

Sympathetic Tripping in Meshed Distribution System and Methods to Mitigate its Effects in Power System

Ms. Shweta S. Karanje
Electrical Engineering Department
Walchand College of Engineering
Sangli, India

Dr. Ramchandra P. Hasabe
Electrical Engineering Department
Walchand College of Engineering
Sangli, India

Abstract— Occurrence of fault is a very common phenomenon in electrical power system network. When this happens, the relay associated with the fault is expected to trip. If the primary relay does not trip and a relay that is not associated with the fault operates, this causes shedding of an unfaulted line which is not desirable. This phenomenon of incorrect relay operation is named as sympathetic tripping. Since this is an unwanted phenomenon, this issue needs to be mitigated. This paper hence discusses the concept of sympathetic tripping, its effects on power system and methods to mitigate it. A study of differential evolution algorithm and genetic algorithm is also included in the paper that can help to mitigate sympathetic tripping. Respective flowcharts of the algorithms are also included for the ease of understanding.

Keywords—Sympathetic tripping, differential evolution algorithm, genetic algorithm.

TERMINOLOGY USED

PS	Plug setting multiplier of the relay.
TMS	Time Multiplier Setting.
P ₁	Primary Relay to the close end of the fault.
P ₂	Primary Relay to the distant end of the fault.
B ₁	Backup relay of P ₁ .
B ₂	Backup relay of P ₂ .
O	Other relays in the network.
I _{sc}	Short Circuit current.
CTR	Current Transformer Ratio.

I. INTRODUCTION

Modern society is contingent heavily on continuous and dependable availability of electricity. A thoroughly coordinated protection scheme is an integral requirement to operate a power system with maximum reliability. Computer and telecommunication networks, railway networks, continuous process commerce and life support structures are just a few applications that cannot operate without a highly efficient source of electric power. In addition to this, the huge number of household electricity users whose life is thrown out of gear, if the electric supply is interrupted. Thus, the importance of sustaining continuous electric supply cannot be overstated.

No power system can be planned in such a way that it would on no occasion fail. In the philosophy of protection engineers, these fiascos are called as faults. It is important to prevent these faults. And incase if they occur, it is even more important to mitigate the consequences of these faults. A decent protection scheme eliminates only the least probable percentage of the network when fault occurs in order to supply to the rest of the

healthy system without being impacted by the fault. The ill effects of the fault are lessened by quickly isolating the faulty part from the rest of the system, thus restrictive the disturbance footmark to as small area as possible. Unwanted operation of relay is called as sympathetic tripping and this phenomenon can be seen reported first in Power System Relaying Committee Report [1].

A commercial and effective protection system for meshed or multi-sourced power systems needs directional overcurrent relays (DOCRs). The action of DOCRs depends on time multiplier setting (TMS) and plug setting multiplier (PSM). For reliable and efficient operation of relays, sufficient planning of relay coordination in meshed distribution scheme is vital. Relay coordination problem comprises of coordinating the operation of primary and backup relays. The primary relay is accountable for immediate clearance of fault in a definite time as decided by the relay features and in the incident of its failure, backup protection must work for the guaranteed segregation of faulty section. The fault must be secluded in such a manner, that it only isolates the faulted part, thus keeping the power continuity over the healthy portion of the system.

Protective relays should identify the fault with the help of voltage and current transformers, and selectively eliminate only the faulty section from the remaining system by tripping a suitable number of circuit breakers. Overcurrent relays respond to the magnitude of the input current and work when this magnitude surpasses the predetermined level (pickup level). When this level is surpassed, the relay closes its trip contacts and circuit breaker trip coils are energized. This the relay has to perform with extreme sensitivity, selectivity and swiftness. In a power system network, faults are not a daily event. A typical relay, hence devotes its entire lifetime monitoring the power system network. It must, hence be prepared all the while in expectation of a fault. Thus, relaying is similar to insurance against harm due to faults.

The occurrence of Sympathetic tripping takes place in the following cases-

1. Case-1: The backup relay or another relay operates before primary relay.
2. Case-2: Other relay operates even before the backup relay.

Both of these cases cause de-energization of healthy lines which is unacceptable.

II. METHODS TO MITIGATE SYMPATHETIC TRIPPING

The sympathetic tripping phenomenon is a complex problem and hence it can be solved in many ways. Study of the literature shows various methods to deal with this problem. One such method is by using current limiting devices in the network to limit the value of fault current. This protects the network from unwanted relay trips [3]. A nonlinear sequential quadratic algorithm is also used to solve this issue [4]. In order to optimize the value of TMS and PSM various scholars have also suggested using optimizing algorithms that will in turn reduce the time of operation of the relays. This will then prevent unwanted relay trips. Various algorithms used are Particle Swarm Optimization [5], bio geography based algorithm [6], teaching-learning based optimization [7], seeker algorithm [8], modified differential evolution algorithm [9], differential evolution algorithm [2], Genetic algorithm [10], hybrid genetic algorithm [11]. In all of the above literature, the authors have suggested and compared optimized value of TMS and PSM so that the primary relay operates first and then the backup relay operates after following a certain time interval also referred as CTI i.e., Co-ordination Time Interval.

In this paper a detailed study of differential evolution algorithm and genetic algorithm is carried out. The steps involved in the working of these both algorithms are elaborated. A flowchart of each is also designed which sheds light on the major differences between the two algorithms.

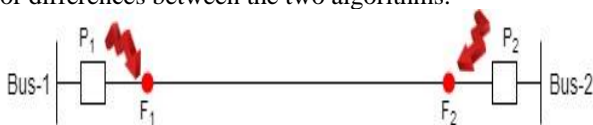


Fig. 1. Placement of Overcurrent Relays

Fig. 1 shows placement of directional overcurrent relays. F_1 and F_2 are the fault locations on the bus. Relay P_1 and P_2 are primary relays to fault F_1 and F_2 respectively. This acts as primary protection for the system. Whenever there occurs a fault at location F_1 , the relay P_1 should operate in minimum time possible. Similarly, if fault occurs on location F_2 , the relay P_2 must operate in minimum time possible and segregate the faulty part of the system. At certain times, primary protection may fail to operate. This could be due to failure of current transformer or voltage transformer or relay or the failure of operation of circuit breaker. In such case there must be a second line of protection. Therefore, it is a typical practice to offer another zone of protection that must operate and segregate the faulty element in case primary protection does not operate. The backup protection must pause for the primary protection to function, prior to issuing trip command to its connected circuit breaker. There will be intentional delay period between the operation of both relays. The relay close to the fault will experience maximum fault current and hence will trip faster. The delay between the operation of both relays is called as co-ordination time interval that comprises of overshoot time and time of operation of circuit breaker of primary relay.

The time of operation of the relay depends on the values of the following parameters:

1. Plug Setting Multiplier. (PSM)
2. Time Multiplier Setting. (TMS)
3. Fault current at that point. (I_f)

Out of these three parameters, since we cannot control the value of the fault current flowing through the circuit, we will be optimizing the values of Plug Setting Multiplier and Time Multiplier Setting.

Time Multiplier Setting: The time of operation of the relay can be set at a desired value. In induction disc type relay, the angular distance by which the moving part of the relay travels for closing the contacts can be adjusted to different operating times. There are 10 steps in which time can be set. The term time multiplier setting (TMS) is used for these steps of time settings. The values of TMS are 0.1, 0.2, ..., 0.9, 1.

Plug Setting Multiplier: The current above which the over-current relay must operate can be set. The relay operates when the measured current surpasses this value of pick up current. There are a number of tapping available on the current coil for current setting. The relay operation needs certain flux and ampere turns. The current setting of the relay is selected by altering the number of turns of current coil by means of plug setting. The plug setting can directly be specified in amperes or indirectly as ratio of rated current. The actual r.m.s. current flowing in the relay is stated as a multiple of pickup current is known as plug setting multiplier (PSM).

A. DIFFERENTIAL EVOLUTION ALGORITHM

The Differential Evolution (DE) algorithm is a meta-heuristic algorithm developed in the year 1995 by Storn and Price. Rather than having local solutions, it obtains the best global solution by using a stochastic search process [12]. DE optimizes a problem by maintaining a population of solutions and, at the same time, by combining existing solutions according to its formulae, it generates fresh solutions and retains the solution that best fits the problem of optimization. In this way the optimization problem is viewed as a black box that merely provides a degree of supremacy of a given candidate solution and thus the gradient is irrelevant. Differential evolution is a process in evolutionary calculation which solves a problem by recursively optimizing a solution with respect to a defined degree of accuracy. Typically, these strategies are known because metaheuristics because they make relatively less or zero assumptions about the problem and can look for vast spaces of solutions. Yet metaheuristics like DE aren't promising to get an optimal solution. DE is used for multidimensional real-established functions, but again does not use the gradient of the problem that is being optimized, which means that DE does not demand that the optimization problem be differentiable, as is the case with classic optimization approaches such as gradient descent and quasi-newton procedures. As a consequence, DE can also be used to optimize problems that aren't quite continuous, are noisy and shift over time.

The major steps involved in this algorithm are:

1. Initialization
2. Mutation
3. Crossover
4. Selection

The thorough working of the algorithm is given below:

1) *Entering the inputs:* The first step in the algorithm is giving adequate inputs. The data that is given as inputs include the size of the population, maximum number of generations for which the optimization is to be carried out. The data regarding

the fault current values, CTR, maximum and minimum limits of TMS and PSM. The population size does not vary in the entire course of optimization process of the algorithm.

2) *Initialization*: The initial population is chosen such that it covers the entire search space. Here since the optimization of PSM and TMS is considered, those parameters will be initialized considering their respective limits.

3) *Mutation*: New parameter vectors are generated in this step of the said evolutionary algorithm. This is done to obtain a third vector in addition to the previously selected random vectors. The difference of two population vectors are added to third vector. This process is called as mutation. The vector obtained is termed as target vector.

4) *Crossover*: The mutation process gives rise to a mutated vector. Now, another pre-determined vector is mixed with the previously obtained vector. The vector obtained in this process is called as trial vector. This method of mixing the parameters is called as crossover. This is an important step because the trial vector obtained in this step will replace the target vector obtained in the previous step.

5) *Selection*: The value of target vector and trial vector is then substituted in the main objective function. The value of the objective function is termed as cost function. The vector that obtains the minimum value of cost function during minimization operation is then selected. This process is called as selection.

6) *Termination criteria*: The algorithm halts when the maximum number of generations is obtained or if the objective function reaches some predetermined threshold value. If the number of generations are not completed but the threshold value is reached, this as well stops the algorithm. And whatever value is attained is considered as the solution of the problem.

The differential evolution algorithm helps in obtaining global best solution of any given optimization problem.

Fig. 1 shows the steps that are involved in the algorithm and hence gives us an idea of the flow of the process.

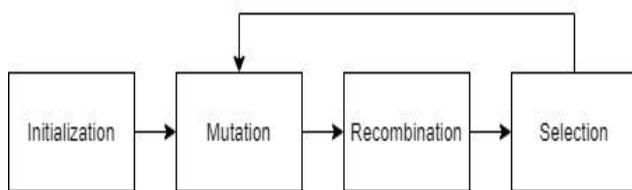


Fig. 2. Steps of Differential Evolution Algorithm

Fig. 2 shows the flowchart of the differential evolution algorithm. Here all the operations are shown in sequential manner and all the equations that are in the algorithm are also expressed. Each of the above-mentioned steps are carried out in the form of these equations and a suitable solution is obtained.

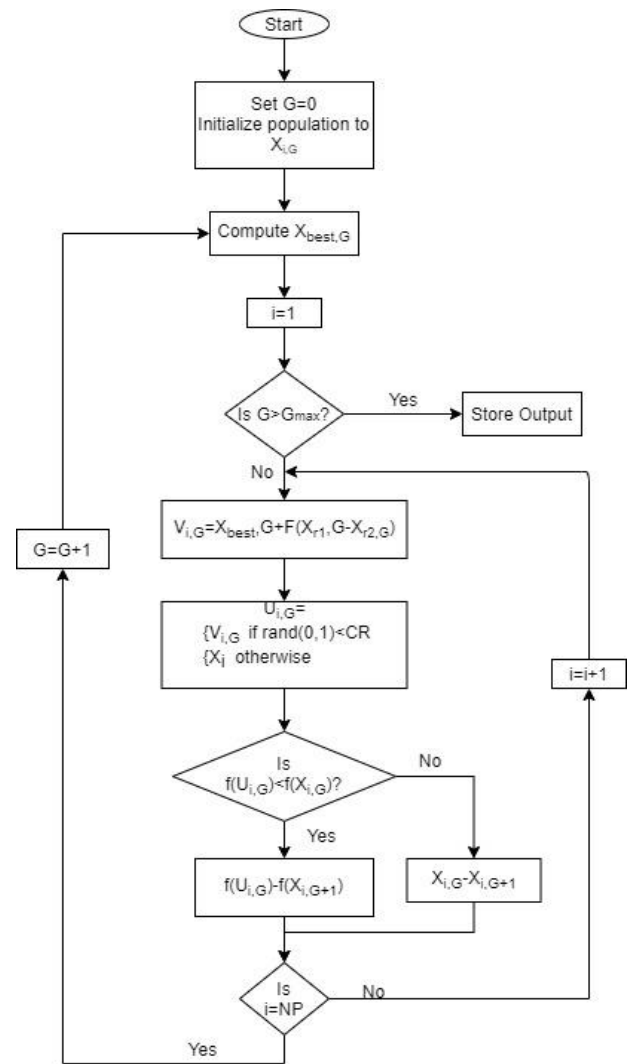


Fig. 3. Flowchart of Differential Evolution Algorithm

B. GENETIC ALGORITHM

Charles Darwin's theory of natural evolution forms the basis of Genetic Algorithm (GA). In computer science and research, a genetic algorithm is a metaheuristic moved by the procedure of natural assortment that have its place in the bigger category of evolutionary algorithms. GA is usually used to produce good quality solutions to optimize the problems by counting on biologically stimulated processes such as mutation, survival of fittest crossover and selecting the best amongst the possible solutions. John Holland presented GA in 1960 built on Darwin's theory of evolution, and David E. Goldberg added further to GA in 1989. GA imitates the procedure of natural selection where the fittest entities are carefully chosen for reproduction so as to generate children of the subsequent generation. GA is a way to solve both bound and unbound problems of optimization based on a natural selection process that imitates biological evolution. The algorithm repetitively adapts a population of discrete solutions. At individual step, the GA randomly chooses individuals from the present population and uses them as parents to yield the offspring for the succeeding generation. The population "evolves" over successive generations in the direction of a best solution. Evolution usually starts with a population of randomly generated individuals and is an iterative process, with the

population being called a generation in each iteration. The fitness index of each member in the population is measured in each generation; fitness is normally the rate of objective function in solving the issue of optimization. The individuals that are extra fit are carefully selected from the present population, and the genome of each individual is used to generate a new generation. In the subsequent iteration of the algorithm the latest generation of solutions is further used. The algorithm normally stops when either the pre-specified number of generations has been established, or an appropriate fitness level has been reached for the given population.

1) *Initialization:* Since, this is a problem to obtain relay settings, several sets of relay settings are randomly generated. This is done by making sure that each of the relay setting is within their set limits. This is called as Parents Chromosome Pool. One chromosome consists of one relay setting. In genetic algorithm, chromosome is the key variable. The number of such sets is called the size of problem.

2) *Evaluation:* In this step, the fitness of each chromosome is evaluated. The value of each chromosome is substituted in the objective function and the value that is best suited to the user is then evaluated further.

3) *Selection:* Since we have obtained many different values of parents it becomes mandatory to choose the values that are most optimal. The parents that give most optimal values of objective function are chosen. This concept is called as elitism. As these parents provide more optimal solution, the chances of their survival in the next generation are heightened. Roulette wheel is one such method of selection that can be used to select parents to obtain offsprings.

4) *Mutation:* Crossover operation results in reproduction of offspring. This is also termed as mutation. In this process, the main aim is to create a new children chromosome pool. The members of the children chromosome pool however must satisfy the conditions of optimal objective function values. If the offspring fail to satisfy the constraints then they are eliminated from the pool. The quality of the offspring can be improved by changing the parents for crossover and mutation process. The offspring then indicate new settings of the relay. This process is continued till the total number of generation criteria is met.

5) *Selecting either parents or children for next generation:* The parents or the children that go forward in next generation depends on fitness value of each of these. The parents or the offspring whichever generates the optimal value is allowed in the next generation.

6) *Termination:* After the predefined number of generations is reached, the algorithm will automatically terminate. Generally, the more are the generations, the more precise the solution will be. The number of generations is also subject to the complexity of the system and size of the population.

Fig. 4 shows the flowchart of GA in which the steps involved are given step by step.

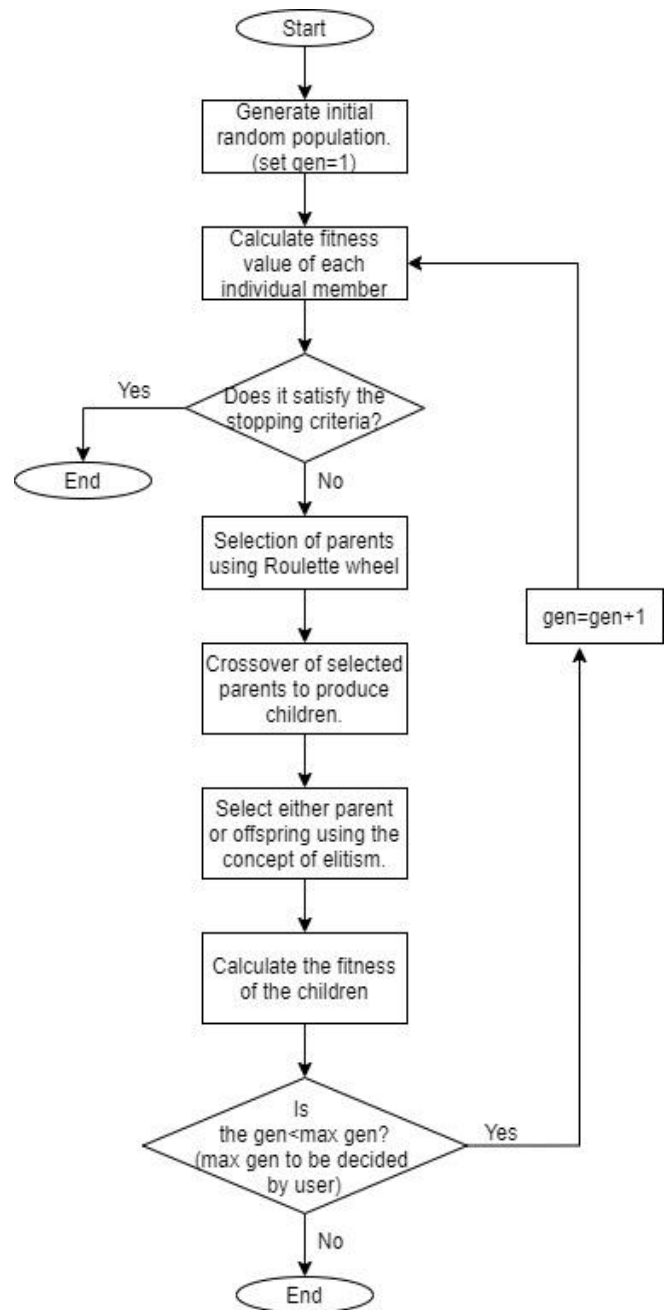


Fig. 4. Flowchart of Genetic algorithm

III. COMPARISON BETWEEN DIFFERENTIAL EVOLUTION AND GENETIC ALGORITHM

Both differential evolution algorithm and genetic algorithms are evolutionary algorithm. However, there are a few differences in the working of both these algorithms. Since the differential evolution calculates the difference between the two selected values and then keeps updating and improving it further, hence the results obtained through the differential evolution will be more accurate as compared to genetic algorithm. Certain algorithms as mentioned in the literature has an issue of getting trapped in local minima. However, this is not an issue with differential evolution algorithm. It will always give global optimum value or global best solution.

REFERENCES

- [1] I. P. S. R. C. Report, "Distribution line protection practices industry survey results," *IEEE Trans. Power Del.*, vol. 10, no. 1, pp. 176–186, Jan. 1995.
- [2] A. Sharma and B. K. Panigrahi, "Framework arrangement of directional relays in meshed networks based on differential evolution algorithm," in *Proc. IEEE 6th Int. Conf. Power Syst.*, Mar. 2016, pp. 1–6.
- [3] A. Othman, A.-Z. Mamdoh, and Y. Kamal, "Prevention of sympathetic tripping phenomena on power system by fault level management," in *Proc. IEEE/PES Tran. Distrib. Conf. Expo.*, Apr. 2008, pp. 1–14.
- [4] D. Birla, R. Maheshwari, and H. Gupta, "An approach to tackle the threat of sympathy trips in directional overcurrent relay coordination," *IEEE Trans. Power Del.*, vol. 22, no. 2, pp. 851–858, Apr. 2007.
- [5] H. Zeineldin, E. El-Saadany, and M. Salama, "Optimal coordination of overcurrent relays using a modified particle swarm optimization," *Electr. Power Syst. Res.*, vol. 76, no. 11, pp. 988–995, 2006.
- [6] M. Ezzeddine, R. Kaczmarek, and M. Iftikhar, "Coordination of directional overcurrent relays using a novel method to select their settings," *IET Gener., Transm. Distrib.*, vol. 5, no. 7, pp. 743–750, Jul. 2011.
- [7] F. Albasri, A. Alroomi, and J. Talaq, "Optimal coordination of directional overcurrent relays using biogeography-based optimization algorithm," *IEEE Trans. Power Del.*, vol. 30, no. 4, pp. 1810–1820, Aug. 2015.
- [8] T. Amraee, "Coordination of directional overcurrent relays using seeker algorithm," *IEEE Trans. Power Del.*, vol. 27, no. 3, pp. 1415–1422, Jul. 2012.
- [9] R. Thangaraj, M. Pant, and K. Deep, "Optimal coordination of overcurrent relays using modified differential evolution algorithms," *Eng. Appl. Artif. Intell.*, vol. 23, no. 5, pp. 820–829, 2010.
- [10] F. Razavi, H. A. Abyaneh, M. Al-Dabbagh, R. Mohammadi, and H. Torkaman, "A new comprehensive genetic algorithm method for optimal overcurrent relays coordination," *Electr. Power Syst. Res.*, vol. 78, no. 4, pp. 713–720, 2008.
- [11] P. Bedekar and S. Bhide, "Optimum coordination of directional overcurrent relays using the hybrid GA-NLP approach," *IEEE Trans. Power Del.*, vol. 26, no. 1, pp. 109–119, Jan. 2011.
- [12] R. Storn and K. Price, "Differential evolution a simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, 1997.