Annals of Nuclear Energy 102 (2017) 37–46

Contents lists available at ScienceDirect

Annals of Nuclear Energy

journal homepage: www.elsevier.com/locate/anucene

Using particle swarm optimization algorithm to search for a power ascension path of boiling water reactors

Tzung-Yi Lin *, Jau-Tyne Yeh, Weng-Sheng Kuo

Nuclear Engineering Division, Institute of Nuclear Energy Research, 1000, Wenhua Rd., Jiaan Village, Longtan Township, Taoyuan County 32546, Taiwan, ROC

ARTICLE INFO

Article history:
Received 3 December 2015
Received in revised form 12 December 2016
Accepted 16 December 2016

Keywords:
Particle swarm optimization algorithm
Power ascension of boiling water reactors
Operating point searching strategy
ASCENTB

ABSTRACT

During the power ascension, the operating point is based on core power, core flow, control rod pattern, and concentrations of fission product, such as xenon. The thermal limits and fuel conditioning at the operating point should meet the constraints. ASCENTB is an automatic power ascension path searching program for boiling water reactors. The control rod movement is searched for by the particle swarm optimization (PSO) algorithm. The operating points of one control rod withdrawal sequence are based on the PSO1 or PSO2 strategy. The results of ASCENTB for two selected cycles are comparable with the power plant records.

1. Introduction

According to the thermal hydraulic characteristics of a Boiling Water Reactor (BWR), the core status at a high power and low flow condition is located in an unstable region for operation. Therefore, the operable domain limits a reactor operator to do the power ascension. A typical power-flow map of a BWR/6 is illustrated in Fig. 1. Before the startup of a nuclear reactor, engineers must manually accomplish the power ascension path planning for the operator in advance. However, the engineers can only predict the power ascension path roughly. In addition, to shorten a working time for the planning of the power ascension path and improve the accuracy of the predicted path to the actual operating points, it is usually done by the experienced engineers. Junior nuclear engineers will need more practices to accumulate experiences in the power ascension path planning.

During the power ascension process, the guarantee of the fuel rod integrity is the most important work. Some thermal limits are usually used as the monitoring parameters, i.e., the minimum critical power ratio (CPR), the maximum linear heat generation rate (LHGR) and the maximum average planar linear heat generation rate (APLGHR). The fuel pellet-cladding interaction (PCI) has to be carefully watched during the power ramp. A PCI calculation method mostly used in a core monitoring system of BWR is based on the pre-conditioning interim operating management recommendation (PCIOMR) Potts et al., 1994; Taiwan Power Company, 1022. In PCIOMR, the fuel conditioned state is defined as that the fuel pellet contacts well with the cladding. The fuel pellet power at this state is called the conditioned-state power (PCS), and PCS is always updated according to the present fuel pellet power. Consequently, the difference between the present fuel pellet power and PCS (P-PCS) is an important monitoring parameter which usually dominates the power ascension path.

Due to the economic requirements of a nuclear power plant, it is preferred to operate the reactor to reach the rated power as soon as possible. However, raising power rapidly is difficult because of the restriction of PCIOMR and thermal limits. Using mathematical algorithms (Lee and Lin, 2007; Lin and Yen, 2010) to search for a detailed and proper power ascension path automatically is a good solution and can help junior engineers to do the power ascension path planning well. For a good power ascension path, more thermal limit margin can be obtained and the rated power can be reached quickly without violating or lowering PCIOMR.

The mathematical algorithms that are often used as the search engine include genetic algorithm (Holland, 1975), ant colony optimization algorithm (Colorni et al., 1991), tabu algorithm (Glover, 1989, 1990), harmony algorithm (Geem et al., 2001), simulated annealing (Kirkpatrick et al., 1983), particle swarm optimization (PSO) algorithm Kennedy and Eberhart, 1995; Shi and Eberhart, 1998, and so on. These algorithms were applied to search for an optimized fuel loading pattern and control rod pattern (Park et al., 2014; Kobayashi and Aiyoshi, 2003; Jagawa et al., 2001; Wang and Lin, 2013) mostly. Lee and Lin (2007) applied the genetic algorithm to the power ascension successfully. The ant colony optimization algorithm also worked well by Lin and Yen (2010). In this study, the PSO algorithm is used as the search engine of our in-house program, ASCENTB.

* Corresponding author.
E-mail address: joey.tylin@gmail.com (T.-Y. Lin).

http://dx.doi.org/10.1016/j.anucene.2016.12.019
0306-4549/

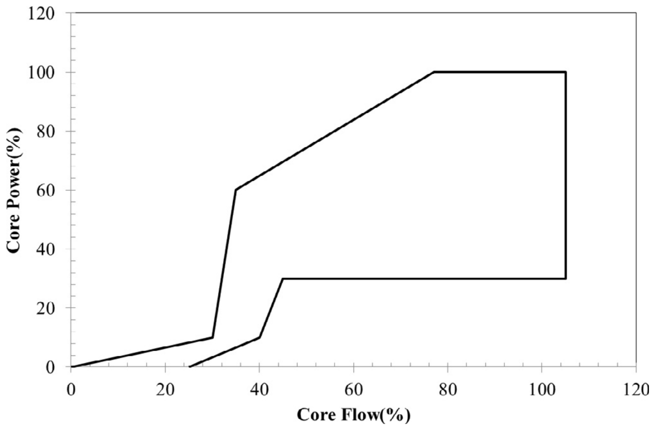

Fig. 1. Typical power-flow map of a BWR/6.

In Section 2, PSO is briefly described. Section 3 describes the implementation of PSO and the operating point searching strategy. Two cycles of a BWR/6 are used to demonstrate the capability of ASCENTB. The results are summarized in Section 4. Finally, Section 5 is the conclusion.

2. Particle swarm optimization algorithm

The particle swarm optimization algorithm, which was first proposed by Kennedy and Eberhart (1995) and Shi and Eberhart (1998), was used to simulate the group behavior. The swarm or group changes its direction during its movement due to the environmental impacts or the leader's command. According to this concept, the best solution of an interested problem can be searched for by changing the swarm direction.

The population (swarm) in PSO contains a lot of candidate solutions (particles). These particles would move to initial positions based on the random direction and speed. If the best one is selected as the leader by one fitness function, the swarm would be guided by it. In other words, the direction and speed of each particle are affected by the leader. Therefore, each particle would get its new position. Then a new best particle is chosen to guide the swarm. After a lot of iterations, the movement of the swarm would stop until it reaches the best or expected position. This position is the solution of the interested problem.

The movement of one particle is illustrated in Fig. 2. The new (iteration k + 1) speed of the particle (V_i^{k+1}) is updated according to the current (iteration k) speed (V_i^k), the current best position of this particle (S_i^{pbest}), and the current best position of the swarm (S^{gbest}). Eq. (1) defines how to update the particle's speed. After the speed is updated, the particle moves to a new position (S_i^{k+1}) from the current position (S_i^k) which is formulated in Eq. (2).

Some parameters which impact the updated speed are the inertia weighting factor (w) and the acceleration factors (c_1 and c_2):

- The inertia weighting factor can improve the converging speed. The suggested value (Shi and Eberhart, 1998) is 0.9–1.2, and the value of 1 is applied in this study.

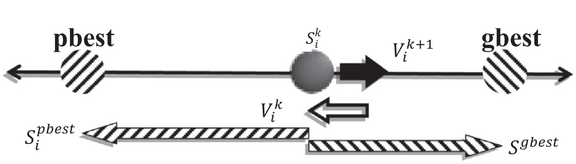

Fig. 2. Illustration of the particle movement.

- The acceleration factors are used to guide the particle to move to the best position, and the suggested value (Kennedy and Eberhart, 1995) of 2 is used in this study.

$$V_i^{k+1} = w \times V_i^k + c_1 \times r_1 \times (S_i^{pbest} - S_i^k) + c_2 \times r_2 \times (S^{gbest} - S_i^k), \quad (1)$$

$$S_i^{k+1} = S_i^k + V_i^{k+1}, \quad (2)$$

where
w is the inertia weighting factor of the particle,
c_1 and c_2 are the acceleration factors, and
r_1 and r_2 are the random numbers (0–1).

3. Implementation

3.1. Particle position definition

PSO is used to search for the control rod withdrawal sequence during the power ascension. The difference between a given initial control rod pattern and a given final control rod pattern is separated into many withdrawal steps, and in each step, a specified control rod (CRD) is withdrawn by 2 notches. The control rods at the symmetric locations to this control rod are also withdrawn by 2 notches at the same time to make the control rod pattern always quarter-core or one-eighth-core symmetric. Thus the total withdrawal step number *Nstep* for all control rods withdrawn to the final control rod pattern is decided. The control rod notch at each step is given an index, e.g. *A1*. All control rod notch indexes compose the particle position. The particle position description and an example of one control rod withdrawal sequence are illustrated in Fig. 3.

In Fig. 3, the pattern is one-eighth-core symmetric. Hence, only three control rods need to be defined for the notch index. The CRD A at location (7,9) is moved to notch 44 from 6, and there are 19 steps for this control rod. The notch indexes of CRD A are therefore defined as *A1* to *A19* The CRD B at (7,11) and CRD C at (9,11) require 10 and 23 steps, respectively. The notch indexes of CRD B are *B1* to *B10*; whereas, the notch indexes of *C1* to *C23* are used for CRD C. The total step, *Nstep*, is 52, and the particle position is one vector containing 52 elements, with the composition of each element being a notch index of the control rod.

An example of the control rod withdrawal sequence is also shown in Fig. 3. This control rod withdrawal sequence is the combination of the notch indexes of three control rods. In the first two steps, CRD A is withdrawn to 10 (*A1*) from 6 (its initial notch). Then CRD B is withdrawn to notch 4 (*B1*) from 2 (its initial notch). In the final step, No. 52, CRD A is withdrawn to its final notch 44 (*A19*).

3.2. Updating the particle position

Each particle represents a complete control rod withdrawal sequence, and the speed of the particle means a change in the control rod notch index. In other words, using the speed to update the particle position means that each step in the control rod withdrawal sequence is changed. According to the speed of the particle at iteration k (V_i^k), a new particle position can be calculated by Eqs. (1) and (2). In this study, it is assumed that inserting a control rod is not allowed. Therefore, the step at the control rod notch index of *A2* always follows the step at *A1* (as shown in the example in Fig. 3). If this rule is violated in some steps, the control rod notch indexes of these steps are switched. After checking the rule for all control rod withdrawal steps, the actual new speed is calculated by the present and previous position. As an example, as shown in Fig. 4, a control rod at the notch index of *A1* is withdrawn at the first step in iteration k. After updating the particle position and