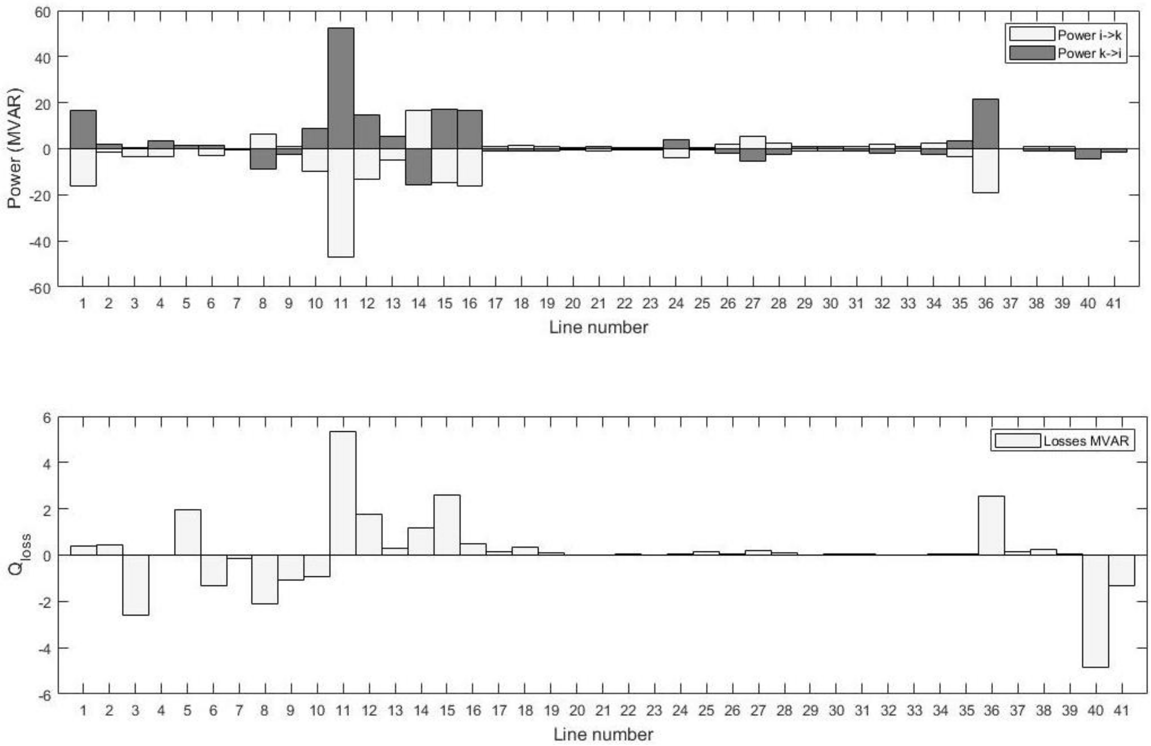

Figure 9. Reactive power flow and losses through transmission lines with IDTBO.

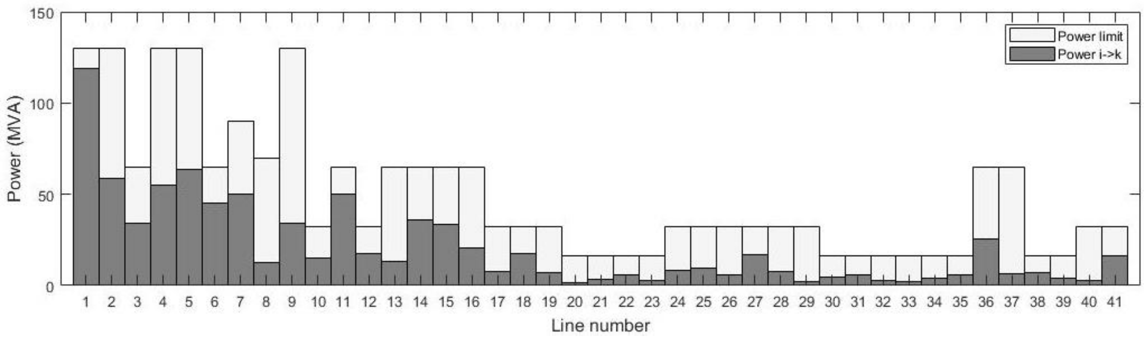

Figure 10. *Apparent power flow through transmission lines with IDTBO.*

5. Conclusions

In this paper, an enhanced version of the DTBO algorithm has been proposed for solving OPF problem. IDTBO introduces two critical enhancing in the original DTBO: first, drivers and learners are chosen using the crowding distance technique, second, initialization phase incorporates the Levy Flight distribution. These improved the algorithm's convergence speed, diversity, accuracy and aid in escaping local optima entrapment. With minimization of fuel cost as objective function, application's results of IDTBO in IEEE 30-bus network are compared to three metaheuristic algorithm: MDTBO, TLBO and PSO. The results show that IDTBO is more interesting with the best optimized value and the best convergence speed.

Abbreviations

OPF	Optimal Power Flow
DTBO	Driving Training-Based Optimization
MDTBO	Modified Driving Training-Based Optimization
IDTBO	Improved Driving Training-Based Optimization
TLBO	Teaching Learning-Based Optimization
PSO	Particle Swarm Optimization
CSA	Cuckoo Search Algorithm
IEEE	Institute of Electrical and Electronics Engineers
DE	Differential Evolution
ITLBO	Improved Teaching Learning-Based Optimization
CDPSO	Crowding Distance Based Particle Swarm Optimization

Author Contributions

Edmond Randriamora: Conceptualization, Methodology, Supervision, Writing – review & editing

Olivier Mickael Ranarison: Software, Writing – original draft

Rivo Mahandrisoa Randriamaroson: Supervision

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Wael AMOR, Power flow for large and poorly conditioned networks, E. T. S., University of Quebec, 2018, pp. 43. https://espace.etsmtl.ca/id/eprint/2245/1/AMOR_Wael.pdf

[2] Ebeed M., Kame S., Jurado F., Optimal Power Flow Using Recent Optimization Techniques, Classical and Recent Aspects of Power System Optimization, Elsevier Inc., 2018, pp. 157-183. https://doi.org/10.1016/B978-0-12-812441-3.00007-0

[3] Emmanuel A. Nyiekaa, Ikenna Onyegbadue, Optimal Power Flow Methods – A survey, International Journal of Advances in Engineering and Manangement, vol. 3, issue 2, feb 2021, pp. 20-32. https://ijaem.net/issue_dcp/Optimal%20Power%20Flow%20Methods%20%20A%20Survey.pdf

[4] M. Abido, Optimal Power Flow Using Particle Swarm Optimization, International Journal of Electrical Power Energy System, 24, 2002, pp. 563-571. https://doi.org/10.1016/S0142-0615(01)00067-9

[5] Arul Ponnusamy, Rengarajan N., Optimal Power Flow Solution using Cuckoo Search Algorithm, ARPN Journal of Engineering and Applied Sciences, vol. 9, no. 12, December 2014, pp. 2687-2691. https://www.arpnjournals.com/jeas/research_papers/rp_2014/jeas_1214_1382.pdf

[6] Bouchekara H., Abido M., Boucherma M., Optimal power flow using teaching-learning based optimization technique, Electr. Power Syst. Res., 114, 2014, pp. 49-59. https://doi.org/10.1016/j.epsr.2014.03.032

[7] O. M. Ranarison, E. Randriamora, H. Andriatsihoarana, Optimal Power Flow Using Modified Driving-Training Based Optimization algorithm, International Journal of Advances in Engineering and Management, vol. 7, Issue 02 Feb. 2025, pp. 846-860. https://doi.org/10.35629/5252-0702846860

[8] Juan Li, Qing An, Hong Lei, Qian Deng, Gai-Ge Wang, Survey of Levy Flight-Based Metaheuristic for Optimization, Mathematics 10, no. 15: 2785, 2022. https://doi.org/10.3390/math10152785

[9] Edmond Randriamora, Faly Ramahatana, Yvon Andrianaharison, Optimal Power Flow by Cuckoo Search Via Levy Flight Algorithm, International Journal of Engineering Research & Technology, vol. 6, Issue 12, December 2017, pp. 32-37. https://www.ijert.org/research/IJERTV6IS120041.pdf

[10] Ghasemi M., Ghavidel S., Gitizadeh M., Akbari E., An improved teaching learning based optimization algorithm using Levy mutation strategy for non-smooth optimal power flow, Int. J. Electr. Power Energy Syst. 65, 2015, pp. 375-384. https://doi.org/10.1016/j.ijepes.2014.10.027

[11] Dr. K. Lenin, Crowding Distance Based Particle Swarm Optimization Algorithm for Solving Optimal Reactive Power Dispatch Problem, International Journal of Research Granthaalayah, vol. 6, Issue 6, June 2018, pp. 226-237. https://doi.org/10.29121/granthaalayah.v6.i6.2018.1369

[12] Mohammad Dehghani, Eva Trojovská, Pavel Trojovský, A new human-based metaheuristic algorithm for solving optimization problems on the base of simulation of driving training process, Scientific reports, 2022. https://doi.org/10.1038/s41598-022-14225-7

[13] Daniel Kwegyir, Michael Dugbartey Terkper, Francis Boafo Effah, Emmanuel Kwaku Antoh, Stacy Gyamfuah Lumor, Improved Driving Training-Based Optimization Algorithm Using Levy Flight and Crowding Distance Techniques, Research Reports on Computer Science, vol. 3, Issue 1, April 2024, pp. 12-28. https://doi.org/10.37256/rrcs.3120244384

[14] Deb K., Pratap A., Agarwal S., Meyarivan T., A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput., 2002, 6, pp. 182-197. https://doi.org/10.1109/4235.996017

[15] Bechikh S., Kessentini M., Said L. B., Ghédira K., Preference incorporation in evolutionary multiobjective optimization: A survey of the state-of-the-art, Adv. Comput., 2015, 98, pp. 141-207. https://doi.org/10.1016/bs.adcom.2015.03.001

[16] Gao Y., Zhang H., Duan Y., A novel hybrid PSO based on levy flight and wavelet mutation for global optimization, PLOS ONE, 18, e0279572, 2023. https://doi.org/10.1371/journal.pone.0279572

[17] Kamaruzaman A. F., Zain A. M., Yusuf S. M., Udin A., Levy Flight Algorithm for Optimization Problems – A Literature Review, Appl. Mech. Mater, 421, 2013, pp. 496-501. https://doi.org/10.4028/www.scientific.net/AMM.421.496?urlappend=%3Futm_source%3Dresearchgate.net%26utm_medium%3Darticle

[18] Chawla M., Duhan M., Levy Flights in Metaheuristics Optimization Algorithms – A Review, Appl. Artif. Intell., 32, 2018, pp. 802-821. https://doi.org/10.1080/08839514.2018.1508807

[19] Abou El Ela A. A, Abido M. A, Optimal power flow using differential evolution algorithm, Electr. Power Syst. Res., 80(7), 2010, pp. 878-885. https://doi.org/10.1016/j.epsr.2009.12.018

[20] Alsac O., Stot B., Optimal load flow with steady-state security, IEEE Trans. Power App. Syst., PAS-93 (3), 1974, pp. 745-751. https://doi.org/10.1109/TPAS.1974.293972