TEPES
TURKISH JOURNAL OF
ELECTRICAL POWER
AND ENERGY SYSTEMS

TEPES, Vol. 2, Issue. 2, 115-123, 2022 **DOI:** 10.5152/tepes.2022.22015

RESEARCH ARTICLE

Enhancement of Transmission System Security with Archimedes Optimization Algorithm

Erdi Doğan[1], Nuran Yörükeren[2]

[1]North-West Anatolian Dispacthing Center of Turkish Electricity Transmission Co.
[2]Department of Electrical Engineering, Kocaeli University, Kocaeli, Turkey

Cite this article as: E. Doğan, N. Yörükeren. Enhancement of transmission system security with archimedes optimization algorithm, *Turk J Electr Power Energy Syst*, 2022; 2(2), 115-123.

ABSTRACT

The reliability of the transmission system is crucial in terms of both economic and social welfare. On one hand, the transmission lines getting overloaded because of a single outage situation may give rise to even a blackout. On the other hand, the excessive short-circuit current might cause distortion of valuable system devices. Therefore, the static security of the transmission system should be provided so as to protect consumers and system components. The bus splitting optimization (BSO) is a tool that maintains system security by distributing transmission feeders to the suitable busbars. The BSO is a non-convex, combinatorial, and binary problem due to AC power flow equations, short-circuit calculations, and the nature of the representation of the feeder position. Hence, the novel Archimedes optimization algorithm (AOA) is utilized in order to solve the BSO problem instead of the deterministic approaches in this study. Analyses we implemented show that the AOA algorithm has the efficacy and robustness to solve the BSO problem in the Istanbul Anatolian Region of the Turkish Electricity Transmission System and to enhance the power system security.

Index Terms—Meta-heuristics, N-1 security, optimization in transmission network, restriction of short-circuit current.

I. INTRODUCTION

Bus splitting optimization (BSO), also known as network topology reconfiguration, bus layout optimization, or transmission line switching, has been recognized as an effective improvement in power system operations for maximizing network infrastructure resources and lowering operational costs [1]. Furthermore, the BSO is implemented in order to both restrict short-circuit current (SCC) [2] level and to maintain static security of the power system in contingency [3,4]. Therefore, the BSO method is a significant and cost-effective way to enhance the system condition in terms of economic and technical aspects.

A higher SCC value results from enhanced network connections as well as an increase in the number of generators [5]. Excessive SCC has already been a concern in several regions such as in the China Power System [5] and the Turkish Electricity Transmission System (TETS) [6]. If the over SCC that the circuit breaker cannot interrupt is not prevented, the system equipment can get exposed to significant damages and large interruptions that reduce social welfare can take place. The impedance of the transmission system can be varied by regulating the network topology in a way that the operation costs are alleviated and the N-1 security is strengthened [7]. Furthermore, although designing an efficient transmission reconfiguration is a convenient approach in order to decline the SCC, the power system reliability might weaken in the N-1 situation due to the alteration of the topological framework. However, the power system reliability decreases in contingency due to alteration of the topological framework. Hence, the appropriate bus layout should be created not only to restrict the SCC but also to provide power system static security in the N-1 contingency.

The BSO problem that focuses to reduce operational costs is described as a mixed-integer non-linear programming problem, with each feeder representing one binary variable. Bus splitting optimization is an Non-Polynomial-hard (NP)-hard problem, and its low computational efficiency prevents it from being used in real-world power system operations [5]. Many techniques such as the Benders decomposition method [8], second-order cone programming incorporating

Corresponding author: Erdi Doğan, erdi.dogan.teias@gmail.com

Received: February 23, 2022
Accepted: March 23, 2022
Publication Date: May 18, 2022

heuristic [9], and DC equations in optimal power flow [7,8,10] are used to deal with the complexity of this problem. On other hand, the BSO problem regarding an improvement of the system security relieving the loading levels and SCC instead of reducing operational costs is considered as an integer non-linear programming problem [3,4,6]. Another option with regard to the BSO problem is to intentionally island in cascading failures so as to hinder large-scale blackouts in the power system [11-14].

Over the last two decades, meta-heuristic optimization approaches have grown in popularity [15]. The reason for the common use of the meta-heuristic techniques in a variety of engineering areas is their simple, derivative-free, and flexible implementation properties. However, the no free lunch theorem showed that on one set of problems, a meta-heuristic may produce excellent results, while on another set of problems, the same approach may produce terrible results [16]. In order to tackle diverse problem sets, various spectacular meta-heuristic strategies such as genetic algorithm [17], particle swarm optimization [18], gravitational search algorithm [19], and ant colony optimization [20] were proposed. Therefore, it is necessary to determine which algorithm can effectively solve the BSO problem in the real-world large-scale power system. This is the main motivation of this research, which utilizes a novel meta-heuristic optimization algorithm for solving the BSO problem.

There are two main voltage levels, one is 154 kV and the other is 400 kV in TETS and bulk power generation in market is connected to a 400 kV voltage level but the over SCC occurs at a 154 kV voltage level. As a result, the focus of this work is on the BSO problem, which was adopted to increase power system security because the optimal power flow at 154 kV voltage level is not required in TETS. The Archimedes optimization algorithm (AOA) [21], a novel physic-based meta-heuristic, is used to tackle the BSO problem in TETS. The Python programming language is used in order to create related software and all information about power flow equations and SCC in TETS is retrieved from the PSS/E version 35.2 program. Mealpy software [22], a collection of the state-of-the-art meta-heuristics algorithms in Python, is utilized to solve the BSO problem with AOA. In order to deal with the binary structure of the BSO problem, some alterations have been implemented to the optimizer module of the Mealpy software.

Main Points

- Formulation of bus splitting optimization problem incorporating short-circuit current and N-1 security in Turkish Electricity Transmission System.
- Introducing how to enhance the static security of the power system without extra cost by solving a problem with non-convex, non-linear, multi-objective, and binary nature through a meta-heuristic optimization algorithm.
- Implementing Archimedes optimization algorithm, a novel math-based algorithm, in order to deal with a real-world large-scale challenging problem.
- Suggesting the two-byte transformation technique from the decimal search space into the binary one so as to achieve a better solution in binary problem from meta-heuristics designed for the continuous problem.

The rest of the paper is organized as follows: section II reveals the structure and mathematical formulation of the BSO problem, the basic principle of the AOA, and how to integrate the AOA into the BSO problem. Section III presents numerical results and discussion on the TETS to demonstrate the efficiency of the suggested strategy. Ultimately, in section IV, conclusions are drawn.

II. METHODS

A. Bus Splitting Optimization Problem Formulation

Bus splitting optimization is related to the network topology reconfiguration (NTR); however, they are not equal concepts since BSO just creates bus islanding by changing the feeder positions of the substations, whereas NTR implements both bus islanding and transmission switching actions. As transmission switching action in the large-scale real-world power system might lead to security violations in a single outage situation, it is not considered in the optimization problem of this study. The objective function of the BSO problem is to minimize overloading of transmission lines and voltage violation of all busbars in both base case and N-1 situations which can be formulated as follows:

$$min.\sum_{l:1}^{N}\frac{S_l}{S_l^{limit}}x\frac{100}{n_l}+\sum_{c:1}^{N_c}\sum_{l:1}^{N}\frac{S_{c,l}}{S_l^{Max}}x\frac{100}{n_l}+\sum_{c:1}^{N_c}\sum_{i:1}^{B}\left({V_{vio}}^{c}\right)^2 * n_R, l\neq c \quad (1)$$

$$if\ {V_{vio}}^{c} > {V_{vio}}^{limit} \rightarrow {V_{vio}}^{c} = \frac{V_i^c - 1}{V_i^{max} - V_i^{min}} \quad (2)$$

$$if\ S_l < S_l^{limit} \rightarrow n_l = n\ and\ if\ S_l \geq S_l^{limit} \rightarrow n_l = 1 \quad (3)$$

$$if\ S_{c,l} < S_l^{Max} \rightarrow n_l = n\ and\ if\ S_{c,l} \geq S_l^{Max} \rightarrow n_l = 1 \quad (4)$$

where n_l symbolizes a constant coefficient for line l, S_l and $S_{c,l}$ represent apparent power flow of line l in base case and contingency case, and S_l^{limit} and S_l^{Max} stand for the thermal limit of line l and maximum allowable power flow before overcurrent protection trips at line l, respectively. B, ${V_{vio}}^{c}$, n_R, V_i^c, V_i^{max}, V_i^{min}, and ${V_{vio}}^{limit}$ represent the number of busbar, the voltage violation for busbar i in contingency c, the constant weight for bus voltage violation, voltage at the bus i in contingency c, the maximum and minimum voltage for bus i, and the voltage violation limit of busbar, respectively:

$$st.\sum_{i:1}^{B}\left(\frac{sc_i}{c_{SC}}\right)^n, if\ sc_i > sc_{limit} \quad (5)$$

$$if\ N_b \geq 4 \rightarrow z_b \epsilon K, \forall b \quad (6)$$

$$r = 2 \leq \sum_{i:1}^{N_b} bc_i \leq N_b - 2, \forall b \quad (7)$$

$$if\ r = True \rightarrow z_b = 0\ else\ z_b = 1, \forall b \quad (8)$$

where sc_i, sc^{limit}, c_{sc}, z_{sub}, K, N_b, b_{ci}, and r represent SCC of bus i, SCC limit at related voltage level, a constant coefficient, zero impedance line in a substation, substation list having zero impedance line, the number of lines in substation b, binary code of line i at substation b, and the necessary condition to disconnect a zero impedance line, respectively.

The SCC is considered as a constraint in the BSO problem in this study. However, various numerical analyses implemented show that if the SCC level of busbars in the power system studied is around the SCC limit, the algorithm does not manage to overcome the SCC constraint, resulting in divergence. Therefore, any violation in the SCC constraint incorporates into the objective function as a punishment. This technique is known as aggregating method and it is used as follows:

$$min.f + \mu_l * f_{sc} \tag{9}$$

$$f_{sc} : \sum_{i:1}^{B} \frac{sc_i}{2m + m * sgn\left(sc^{limit} - sc_i\right)} \tag{10}$$

where f, m, μ_l, and sgn represent original objective function shown in eq-1, constant coefficients, and signum function, respectively.

The topological framework created in the transmission system should cover the needs of the power flow equations. If the binary codification designed by the meta-heuristic algorithm brings about a busbar that its only feeder is a transformer, the base case power flow equations do not converge. Therefore, any divergence of power flow equations in created bus layout should be included in the problem as either a constraint or an objective and it can be attached to the objective function by using the aggregate method as follows:

$$min.f + \mu_l * f_{sc} + M \tag{11}$$

where M is a big number to be used in order to punish related topology due to divergence in power flow equations.

B. Archimedes Optimization Algorithm

The AOA, a physic-based meta-heuristic algorithm, is based on the principle that when an object is immersed in a fluid, the fluid enforces an upward buoyant force on the object equivalent to the weight of the fluid displaced by the object [21]. We focus on the mathematical formulation of the AOA but further information about the inspiration of the algorithm can be viewed from the original paper.

$$O_i = lb_i + rand * \left(ub_i - lb_i\right), \forall i \tag{12}$$

$$den_i, vol_i = rand, \forall i \tag{13}$$

$$acc_i = lb_i + rand * \left(ub_i - lb_i\right), \forall i \tag{14}$$

Where O_i, lb, ub, den, vol, acc, and $rand$ represent object i of the population, lower and upper bounds of the search space, density, volume, acceleration, and randomly generated number between 0 and 1. These equations are used to create initial positions (x_{best}) of entire population and x_{best}, den_{best}, vol_{best}, and acc_{best} are determined by evaluating the performance of each position.

$$den_i^{t+1} = den_i^t + rand * \left(den_{best} - den_i^t\right), \forall i \tag{15}$$

$$vol_i^{t+1} = vol_i^t + rand * \left(vol_{best} - vol_i^t\right), \forall i \tag{16}$$

$$TF = exp\left(\frac{t - t_{max}}{t_{max}}\right) \tag{17}$$

$$d^{t+1} = exp\left(\frac{t_{max} - t}{t_{max}}\right) - \left(\frac{t}{t_{max}}\right) \tag{18}$$

$$if\ TF \le 0.5 \rightarrow acc_i^{t+1} = \frac{den_k + vol_k * acc_k}{den_i^{t+1} * vol_i^{t+1}} \tag{19}$$

$$if\ TF > 0.5 \rightarrow acc_i^{t+1} = \frac{den_{best} + vol_{best} * acc_{best}}{den_i^{t+1} * vol_i^{t+1}} \tag{20}$$

$$acc_{i-norm}^{t+1} = 0.9 * \frac{acc_i^{t+1} - \min\left(acc\right)}{\max\left(acc\right) - \min\left(acc\right)} + 0.1 \tag{21}$$

$$if\ TF \le 0.5 \rightarrow x_i^{t+1} = x_i^t + C_1 * rand * acc_{i-norm}^{t+1} * d * \left(x_{rand} - x_i^t\right) \tag{22}$$

$$if\ TF > 0.5 \rightarrow x_i^{t+1} = x_{best}^t + F * C_2 * rand * acc_{i-norm}^{t+1} * d * \left(C_3 * TF * x_{best} - x_i^t\right) \tag{23}$$

$$F = \begin{cases} +1\ if\ 2 * rand - C_4 \le 0.5 \\ -1\ if\ 2 * rand - C_4 > 0.5 \end{cases} \tag{24}$$

where den_{best}, vol_{best}, and $rand$ are density and volume of the best object found so far and uniformly distributed random number, respectively, TF, d, t, and t_{max} represent transfer operator, density factor, current, and maximum number of iteration, den_k, vol_k, and acc_k symbolize density, volume, and acceleration of kth object randomly chosen from the population, respectively, acc_{i-norm}^{t+1} is the normalization of acc_i^{t+1}, x_i^{t+1} is the next position of the object i, C_1, C_2, C_3, and C_4 are constant coefficients, F is utilized like a flag changing the direction of motion. Further explanation about these variables can be examined from [22].

The balance between exploration and exploitation phases is ensured through transfer operator TF and density factor d. Transfer operator imitates the idea that if two objects are immersed in the fluid, a collision between objects occurs but the objects reach an equilibrium state after a period of time. In this direction, the exploration phase is operated by using the concept that if the transfer operator is lower than 0.5, the collision between two agents of the population happens through equations 19 and 22. However, if the objects arrive at an equilibrium state, the exploitation phase is served via using equations 20 and 23.

C. Integration of the Archimedes Optimization Algorithm into the Bus Splitting Optimization Problem

The AOA algorithm is designed to solve problems having continuous search space although the structure of the BSO problem is binary. Hence, we should deal with the transition process from continuous to binary search space. There have been used various transfer functions (TF) and position updating rules (PUR) in order to handle this problem [23,24]. However, additional stochastic operations are attached to the original meta-heuristic algorithm in case TF and PUR are utilized. Therefore, we focus on decimal to binary transition without using an additional operator in this study.

1) Position Structure of the BSO Problem

The binary codes representing the busbar location of the feeder are used to accomplish proper distribution. Fig. 1 shows that if line-1 has zero binary code, this means that it will be connected to the bus-1. Therefore, the position of all feeders in each substation can be symbolized with the binary codification.

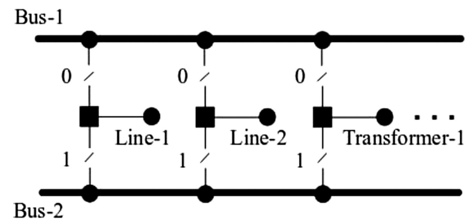

Fig. 1. Binary representation of the position of the feeders.

2) Transition from Decimal to Binary (Two-Byte Transformation)

The direct transformation can be implemented, nonetheless as mentioned in [25], the length of the binary codification should be rigorously chosen. It is worth stating the phenomenon that if decimal codification is changed from 3 to 4, the binary codification will alter from "011" to "100." This is a great example to see that small changes in decimal codification can result in a significant alteration in binary codification. Therefore, the maximum length of the binary code related to decimal has been chosen as two in this study and the structure of the transformation can be seen in Fig. 2.

As seen in Fig. 2, one decimal variable is related to six binary codes in which two of them represent the integer part while the rest symbolize the fractional part. Since the length of the binary code corresponding to one integer number is chosen as two, the upper bound of the integer should be three ("11"). Nevertheless, because the fractional part of the decimal code can be greater than three, the modulo operation is implemented for the fractional part.

3) Algorithm

A meta-heuristic technique is only interested in input and output, and the internal structure of the problem is considered as a black

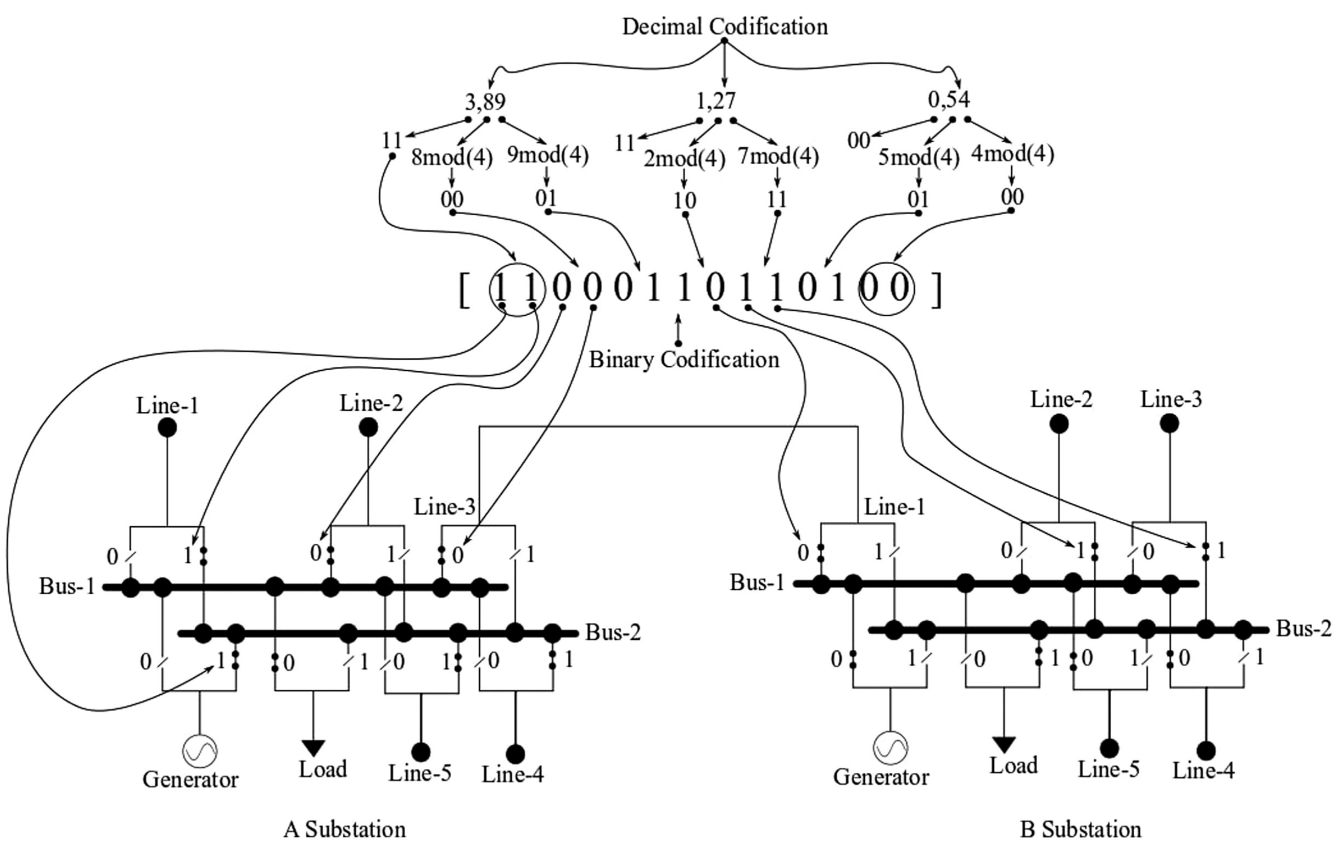

Fig. 2. Transition from decimal to binary codification regarding the representation of the bus layout.

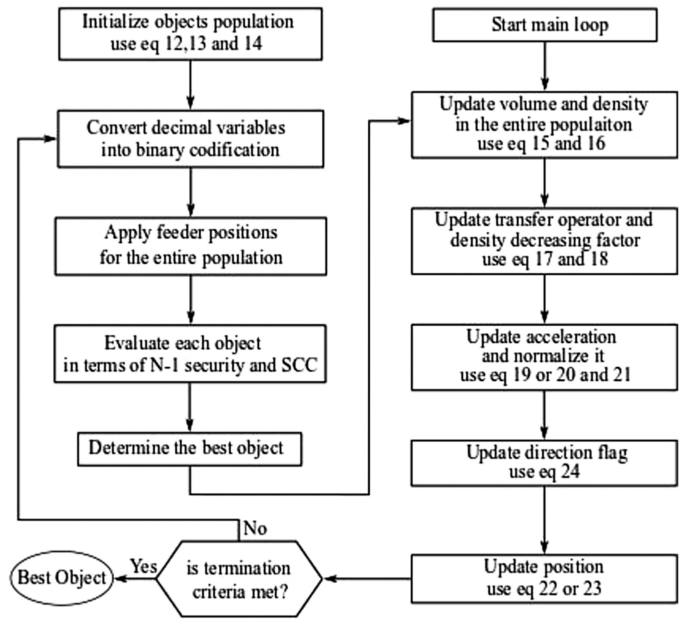

Fig. 3. Algorithm used to solve the BSO problem with the AOA. BSO, bus splitting optimization; AOA, Archimedes optimization algorithm.

TABLE I

SHORT-CIRCUIT CURRENTS OF BUSES IN ISTANBUL ANATOLIAN REGION FOR THE BASE CASE (ALL ZERO IMPEDANCE LINES ARE IN SERVICE) AND AFTER BUS SPLITTING OPTIMIZATION

		Base Case		After Bus Splitting Optimization	
No	**Bus Name**	**Three-Phase Fault**	**Line-Ground Fault**	**Three-Phase Fault**	**Line-Ground Fault**
1	210121	**35983.9**	**32259.8**	30279.8	27820.5
2	210122	**33029.3**	28882.9	28594.0	25684.1
3	211221	**42774.5**	**39715.6**	28900.8	25200.1
4	211321	29045.9	23935.9	24548.5	20085.2
5	211421	27196.9	21426.3	23258.4	18391.0
6	211521	**35417.3**	**37813.5**	23743.5	24302.1
7	211522	**35417.3**	**37813.5**	26318.6	27992.4
8	211621	**35723.6**	**33319.2**	21269.1	19722.5
9	211622	**35723.6**	**33319.2**	20676.6	19301.2
10	211821	**35112.4**	28549.7	26483.8	21588.8
11	211822	**35112.4**	28549.7	16371.6	11225.5
12	214321	**33965.4**	**33267.0**	24714.3	24918.4
13	214322	**33965.4**	**33267.0**	14096.0	10422.8
14	214821	**33650.0**	30762.0	21543.4	19767.4
15	214822	**33650.0**	30762.0	7026.6	4120.0
16	216721	**30488.5**	25602.0	6849.7	3986.0
17	216722	30488.5	25602.0	15797.4	12236.6
18	217221	24871.6	22444.0	21238.0	19828.1

box. In this paper, AOA provides binary codes representing the position of the feeders as an input by using the transformation from decimal to binary and alterations of positions in substations according to this binary codification are implemented. The overall algorithm related to the search process is presented in Fig. 3.

First, an initial population is created with randomly generated objects. Then, the main loop is started after evaluating initial positions and determining the best fitness. The normalized accelerations for the entire population are calculated by utilizing obtained volumes, densities, transfer operator, and density decreasing factor. In the final step used for the main loop of the AOA, positions are updated in continuous search space after determining the position direction flag and these obtained decimal positions are converted into the binary codification. These new positions are evaluated in terms of SCC and N-1 security and the performance of the solution is transferred to the AOA as an output. A number of modifications in binary codification are performed by the AOA at each step and this process goes on over iterations in order to obtain a better solution.

III. RESULTS AND DISCUSSION

Archimedes optimization algorithm that mimics the idea that the weight of the displaced fluid is proportionate to the buoyant force produced upward on an object immersed in a fluid is used to solve the BSO problem of the Istanbul Anatolian Side of the TETS. For the sake of protecting information security, any impedance value or substation name related to the TETS is not presented. The algorithm is coded using Python programming language and PSS/E version 35.2 program on an Intel® Core™ i7-8850U CPU at 2.60GHz with 16GB of RAM.

The Istanbul Anatolian Region contains 28 substations and 51 lines with a voltage of 154 kV. In 14 of these substations, there is a coupling circuit breaker between busbars. On one hand, many autotransformer connections between 400 kV and 154 kV voltage levels have been built in this region resulting in the generation of high-level SCCs. It is worth mentioning that the SCC limit of the circuit breakers in this region remains at 31.5 kA for the 154 kV voltage level. On the other hand, the intensive consumption of this region requires that special attention should be devoted to N-1 security [6].

C_1, C_2, C_3, and C_4 coefficients used in the equations 22-24 are set as 2, 6, 2, and 0.5, respectively. The population size and the number of iterations of the AOA have been applied as 100 and 500. Obtaining a quality solution to the BSO problem in the real-world power system took 119 hours. If all zero impedance lines are in service in the related region, the N-1 security score of the power system will be 3.13, while the SCC score will reach 1.142415×10^6. Table I also shows that almost every busbar is exposed to the over SCC before the implementation of the suitable bus splitting. After the best solution obtained from the AOA is executed to the power system, both the system static security in the single outage is preserved by acquiring the 3.11 score and the SCC score is reduced to 131.9, which do not lead to any complication in the related region of the real-world power system.

The convergence curve showing the solution quality of the algorithm over iterations is demonstrated in Fig. 4. As can be seen in this curve, the best fitness is obtained around 200th iterations (exactly 176th) which means that a quality solution that can be implemented in the power system is achieved in around 60 hours. The maximum function evaluation number was implemented to be able to avoid local optimal points. The amount of the function evaluation can

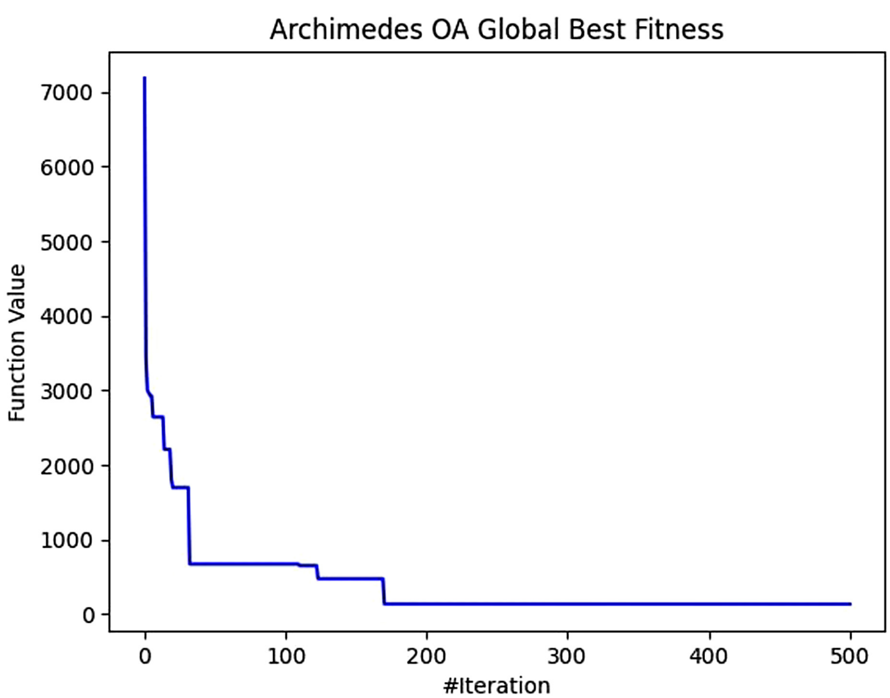

Fig. 4. Convergence curve of the AOA in the BSO problem. BSO, bus splitting optimization; AOA, Archimedes optimization algorithm.

Fig. 5. Trajectory of some individuals of the population at first and second dimension.

be reduced in order to decrease the solution time of the problem. However, it is worth mentioning that the lower function evaluation might lead to the algorithm not converging to the global minimum.

There are 11 substations to be split and 96 feeders to be distributed in the Istanbul Anatolian Region of the TETS and so there are 96 decision variables in the binary search space of the problem. However, the number of continuous variables to be used in AOA is 16 because of implementing the transition from the decimal to the binary. The trajectory for the first element of the 6th, 18th, and 23rd individuals, each of whom has 16 dimensions, is presented in Fig. 5 and the qualified bus splitting strategy, produced for the

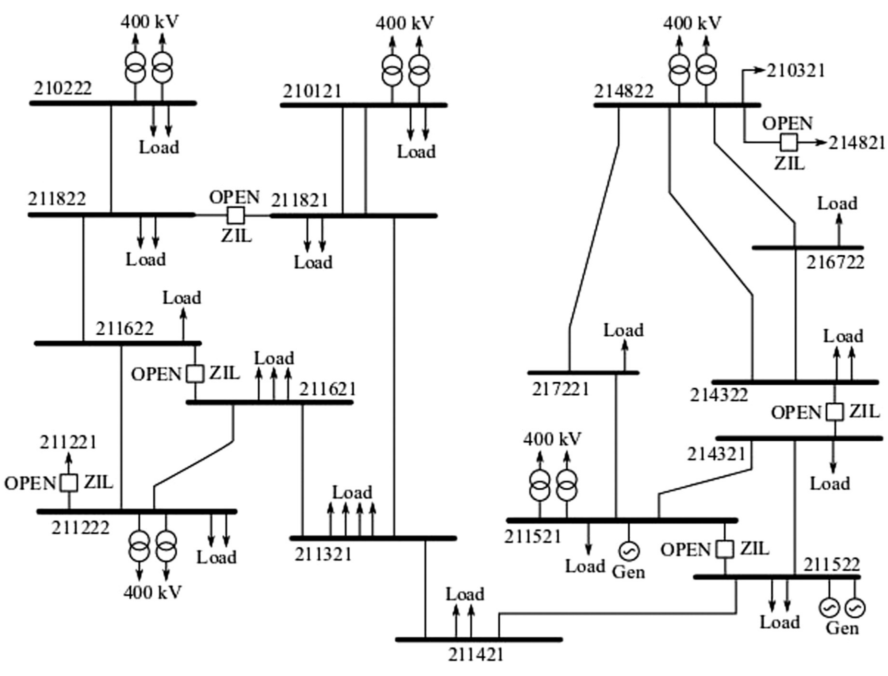

Fig. 6. Feeder distributions at substations in the Istanbul Anatolian Region of TETS obtained from AOA. AOA, Archimedes optimization algorithm; TETS, Turkish Electricity Transmission System.

Istanbul Anatolian Side of TETS acquired from AOA, is shown in Fig. 6. The trajectory of the first agents shows that individuals investigate different regions of the search space.

The zero impedance lines between “211821-211822,” “211621-211622,” “211221-211222,” “211521-211522,” “214321-214322,” and “214821-214822” are disconnected in order to reduce the over SCC as depicted in Fig. 6. This topological framework not only restricts the SCC but maintains the system security in the N-1 situation. Therefore, the results demonstrate the efficacy of the AOA technique in presenting valuable solutions that can help the system operators in their decision process.

IV. CONCLUSION

The BSO problem is one of the most substantial problems that transmission system operators might encounter with everyday operations in the power system due to a large number of possibilities. Although the bus splitting technique is beneficial in terms of curtailing the SCCs, the static security of the transmission system should be sustained after splitting. To deal with this crucial problem in the real-world large-scale power system, the novel AOA is used in this study. The approach for the transformation from decimal to binary is introduced in order to cope with the BSO problem. The numerical study on the TETS substantiates the effectiveness of the AOA with the proposed decimal-binary transition architecture in order to enhance the power system security in terms of SCC reduction and avoidance of overloading.

As mentioned in the numerical results section, 50 000 function evaluation was implemented to be able to deal with the BSO problem, and therefore, the solution time of the optimization process reached long hours. Network reduction might be applied or DC power flow equations instead of AC can be utilized to decrease the solution time of the problem. Another option is machine learning techniques that can be performed so as to handle this problem in future works. However, it is worth keeping in mind that any negligence in power flows or SCC may bring about an inappropriate topology since the related region studied in TETS is operated already in its extremity.

Peer-review: Externally peer-reviewed.

Declaration of Interests: The author has no conflicts of interest to declare.

Funding: The authors declared that this study has received no financial support.

REFERENCES

1. P. Dehghanian, and M. Kezunovic, “Probabilistic impact of transmission line switching on power system operating states,” IEEE Publications/PES Transmission and Distribution Conference and Exposition (T&D), Dallas, TX, USA, 3–5 May 2016, pp. 1–5.
2. S. Namchoat, and N. Hoonchareon, “Optimal bus splitting for short-circuit current limitation in metropolitan area,” 10th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, ECTI-CON 2013, Krabi, Thailand, 15–17 May 2013, pp. 1–5.
3. E. Doğan, and N. Yörükeren, “Binary pathfinder algorithm for bus splitting optimisation problem,” *IET Gener. Transm. Distrib.*, vol. 14, no. 26, pp. 6613–6623, 2020. [CrossRef]
4. E. Doğan, and N. Yörükeren, “Optimal bus layout in transmission system by using meta-heuristic approaches,” *Electr. Power Compon. Syst.*, vol. 48, no. 12–13, pp. 1390–1400, 2020. [CrossRef]
5. Z. Yang, H. Zhong, Q. Xia, and C. Kang, “Optimal transmission switching with short-circuit current limitation constraints,” *IEEE Trans. Power Syst.*, vol. 31, no. 2, pp. 1278–1288, 2016. [CrossRef]
6. E. Doğan, and N. Yörükeren, “Optimal bus layout in electricity transmission system by using binary particle swarm optimization: Turkey case study,” *SEERC Colloq.*, vol. 2021, 2021, Austria-online.
7. M. Heidarifar, and H. Ghasemi, “A network topology optimization model based on substation and node-breaker modeling,” *IEEE Trans. Power Syst.*, vol. 31, no. 1, pp. 247–255, 2016. [CrossRef]
8. M. Heidarifar, M. Doostizadeh, and H. Ghasemi, “Optimal transmission reconfiguration through line switching and bus splitting,” IEEE PES Gen. Meeting | Conference & Exposition, National Harbor, MD, USA, 27–31 July 2014, 2014, pp. 1–5.
9. M. Heidarifar, P. Andrianesis, P. Ruiz, M. C. Caramanis, and I. C. Paschalidis, “An optimal transmission line switching and bus splitting heuristic incorporating AC and N-1 contingency constraints” *Int. J. Electr. Power Energy Syst.*, vol. 133, p. 107278, 2021. [CrossRef]
10. Y. Zhou, A. S. Zamzam, A. Bernstein, and H. Zhu, “Substation-level grid topology optimization using bus splitting,” Am. Control. Conference (ACC), New Orleans, LA, USA, 25–28 May 2021, pp. 2868–2874.
11. T. Ding, K. Sun, Q. Yang, A. W. Khan, and Z. Bie, “Mixed integer second order cone relaxation with dynamic simulation for proper power system islanding operations,” *IEEE J. Emerg. Sel. Top. Circuits Syst.*, vol. 7, no. 2, pp. 295–306, 2017. [CrossRef]
12. N. Z. Saharuddin, I. Z. Abidin, H. Mokhlis, A. R. Abdullah, and K. Naidu, “A power system network splitting strategy based on contingency analysis,” *Energies*, vol. 11, no. 2, 2018. [CrossRef]
13. P. Fernández-Porras, M. Panteli, and J. Quirós-Tortós, “Intentional controlled islanding: When to island for power system blackout prevention,” *IET Gener. Transm. Distrib.*, vol. 12, no. 14, pp. 3542–3549, 2018. [CrossRef]
14. P. Demetriou, A. Kyriacou, E. Kyriakides, and C. Panayiotou, “System splitting strategy considering power system restoration,” *IEEE Manchester PowerTech*, vol. 2017, pp. 1–6, 2017.
15. S. Mirjalili, S. M. Mirjalili, and A. Lewis, “Grey wolf optimizer,” *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014. [CrossRef]
16. D. H. Wolpert, and W. G. Macready, “No free lunch theorems for optimization,” *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, 1997. [CrossRef]
17. J. H. Holland, “Genetic algorithms,” *Sci. Am.*, *Scientific American*, vol. 267, no. 1, pp. 66–72, 1992. [CrossRef]
18. R. Eberhart, and J. Kennedy, “A new optimizer using particle swarm theory,” Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 4–6 October 1999, pp. 39–43.
19. E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, “GSA: A Gravitational Search Algorithm” *Inf. Sci.*, vol. 179, no. 13, pp. 2232–2248, 2009. [CrossRef]
20. M. Dorigo, M. Birattari, and T. Stutzle, “Ant colony Optimatization,” *IEEE Comp. Intell. Mag.*, vol. 1, no. 4, pp. 28–39, 2006.
21. F. A. Hashim, K. Hussain, E. H. Houssein, M. S. Mabrouk, and W. Al-Atabany, “Archimedes optimization algorithm: A new metaheuristic

algorithm for solving optimization problems," *Appl. Intell.*, vol. 51, no. 3, pp. 1531–1551, 2021. [CrossRef]

22. N. van Thieu, and D. Molina, "Meta-heuristic algorithms using Python (MEALPY-software)," *Zenodo*, 2021. [CrossRef]
23. E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "BGSA: binary gravitational search algorithm," *Nat. Comput.*, vol. 9, no. 3, pp. 727–745, 2010. [CrossRef]
24. S. Mirjalili, and A. Lewis, "S-shaped versus V-shaped transfer functions for binary Particle Swarm Optimization," *Swarm Evol. Comput.*, vol. 9, pp. 1–14, 2013. [CrossRef]
25. R. A. Romero, A. Monticelli, and R. Romero, "Transmision system expansion planning by an extended genetic algorithm," *IEE Proc. Gener. Transm. Distrib.*, vol. 145, no. 3, pp. 329 – 335, 1998. [CrossRef]