Genetic Algorithms in Loading Pattern Optimization

Aylin YILMAZBAYHAN, Mehmet TOMBAKOGLU*, Kürsat B. BEKAR, A. Özgür ERDEMLİ
Hacettepe University Nuclear Engineering Department,
06532 Beytepe Ankara, Turkey
* mt@nuke.hacettepe.edu.tr

ABSTRACT
Genetic Algorithm (GA) based systems are used for the loading pattern optimization. The use of Genetic Algorithm operators such as regional crossover, crossover and mutation, and selection of initial population size for PWRs are discussed. Antithetic variates are used to generate the initial population. The performance of GA with antithetic variates is compared to traditional GA. The results of multi-cycle optimization are discussed for objective function taking into account cycle burn-up and discharge burn-up.

INTRODUCTION
The main goal of in-core-fuel management activities is to meet the design objectives. One of them is the core excess reactivity to ensure energy requirement. Safety is another major concern during the operation of a nuclear power plant, and requires the knowledge of power distribution and depletion characteristics of the fuel assemblies from the beginning-of-cycle (BOC) through the end-of-cycle (EOC). Once the energy demand is specified for a given period, the core excess reactivity calculations can be performed. Then other unknowns, such as amount and enrichment of the fresh fuel assemblies, fraction of the depleted assemblies to be removed, burnable poison (BP) requirements and core loading pattern map, must be determined. Such calculations are required to optimize in-core fuel cycle under some constraint to satisfy the utility's demand. To perform these calculations, optimum loading pattern map needs to be determined.

In the last 3 decades, considerable work has been completed employing several optimization techniques in determining the core-loading pattern which minimizes the fuel cost. With the emergence of artificial intelligence tools and further advances in computer performance and architecture, adaptive optimization techniques were developed. However, these adaptive methods such as simulated annealing and genetic algorithms need to evaluate large numbers of trial loading patterns. One of the drawbacks of these techniques is the computational cost which mainly depends on the technique used to obtain core power distribution and the total number of trial loading pattern evaluation.

Genetic Algorithm, first introduced by Holland in 1970’s is one of the stochastic optimization techniques that have become popular in the last decades. By using Genetic Algorithm (GA), Parks developed in-core fuel management strategy that can be used to optimize more than one
A loading pattern with minimum feed enrichment, maximum burn-up and power peaking factor under given constraint was sought. DeChaine and Feltus used expert knowledge concept developed at The Pennsylvania State University to create an initial population of fairly good solutions for GA optimization system. Using expert knowledge for creation of initial population, they found the best population more rapidly. Yamamoto compared genetic algorithms, simulated annealing, direct search and binary exchange methods for in-core fuel management optimization. He also tried Genetic Algorithm + Binary Exchange and Binary Exchange + Direct Search hybrid optimization methods. Yamamoto and Kanda compared equilibrium and successive multi cycle optimization results with the simulated annealing method for in-core fuel management of PWRs. They developed a multi cycle optimization system using simulated annealing method.

In this study, we shall be concerned with optimization for in-core fuel management of PWR's via Genetic Algorithm. Mainly, we address the question of population size and diversity problem in GA. The optimum size of the initial population, the crossover and mutation operators used to obtain new members of the population are discussed. Moreover, the different selection rules to generate new members and use of antithetic variates are also introduced and primary results are given.

**Genetic Algorithms for in core fuel management of PWRs**

In-core fuel management optimization is a difficult problem for traditional optimization methods. This is because the arrangement of fuel assemblies in a reactor core is a discrete problem without direct derivative information. It is based on biological genetics of a group of trial solutions called population. GA does not require derivative information and is insensitive to the problem and works well with discrete functions.

In GA, there are some specific terms like individual, chromosome, gene and fitness. In our study, individual is represented by a chromosome which is a 1/8 symmetric core loading pattern, and a gene denotes the type of assembly in the 1/8 symmetric core loading pattern and its fitness is the value of assigned objective function. During the evaluation of loading pattern with GA, neutronic calculations are performed with Reload Power Method (RPM) computer code developed by Sauer and Driscoll. The RPM code allows the user to specify up to 36 different types of fuel assemblies with two-dimensional nodal calculations using 1/8 core symmetry. RPM is used to calculate the cycle burnup and the power map at the beginning of cycle and at the end of cycle according to linear reactivity model. The results of RPM code, i.e., cycle burn-up and power distribution, are utilized to determine fitness values for each individual loading pattern. The flow diagram of GA is shown in Figure 1.

GA has three genetic operators: selection, crossover and mutation. The selection operator selects the individuals from current population according to their fitness. The crossover operator pairs the selected individual's chromosomes to create new individual with different patterns. Finally,
the mutation operator makes small random changes on the selected chromosomes. That maintains the diversity of the population and allows the search to cover the entire search space. Fitness value of each individual generated by crossover and/or mutation is calculated and compared with the fitness values of other individuals in the current population. If the fitness value of the new individual is larger than the smallest fitness value of the population, individual having smallest fitness value is killed, the new individual becomes member of the population, and population size remains constant. Individuals generated by crossover and/or mutation may have the same loading pattern compared with other members. In that case, updating is not performed.

**Genetic Algorithm Results**

**Population size**

The chromosome of each individual in the population is a fuel-loading pattern. There are 31 fuel assembly locations in the 1/8 symmetric core having a total of 157 assemblies. Therefore the number of the genes in the chromosome of the individual is equal to 31. There are three kinds of genes ie, fuel batches, having different initial reactivities. The chromosome of each individual is formed randomly. An example of the one dimensional and two dimensional chromosome is shown in Figure 2. During the examination of the chromosomes of the individuals, number of assemblies of each batch in the full core must be conserved. The chromosomes of all individuals are formed with respect to that rule and their fitness values are evaluated using RPM code.

![Flow diagram of GA optimization system.](image)

**Figure 1.** Flow diagram of GA optimization system.
The GA algorithm is used with different initial population sizes. In Figure 3, average burnup of the population is plotted as a function of generation number. The results show that when the population size increases the number of individuals having the same patterns increases. Population size of 32 and 64 give better results compared with population sizes of 16 and 128.

**Figure 2** One and two dimensional chromosome.

**Figure 3** Effect of Number of Individuals on Optimization.
Initial population generated with antithetic random vectors

To obtain more diverse initial population, antithetic random vectors are used. Chromosomes of the individuals belonging to initial population are generated using two different random vectors, \( z_1 \) and \( z_2 \). After that four random vectors are generated from these vectors. These new random vectors are called the antithetics of \( z_1 \) and \( z_2 \) vectors. The formation of antithetic random vectors is shown in Figure 4.

![Figure 4. Antithetic random vectors used to generate initial population.](image)

The aim of using the antithetic random vectors is to make the population more diverse. As a result, individuals with different loading patterns are utilized to obtain the new members of the population with better fitness values. The use of antithetic random vectors on optimization is presented in Figure 5.

![Figure 5. Effect of Antithetic on the Initial Population](image)

Figure 6 presents the variance of the population as a function of generation number. It can be seen that initial population formed by using two antithetic random vectors is more diverse than other initial populations. Moreover, variance of the population decreases more rapidly than
Selection rules

For crossover and mutation, two individual are selected from current population according to their fitness values. This selection process is performed by using four different rules: First Rule: The individuals are selected randomly from the current population. Second Rule: The selection probability of each individual is evaluated according to their fitness values. Third Rule: One of the selected individual is the one with best fitness value. The second one is selected randomly. Fourth Rule: The individuals in the population are divided into two or four groups according to fitness values of the individuals. After that, a selection probability is assigned to each group. The group formed by the individuals having better fitness values must have the higher selection probability. During selection, first the group is selected according to selection probability, and then an individual is selected from the group with equal probability.

In Figure 7, burnup average of the population is given as a function of generation number for different selection rules. When fitness average of population increases, fitness values of the individuals in the population gets closer to each other. Therefore the selection probabilities of individual become nearly equal. In order to prevent this, population is divided into two or four groups. Instead of setting selection probabilities for each individual, selection probability is set for each group. As the result of the comparison of selection methods, it can be seen that method 4 finds best population more rapidly than others.
Crossover operators

There are three methods used for crossover:

One Point Crossover: In order to make one point crossover, a point in the one dimensional chromosome is selected randomly and at that point, chromosomes are divided into two parts and the first parts of the chromosomes are interchanged.

Two Point Crossover: Two points are randomly selected in the one-dimensional chromosomes. The parts of the chromosomes between these points are interchanged.

Regional Crossover: The regional crossover is applied on two-dimensional chromosomes. It is based on interchanging two randomly selected lines between two chromosomes. In order to make a regional crossover, a gene in the two-dimensional chromosome and length of the line to be interchanged is selected randomly. This line is formed by the genes of the selected chromosomes.
The genes forming the line are selected randomly from the non-selected neighbours of the previously selected gene. After one of these crossover operators is applied on the selected chromosomes, the number of assemblies of each batch in the full core becomes unequal. By ranking the assemblies with respect to their reactivities, number of assemblies in each batch in the full core is made equal to each other. In Figure 8, burnup average of the population is given as a function of generation number for each Crossover operator. The results show that the regional crossover operator is the most suitable one compared with others.

![Figure 8. Comparison of Different Crossover Operators](image)

**Mutation operator**

By using mutation operator, some genes are randomly mutated in the chromosome. The frequency of mutation and the location of the mutated genes are the important topics. In order to examine the effect of mutation, different techniques are used. The effect of mutation can be seen in Figure 9. During the generation of new members, fitness values of individuals get closer to each other and the number of new individuals accepted into the population decreases. When the probability of mutation is increased, the number of accepted new individuals having different data increases. Therefore, the probability of mutation is increased as a function of generation number. The initial mutation probability needs to be determined. In Figure 10, the results are presented for initial mutation probabilities of 10% and 30%.

![Figure 9 Effect of mutation](image)
Use of antithetic variates in Crossover and mutation

Antithetic random vectors are used in the crossover and mutation operators. In crossover operator, antithetics of the random vectors are used and four different individuals are generated from two selected individuals at the end of the crossover. The effect of using antithetic random vectors in Figure 11.

Multi cycle optimization

The fresh fuel assemblies located in the core are loaded during several proceeding cycles. So burnup and other calculated variables depending on the loading pattern are not the same for each cycle. Consequently, the aim of the multi cycle optimization is to find the equilibrium loading pattern that has the maximum discharge burnup and ppf less than 1.48. At the end of each cycle, the assemblies having the least reactivity are removed from the core. The assemblies of the second batch are moved to locations of these and the assemblies of the third batch are moved to the locations of the second ones. The locations of assemblies are determined
randomly. After that, remaining locations are filled with fresh fuel assemblies. This shuffling process is repeated after each cycle. The optimization results with cycle burnup and discharge burnup dependent objective function are given in Figure 12.

**Figure 12. Effect of Fitness Function on Discharge Burnup Optimization**

When discharge burnup, instead of cycle burnup, is taken into consideration in the objective function, the average discharge burnup increases.

**SUMMARY**

In the present work, we have developed GA model for in-core fuel management of PWR’s. We introduced antithetic variates as an acceleration technique. The use of different genetic operators on the PWR optimization problem were discussed. The performance of GA with antithetic variates is compared with standard GA. Finally, the results of multi-cycle optimization are discussed for two objective function. In the first one, the cycle burnup and in the second one, discharge burnup was used as an objective function with power peaking constraint. The results show that in the multi-cycle optimization problems discharge burnup gives better result.

**REFERENCES**