Application of Hybrid HS and Tabu Search Algorithm for Optimal Location of FACTS Devices to Reduce Power Losses in Power Systems

Zynab Masomi Zohrabad
Department of Electrical Engineering, Faculty of Engineering
Mashhad Branch, Islamic Azad University
Mashhad, Iran
Zynab.MasomiZohrabad@chmail.ir

***Abstract*—Power networks continue to grow following the annual growth of energy demand. As constructing new energy generation facilities bears a high cost, minimizing power grid losses becomes essential to permit low cost energy transmission in larger distances and additional areas. This study aims to model an optimization problem for an IEEE 30-bus power grid using a Tabu search algorithm based on an improved hybrid Harmony Search (HS) method to reduce overall grid losses. The proposed algorithm is applied to find the best location for the installation of a Unified Power Flow Controller (UPFC). The results obtained from installation of the UPFC in the grid are presented by displaying outputs.**

Keywords-Tabu search optimization algorithm; harmony search optimization algorithm; FACTS devices; UPFC; IEEE 30-bus grid; reduced losses

I. Introduction

Grid stability is an important topic in power systems as any delay or inability in stability control will affect quality and servicing loads. To provide a reliable and practical function, it is required to use precise control systems with high efficiency. Flexible alternating current transmission systems (FACTS) allow a significant improvement in energy transmission with low investment cost and quick control [1]. The focus of this study is to find the optimal location of a Unified Power Flow Controller (UPFC) in an IEEE 30-bus grid for reducing losses in the power system. The method used is a hybrid Harmony search (HS) and Tabu search algorithm, the results of which are compared to that of standard HS.

Basically, UPFC has two voltage source inverters (VSI) and a common DC link capacitor which is connected to the grid by two series-parallel transformers. Potentially, UPFC performs all series and parallel compensations simultaneously and can control phase angle, impedance and voltage range continuously. Thus, it can control the real and reactive power of transmission line independently. UPFC series and parallel elements work independently [2]. Harmony search (HS) algorithm was first presented in [3]. HS algorithm has been inspired by music. The effort to find harmony in music resembles the effort to find the optimal conditions in an optimization process. In other words, the creativity of a musician to create an art piece could be compared with the search process in an optimization problem. In fact, there is a relative similarity between music and optimization as musical instruments correspond with decision variables in the optimization problem. In general, this algorithm is able to solve different optimization problems. Moreover, the similarities between these two processes raise the idea of developing a new algorithm. In musical improvisation, the musician plays a musical scale at potential intervals, which together form a harmony vector. If all scales create an aesthetically good and acceptable harmony, this experience is stored in the memory of the musician and it will increase the likelihood of a better harmony. Similarly, a decision variable in the engineering optimization problems is initially allocated a value within a permissible range, which can form a solution vector. If all decision variables generate a good and acceptable solution based on objective function, this experience is stored in the memory of the variable and increases the likelihood of better solutions.

II. Harmony Search

In [4], an optimization process with three corresponding components was developed: 1: Harmony memory (HM), 2: Pitch adjustment, 3: Random selection. Harmony memory is very important, because it determines whether new solutions and harmonies can be accepted. To use HM effectively, a parameter is presented to indicate the likely consideration of HM harmony as HMCR (harmony memory considering rate) which is examined according to its rate and extent. For very low and close to zero HMCR, a small number of variables chosen for harmony are selected; for high and close to one HMCR, new values are randomly selected from the search space for the harmony variable. For a very high and close to one HMCR, it is more likely to select from HM and less likely to select from the search space randomly. In both cases, the algorithm does not converge to the optimal values or optimization and convergence is very slow. To balance variables and higher quality solutions in HM (intensification) and random search in the search space and escape from local optimums (diversification), the value considered for HMCR is approximately 0.7 to 0.9. The second operator is pitch adjustment involving parameters such as bandwidth (bw, maximum value of the change in the selected variable) and pitch adjusting rate (PAR). These two parameters are used to create a new harmony (solution vector). HS algorithm is described below step by step:

1. Define the objective function and algorithm parameters (HMCR, PAR and bw): to start solving the algorithm, fitness function and algorithm parameters are first determined. Then, HMCR (representing likelihood of considering a variable of the memory) and PAR (representing likelihood of change in the selected variable of the matrix memory to a random size) which are two parameters ranging from zero to one are defined. Usually, HMCR value is about 0.7 to 0.95, meaning that the likelihood of selection of HM is equal to 0.7 to 0.9. PAR value is about 0.1 to 0.5, meaning that the likelihood of minor changes in the selected value of HM is 0.1 to 0.5. To determine value of the change on the selected variable of matrix memory, another parameter (bw) is also defined, which is explained below.

2. Form a matrix memory: a HM matrix is composed of several solutions or harmonies. The overall shape of this memory matrix is as follows:

$$HM = \begin{bmatrix} x_1^1 \ldots\ldots\ldots x_N^1 \\ \vdots \\ x_1^{HMC} \ldots . x_N^{HMC} \end{bmatrix} \quad (1)$$

3. The number of rows in the memory matrix is equal to harmony memory size (HMS); each row means a solution generated randomly. In this matrix, N is the number of variables. This matrix is the same popular music that a musician wants to play. Each row is a harmony or piece of music. Most likely, none of these harmonies or solutions is potentially the best harmony or solution. Using a fitness function, fitness value or value of each solution is obtained.

4. Produce a new solution or harmony: for a new harmony, several operators are required to generate the variables separately. To create value for the i-th variable, a random number is first generated between zero and one; this random number is compared with HMCR. If the random number is smaller than HMCR, a value will be selected for i-th variable of the memory matrix and i-th column. Otherwise, a random value will be selected for i-th variable of search space. If a value of memory matrix is selected, another random number will be generated and compared with PAR. If the random number is smaller than PAR, the variable selected of the memory matrix will be changed to a smaller value by the following equation. To determine the change value on the variable selected of memory matrix, another parameter (bw) is defined. The new variable is obtained by the following variable:

$$X_{new} = X_{old} + bw + \varepsilon$$

5. Update the memory matrix: similarly, all variables of a harmony are generated; then, its value is calculated by the fitness function and compared to the worst harmony in the memory matrix. If it surpasses the worst harmony in the memory matrix, a new harmony will replace the previous harmony.

6. Stop condition: stop condition of the algorithm can be determined in different ways. Usually, stop condition is to achieve the optimal value considered. Other conditions are to keep the best harmony in the memory matrix intact or reach the specified number of repetitions [4].

III. TABU SEARCH

Tabu search which was first presented by Glover in 1986 [5] is a meta-heuristic optimization method for solving hybrid optimization problems based on local search algorithms to overcome their flaws. The overall structure of Tabu search is described below:

To achieve the optimal solution in an optimization problem, Tabu search starts to move from an initial solution. Then, the algorithm selects the best neighbor solution among neighbors of the current solution. If the solution is not on the Tabu list, the algorithm will move to the neighbor solution. Otherwise, the algorithm will check the aspiration criterion. Based on the aspiration criterion, if the neighbor solution is better than the best solution found so far, the algorithm will move towards that solution, even if it is in the Tabu list. After moving to neighbor solution, the Tabu list is updated; that is, the previous move to the neighbor solution is placed on the Tabu list to avoid return to that solution in a cycle. In fact, Tabu list is an instrument in the Tabu search algorithm by which the algorithm is prevented from falling into the local optimum. Then, a number of moves previously put in the Tabu list are removed from the list. The time when moves are placed in the Tabu list is determined by a parameter called as Tabu tenure. The move from the current solution to the neighbor solution continues until the stop criterion is met. Different stop criteria can be considered for the algorithm. For example, limited number of moves to the neighbor solution can be a stop criterion [6].

IV. SIMULATION RESULTS

In this study, the objective function is to reduce losses in an IEEE 30-bus grid, for which it is required to extract the Y Bus grid matrix. Optimization concept suggests that those values of parameters are considered that minimize or maximize the function. All proper values for this are called possible solutions and the best value of these values is called optimal solution. Optimization algorithms cover both maximization and minimization problems. The purpose of the defined problem is to reduce grid losses. By multiplying the loss function by -1, the algorithm can track the minimization of loss function [7]. In Table I, ploss is total grid losses, i is the grid line current and q is the reactive power of the IEEE 30-bus grid lines.

TABLE I. OUTPUT RESULTS OF HS ALGORITHM

ploss	i	q
2.13E+07	64.9445	-29.7154
2.13E+07	63.6607	37.7239
2.13E+07	65.4033	141.5347
2.13E+07	68.7965	-96.9429
2.13E+07	64.0102	6.6182
2.13E+07	55.4692	182.9972
2.13E+07	78.2278	136.3326
2.13E+07	46.6269	76.3367
2.13E+07	57.7902	99.7846
2.13E+07	62.7782	167.5033
2.13E+07	63.7469	-79.1372
2.13E+07	70.3663	-24.9722
2.13E+07	67.6466	142.7435
2.13E+07	56.0573	65.3306
2.13E+07	61.7173	15.1391
2.13E+07	68.983	224.2357
2.13E+07	68.0813	47.7569
2.13E+07	65.1849	-32.1558
2.13E+07	68.9633	-5.2751
2.13E+07	70.6929	143.7993
2.13E+07	69.7048	-123.3688
2.13E+07	65.7603	677.4
2.13E+07	58.19	196.0636
2.13E+07	57.1071	112.4595
2.13E+07	65.3285	187.7668
2.13E+07	59.4643	-314487
2.13E+07	61.8369	46.2256
2.13E+07	71.7156	143.9211
2.13E+07	67.7943	-73.5452
2.13E+07	66.1547	-15.941

TABLE II. OUTPUTS OF HYBRID HSAND TABU SEARCH ALGORITHM

ploss	Bus No	i	q	UPFC
2.07E+02	1	64.9445	-29.715	
2.07E+02	2	63.6607	37.7239	
2.07E+02	3	65.4033	141.535	
2.07E+02	4	68.7965	-96.943	2
2.07E+02	5	64.0102	6.6182	
2.07E+02	6	55.4692	182.997	
2.07E+02	7	78.2278	136.333	
2.07E+02	8	46.6269	76.3367	
2.07E+02	9	57.7902	99.7846	
2.07E+02	10	62.7782	167.503	
2.07E+02	11	63.7469	-79.137	3
2.07E+02	12	70.3663	-24.972	
2.07E+02	13	67.6466	142.744	
2.07E+02	14	56.0573	65.3306	
2.07E+02	15	61.7173	15.1391	
2.07E+02	16	68.983	224.236	
2.07E+02	17	68.0813	47.7569	
2.07E+02	18	65.1849	-32.156	5
2.07E+02	19	68.9633	-5.2751	
2.07E+02	20	70.6929	143.799	
2.07E+02	21	69.7048	-123.37	1
2.07E+02	22	65.7603	677.4	
2.07E+02	23	58.19	196.064	
2.07E+02	24	57.1071	112.46	
2.07E+02	25	65.3285	187.767	
2.07E+02	26	59.4643	-314487	
2.07E+02	27	61.8369	46.2256	
2.07E+02	28	71.7156	143.921	
2.07E+02	29	67.7943	-73.545	4
2.07E+02	30	66.1547	-15.941	

An advantage of the HS algorithm is its rapid convergence due to its structure. Its disadvantage is trapping in a local optimum because of less diversified search in final iterations. To overcome this problem Tabu search is used in the final iterations of the algorithm. In this study, 50 HS iterations are selected. When the solution of this algorithm approached the optimal solution, the Tabu search algorithm should start. Figure 1 compares the hybrid algorithm presented by this study and HS algorithm. Obviously, convergence speed in approaching the solution is greater in the hybrid algorithm. Figure 2 depicts the difference clearer. If the number of iterations is changed to 3000, it will take the HS algorithm about 10 minutes (and more than ten minutes in some cases) to solve the problem whereas the hybrid algorithm suggested here will take about 4 minutes.

Through the search by the hybrid algorithm, UPFC installation was located based on the lack of reactive power. This means that the buses with the most reactive power shortage are candidate for UPFC installation. According to Table II, 5 buses are considered as candidates for a UPFC installation. In this algorithm, candidates for UPFC installations correspond to priorities of the buses 21, 4, 11, 29 and 18. Both algorithms lead to a same solution. However, total loss is 2.0732*107 in hybrid algorithm and 2.1254*107 in HS algorithm, suggesting that the hybrid search algorithm is more able to minimize the loss than the HS algorithm.

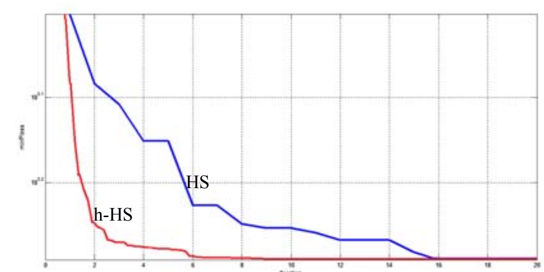

Fig. 1. HS algorithm and hybrid algorithm (improved HS with Tabu search)

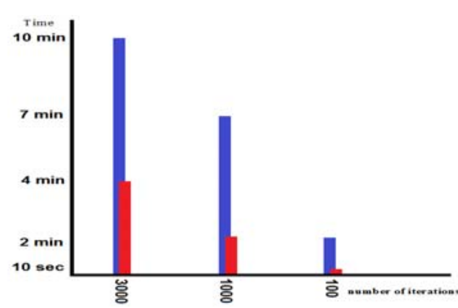

Fig. 2. Time versus the number of iterations

V. CONCLUSION

Considering the disadvantage of the HS optimization algorithm, a hybrid HS and Tabu search algorithm was suggested in this study. HS algorithm can be improved by another algorithm, such as Tabu search, employed to find local optimums (in neighborhood to the global optimum). Grid loss minimization and the best location for a UPFC installation in a IEEE 30-bus grid was considered as a focus of the comparison. This study showed that Tabu search algorithm is very effective in improving HS algorithm.

REFERENCES

[1] N. G. Hingorani, L. Gyugyi, Understanding FACTS: Concepts and technology of flexible AC transmission systems, Wiley-IEEE Press, 1999

[2] A. Kazemi, A. Parizad, H. R. Baghaee, "On the use of harmony search algorithm in Placement of FACTS Devices to improve power system security", IEEE EUROCON '09, pp. 570-576, 2009

[3] X. Wang, X. Z. Gao, K. Zenger, An Introduction to Harmony Search Optimization Method, Springer Cham Heidelberg New York Dordrecht London, 2014.

[4] Z. W. Geem, J. H. Kim, G. V. Loganathan, "A new heuristic optimization algorithm: harmony search", Simulation, Vol. 76, No. 2, pp. 60–68, 2001

[5] M. K. Saka, "Optimum design of steel skeleton structures", in Music-Inspired Harmony Search Algorithm, ed. by Z.W. Geem, Springer, Berlin, pp. 87–112, 2009

[6] C. H. Guan, Y. Cao, J. Shi, "Tabu Search Algorithm for Solving the Vehicle Routing Problem", IEE Third International Symposium on Information Processing, pp. 74-77, October 15 - 17, 2010

[7] M. H. Navidadham, "Solving generalized vehicle routing problem by parallel universes and Tabu search", IEEE 6th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Dallas-Fortworth, TX, USA, pp. 1-7, 2015

[8] D. Shilane, J. S. Martikainen, S. Dudoit, S. J. Ovaska, "A general frame-work for statistical performance comparison of evolutionary computation algorithms", Information Sciences Journal, No. 178, No. 14, pp. 2870-287, 2008