Multiobjective Loading Pattern Optimization by Simulated Annealing Employing Discontinuous Penalty Function and Screening Technique

Tong Kyu Park, Han Gyu Joo,* and Chang Hyo Kim

Seoul National University, San 56-1 Sillim-dong, Seoul 151-744, Korea

and

Hyun Chul Lee

Korea Atomic Energy Research Institute, 150 Deokjin-dong, Daejeon 305-353, Korea

Abstract — *The problem of multiobjective fuel loading pattern (LP) optimization employing high-fidelity three-dimensional (3-D) models is resolved by introducing the concepts of discontinuous penalty function, dominance, and two-dimensional (2-D)–based screening into the simulated annealing (SA) algorithm. Each constraint and objective imposed on a reload LP design is transformed into a discontinuous penalty function that involves a jump to a quadratic variation at the point of the limiting value of the corresponding core characteristics parameter. It is shown that with this discontinuous form the sensitivity of the penalty coefficients is quite weak compared to the stochastic effect of SA. The feasible LPs found during SA update the set of candidate LPs through a dominance check that is done by examining multiple objectives altogether. The 2-D–based screening technique uses a precalculated database of the 2-D solution errors and is shown to be very effective in saving the SA computation time by avoiding 3-D evaluations for the unfavorable LPs that are frequently encountered in SA. Realistic applications of the proposed method to a pressurized water reactor reload LP optimization with the dual objectives of maximizing the cycle length and minimizing the radial peaking factor demonstrate that the method works quite well in practice.*

I. INTRODUCTION

The search for an optimum fuel assembly (FA) loading pattern (LP) in the reload core design amounts to a multiobjective constrained optimization problem. Given the number of fresh FAs and available burned FAs, the primary objective of the LP design in most pressurized water reactor (PWR) reload problems is to maximize the cycle length while satisfying all safety constraints such as the maximum peak linear heat generation rate and negative moderator temperature coefficient (MTC). The secondary objectives would be to minimize the integrated radial peaking factor as well as the fuel cost to ensure higher thermal margin and better economy. Some of these objectives, however, are mutually exclusive.

Therefore, satisfying all of them simultaneously can hardly be achieved, and compromises of competing objectives need to be made.

For the solution of the LP optimization problem, the simulated annealing (SA) algorithm[1] has been used widely[2–4] because of the powerful feature of SA that prevents the solver from being trapped in local minima by allowing moves to worse states, let alone to better states. In the single-objective SA (SOSA) algorithm, which involves only a mono-objective function, the LP optimization problem has been solved with the constrained objective function method in which the main objective function is augmented by the penalty terms reflecting the extent of violation of the constraints. The multiobjective problem involving multiple objectives has also been solved by SA algorithms.[5–7] Engrand[5] was the first to present a multiobjective SA (MOSA) algorithm,

---

*E-mail: joohan@snu.ac.kr

which makes combined use of what he called the *G* function, which is defined as the sum of all the logarithmic values of individual objective functions, and the concept of dominance.[8,9] He showed that the use of the *G* function for calculating the acceptance probability of trial solutions and the concept of dominance[8] for archiving candidate solutions enable his MOSA to find the surface of Pareto points or the trade-off surface that multiobjective optimization solvers seek. Parks and Suppapitnarm[6] noted that Engrand's MOSA algorithm has two weaknesses such as the possibility of favoring some objectives over others and that of discriminating trial solutions moving toward the trade-off surface. Pointing out that these weaknesses are attributed to the dissimilarity in magnitude of individual objective functions forming the *G* function and a deficiency of the acceptance logic of trial solutions, they suggested an improved MOSA algorithm that uses individual annealing temperatures for each objective instead of the composite objective function like Engrand's *G* function and a modified acceptance logic. In their acceptance logic, an archiving for the trial solution and then the routine SA acceptance test is followed for the failed LPs.

Keller[7] observed that the improved MOSA algorithm[6] has several drawbacks when applied to constrained in-core optimization problems with a large number of constraints. Their formula to calculate the acceptance probability as the product of individual acceptance probabilities is apt to cause arithmetic underflow or overflow errors when a new LP accompanies a large change in one of the objective function values. The individual annealing schedule for each objective may lead to premature cooling with the increasing number of objective functions, which would either cause trapping in local minima when the optimization proceeds with an LP far from the trade-off surface or not allow a search length adequate to characterize a trade-off surface in the presence of constraints. Keller[7] showed that he could get around three drawbacks by introducing several modifications in the formula for acceptance probability calculation, annealing rate parameters for individual objectives, annealing schedule for the primary objectives, adaptive penalty control algorithm, a termination criterion, etc., at the cost of complexity and empiricism.

As noted from these developments, one of the central elements that characterize a MOSA algorithm is how to determine the acceptance probability of a new trial solution along with the annealing schedule. Using the *G* function as if it were a composite objective function is the salient feature of Engrand's MOSA, even though the *G* function is the source of its weakness. The improved MOSA (Refs. 6 and 7) avoids the *G* function because of its deficiency by using individual annealing temperatures for each objective. Formulating the composite objective function adequately, however, one can devise another MOSA algorithm that gets around not only the weaknesses of Engrand's MOSA but also the complexity and empiricism of the improved MOSA.

One of the objectives of this paper is to present a new MOSA algorithm that utilizes a discontinuous penalty function as the composite objective function fit for a given multiobjective optimization problem. The new MOSA is similar to Engrand's because it uses the composite objective function to determine the acceptance probability of trial solutions, yet it does not have the shortcoming of the latter because individual objective functions forming the composite objective function are similar in magnitude. The new MOSA is different from Engrand's because it treats the composite objective function just like the constrained objective function of SOSA optimization problems. As a consequence, the parallel computing SOSA algorithm available in Ref. 4 can be implemented automatically without complexity and empiricism[7] introduced by Keller for the efficacy of the improved MOSA. Specifically, the new MOSA is free from the possibility of committing underflow or overflow errors. It does not need adaptive penalty control measures to avoid the constraint violation of the archive solutions because discontinuous penalty functions are used to construct the composite objective function. It does not require introducing the two-stage annealing schedule for the primary objectives to promote the SA search along the trade-off surface. The termination criterion as well as the cooling rate parameters do not rely on empiricism. These points will be discussed in Sec. II.

In contrast to the merit of a better global minimum search capability, a drawback of SA is the huge computing requirement leading to a long execution time. One of the measures that can be used to reduce the computing time is parallelization, and the other would be to use a lower-order model.[10] In the previous work,[4] we proposed a parallel computing adaptive SA scheme employing a two-dimensional (2-D) core model. However, it is often encountered that the final solution obtained from the 2-D model–based SA calculation is not satisfactory when it is reevaluated by the three-dimensional (3-D) model. In order to avoid this problem, it is better to adopt the 3-D model being used in the design analysis directly during the SA calculation although this will lead to a significant increase in the computing time. However, it is possible to establish an efficient 3-D model–based SA scheme that can avoid excessive 3-D evaluations by utilizing a 2-D model–assisted acceptance or rejection criterion. As the second objective of this paper, we present a 2-D model–based screening technique[11,12] (ST) that allows a considerable portion of the LP evaluation done by a 2-D model rather than by a 3-D model. In order to assess the performance of the new MOSA-based LP optimization method characterized by both the discontinuous penalty function and the 2-D–based ST, the LP optimization problem of a real PWR plant, cycle 4 of Yonggwang Nuclear Unit 4 (YGN4) of Korea,[13] is solved with the dual objectives of maximizing the cycle length and minimizing the radial peaking factor.

II. MULTIOBJECTIVE SIMULATED ANNEALING WITH DISCONTINUOUS PENALTY FUNCTIONS

The LP optimization problem here is to find the best FA LP for a reload cycle PWR core that maximizes the cycle length and minimizes the power peaking factor as well as the fuel cost while meeting all the design constraints. It is thus a constrained optimization problem. Mathematically, the problem can be stated as follows:

$$\text{Find } X \text{ that minimizes } \mathbf{f}(X) \text{ subject to } \mathbf{g}(X) \le 0 \ , \quad (1)$$

where X designates an LP. $\mathbf{f}(X) = [f_1(X), \ldots, f_m(X)]^T$ is an m-dimensional column vector with the superscript T denoting the transpose of a vector. f_i's $(i = 1, 2, \ldots, m)$ are individual objective functions to be minimized individually and are associated with the cycle length, the peaking factors, and other performance parameters of LP X. $\mathbf{g}(X) = [g_1(X), \ldots, g_n(X)]^T$ is an n-dimensional column vector formed by n design constraint functions, $g_j(j = 1, 2, \ldots, n)$. For our problem, the design constraints generally include the limits for the radial pin power peaking factor (F_r), the peak linear heat rate or the 3-D pin power peaking factor (F_q), the pin discharge burnup (PDB), the MTC at hot full power and at hot zero power at beginning of cycle, etc.

In order to solve the minimization problem by SA, it is necessary to define a scalar functional in terms of the multiobjective functions and constraint functions. In the following, several methods for defining the scalar functional are discussed starting from the conventional penalty function method.[14] The goal is to establish a new multiobjective function that can be less sensitive to the penalty coefficients.

II.A. Formulation of Discontinuous Penalty Function

The penalty function method treats design constraints as penalty terms on the optimization functional that is to be minimized. As such it makes use of the scalar functional $F(X)$ defined as a weighted sum of individual objective functions:

$$F(X) = \mathbf{w}^T \mathbf{f}(X) + \boldsymbol{\theta}^T \mathbf{g}(X) \ , \quad (2)$$

where

$\mathbf{w} = [w_1, \ldots, w_m]^T = m$-dimensional column vector consisting of m weighting factors, $\omega_i(i = 1, 2, \ldots, m)$

$\boldsymbol{\theta} = [\theta_1, \ldots, \theta_n]^T = n$-dimensional column vector consisting of n penalty coefficients, $\theta_j(j = 1, 2, \ldots, n)$.

The optimization problem stated by Eq. (1) is then rephrased in this method as

$$\text{Find } X \text{ that minimizes } F(X) \ . \quad (3)$$

The vector $\boldsymbol{\theta}$ or the penalty coefficients are fixed constants in the static penalty function method while they may change adaptively in the dynamic penalty function method during the course of SA optimization calculations. In either case, the determination of the proper $\boldsymbol{\theta}$ is a difficult problem because the relative weight of any one constraint function to another constraint function can hardly be known. Actually, how to set the penalty coefficients for all the involved constraint functions is not a well-defined problem either. Besides, there is no 100% guarantee that the optimum solution from it does not violate the constraints all the time, unless $\boldsymbol{\theta}$ is carefully controlled. Because of these reasons, it is desirable to convert the constrained minimization problem to an unconstrained problem that does not require determining penalty coefficients.

Suppose that the design constraints are given by

$$\alpha_i(X) \le \alpha_i^{\lim} \ , \quad i = 1, 2, \ldots, n \ . \quad (4)$$

Let us define the penalty function, $J_i(X) \ (i = 1, 2, \ldots, n)$, corresponding to each constraint by

$$J_i(X) = \left(a_i + \frac{\delta_i(X)^2}{\overline{\delta_i^2}} \right) u(\delta_i(X)) \ , \quad (5)$$

where

$$\delta_i(X) = \alpha_i(X) - \alpha_i^{\lim} \ ,$$

$$\overline{\delta_i^2} = \frac{1}{N} \sum_{k=1}^{N} \delta_i(X_k)^2 u(\delta_i(X_k)) \ ,$$

and a_i is a user-defined constant that may be taken simply to 1. $u(x)$ is the unit step function defined as

$$u(x) = \begin{cases} 1 \ , & x > 0 \\ 0 \ , & x \le 0 \ . \end{cases} \quad (6)$$

$\delta_i(X)$ is the deviation of the i'th constraint of a given LP X. $\overline{\delta_i^2}$ is the expected value of the squared deviation of the i'th constraint. It is used here to normalize the value of δ_i^2 so that $J_i(X)$ becomes more or less similar in magnitude regardless of the types of constraints. Note that $u(\delta_i(X))$ and $J_i(X)$ are 0 if LP X satisfies the constraint, Eq. (4), and they are nonzero otherwise. Thus, $J_i(X)$ has a discontinuity at the limiting value, $\alpha_i(X) = \alpha_i^{\lim}$, and becomes a quadratic function of $\alpha_i(X)$ beyond this limit.

The penalty function given by Eq. (2) can be extended to the individual objective functions as well by setting $\alpha_{k+n}^{\lim}$ as the required maximum for the k'th objective function. For example, it can be the maximum pin

peaking factor for the reload core. In this way, one can treat the m objectives in exactly the same way as n constraints. Now, let us define a global penalty function as the summation of all the penalty functions representative of n constraints and m objective functions:

$$J(X) = \sum_{i=1}^{m+n} J_i(X) \ . \tag{7}$$

The combination of this global penalty function with the original objective functions can convert the constrained minimization problem into an unconstrained one, which may be stated as follows:

Find X that minimizes $\mathbf{F}(X)$

$$= [J(X), f_1(X), \ldots, f_m(X)]^T \ . \tag{8}$$

Note that the minimum value of $J(X)$ is zero. Therefore, any LP X satisfying $J(X) = 0$ not only meets all the constraints but also guarantees that each individual objective function is less than the required maximum set individually. This means that LP X satisfying $J(X) = 0$ is qualified as a solution of our optimization problem, and therefore, it is called the feasible solution.

Our new MOSA takes advantage of this fact. The global penalty function is viewed as the composite objective function to be minimized and is treated just like the constrained objective function of the SOSA optimization problem in the new MOSA. Consequently, the MOSA solution can be obtained by a routine SOSA algorithm with[4] or without[2,3] parallel computing arrangements. The new MOSA seeks the optimum LP through repeated applications of the three-step SA algorithm:

1. creation of a new LP, X_{new}, from the current LP, X_{cur}

2. comparison of the objective function values of the current and the new LPs, $J(X_{cur})$ and $J(X_{new})$

3. acceptance of X_{new} with the acceptance probability of $\exp(-\Delta J/C)$, where $\Delta J = J(X_{new}) - J(X_{cur})$ and C represents the cooling temperature.

At the end of the SA calculation, there may be one or more feasible LPs satisfying $J(X) = 0$. If this is the case, the best LP can be chosen among the feasible LPs listed in the order of importance of the designer-set main objective. In case that no feasible LP is produced, one may repeat the SA calculation with some adjustment of the SA parameters involved, such as the cooling rate, the number of LPs being sampled at each cooling stage, or the limits set for the main objective functions.

The SOSA algorithm employed for the new MOSA has a serious shortcoming in that it lacks a recipe for recognizing the superiority of one feasible LP over the other feasible ones when determining acceptance of X_{new} because X_{new} is always accepted once it is a feasible LP, namely, when $J(X_{new}) = 0$. As a result, discriminating the inferior feasible LP against the superior feasible LP and intervening in the acceptance or rejection of the new LP is not possible with the J function only. To overcome this shortcoming, we incorporate the concept of dominance,[8] which has been used in the multiobjective genetic algorithm[9] and also in the existing MOSA algorithms.[5–7] The definition is available in the literature. For clarity of our new MOSA algorithm, it is defined here again at the risk of repetition.

Suppose that two LPs X_1 and X_2 are feasible. X_1 is said to dominate X_2 if all the objective functions of X_1 are smaller than those of X_2, namely, if $f_i(X_1) < f_i(X_2)$ for all i. Thus, a dominating LP is the better of the two LPs from the standpoint of the objectives of the LP optimization. Incorporation of the concept of dominance into the SA algorithm enables one to accept a new feasible LP when it dominates the current feasible LP that has produced it and makes it possible to remove those feasible LPs archived as the candidate feasible LP set (simply the candidate LP set hereafter) if they are dominated by the new feasible LP. The candidate LP set here refers to the set of the feasible LPs that are created and updated through the dominance check during the course of MOSA optimization. The logic for updating the candidate LP set is explained below together with the acceptance logic for the new LP.

Figure 1 shows the SA acceptance algorithm modified by incorporating the concept of dominance. As noted in Fig. 1, the conventional SA algorithm is basically applied, but it is subject to modification when a new feasible LP is produced from the current feasible LP. Suppose that a feasible LP X_{new} is generated from X_{cur}, i.e., $J(X_{new}) = 0$. If X_{cur} is not a feasible LP, i.e., if $J(X_{cur}) \neq 0$, X_{new} is automatically accepted, and a dominance check is made with the members of the candidate LP set. If X_{cur} is already a feasible one, however, X_{new} is accepted only when it meets dominance conditions dictated by the modified SA algorithm. First of all, X_{new} is always accepted if it dominates X_{cur}. Otherwise, dominance checks are made to determine (a) whether X_{new} dominates any member of the existing candidate LP set or (b) whether X_{new} and the existing candidate LPs are not dominated mutually. If these are the cases, X_{new} is accepted and registered as a new member of the candidate LP set. Otherwise, it is rejected. It must be noted that any existing candidate LPs dominated by X_{new} are eliminated from the candidate LP set whenever the feasible X_{new} is accepted. Besides the acceptance logic of the feasible X_{new}, the new MOSA algorithm is distinguished from the SOSA algorithm in that the former eliminates those member LPs of the existing candidate LP set that are dominated by the newly accepted feasible LP, while the latter retains all the new feasible LPs as the member of the candidate LP set. The new MOSA algorithm updates the candidate LP set at every acceptance of a new feasible LP. After the search is completed, the resulting candidate

Fig. 1. Acceptance logic of the new MOSA algorithm.
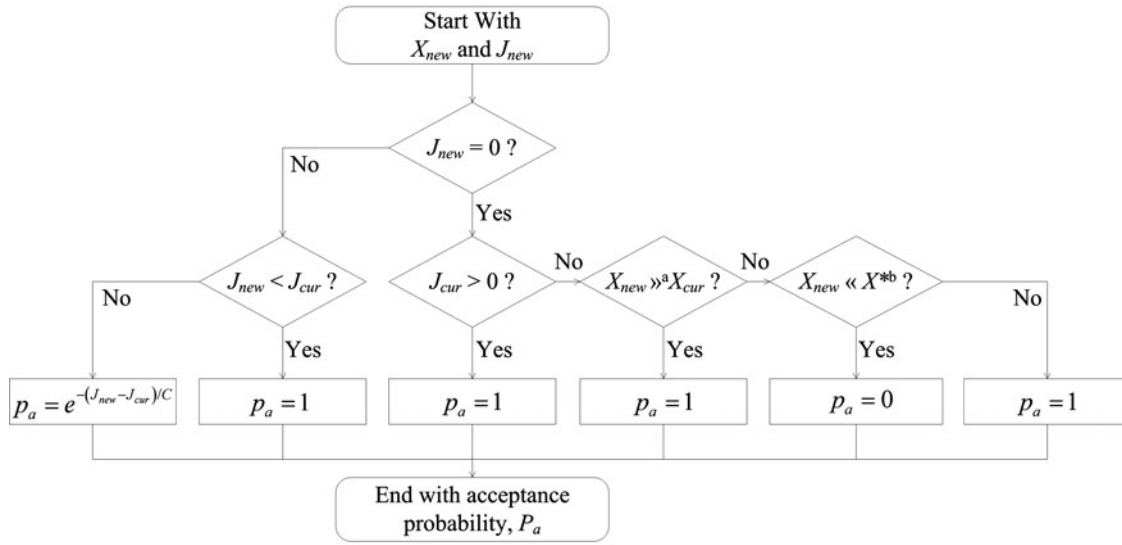
[a] $A \gg B : A$ dominates B

[b] $X^* :$ Any LP in Candidate LP set

LP set will contain the feasible LPs that are mutually nondominating but dominate all the other feasible LPs that have been generated during the SA calculation. The designer can choose one of them as the final solution, namely, the best LP.

In summary, incorporation of the concept of dominance into the SA algorithm affects the LP search calculations in two ways. One is to fill gradually the candidate LP set by better LPs dominating the previous feasible LPs. Another is to affect the LP search calculation itself by rejecting the feasible new LPs that are dominated by any existing candidate LP. The effect is nonexistent in the conventional SA calculation and stands out at the later stage of SA calculation when the cooling temperature is very low and it is highly likely to generate new feasible LPs from the current feasible LP. Needless to mention, when the feasible X_{cur} registered as a member in the candidate LP set produces a nonfeasible X_{new}, this has the chance of being accepted as dictated by the SA acceptance probability and may induce a move toward a feasible LP or the trade-off surface according to the subsequent applications of the MOSA algorithm.

The penalty function defined by a weighted sum of discontinuous penalty functions given in Eq. (7) has an advantage that its minimum value is always zero irrespective of the weighting factors. Obviously, it is a special case of the general penalty function,

$$\tilde{J}(X) = \sum_{i=1}^{n+m} w_i J_i(X) \ , \qquad (9)$$

which has the minimum value of zero, for any set of w_i's Note that Eq. (7) is obtained with uniform weights, e.g.,

$w_i = 1$ for all i. For this reason, the use of the uniformly weighted penalty function, Eq. (7), with the penalty functions modified slightly from those in Eq. (5),

$$J_i(X) = \left(1 + \tilde{c}_i \frac{\delta_i(X)^2}{\overline{\delta_i^2}}\right) u(\delta_i(X)) \ , \qquad (10)$$

can produce the same result as the nonuniformly weighted penalty function given by Eq. (9). In this context, let us note that $\tilde{c}_i$ plays a similar role as the relative importance of the constraint function or the objective function.

To be able to apply the SA algorithm for minimization of the penalty function discussed above, one should specify constants $\tilde{c}_i$ and $\overline{\delta_i^2}$ in Eq. (10). By definition, determination of $\overline{\delta_i^2}$ requires sampling all the possible LPs. In practice, however, it is inevitable to use an average value of δ_i^2 over a finite sampling of LPs, say, $\overline{\delta_{iN}^2}$, which is an average value of δ_i^2 obtained from N LPs sampled initially. There must be the discrepancy between $\overline{\delta_i^2}$ and $\overline{\delta_{iN}^2}$ by $\Delta\overline{\delta_{iN}^2} = \overline{\delta_i^2} - \overline{\delta_{iN}^2}$. In terms of $\overline{\delta_{iN}^2}$ the penalty function given by Eq. (10) can be rewritten as

$$J_i(X) = \left(1 + c_i \frac{\delta_i(X)^2}{\overline{\delta_{iN}^2}}\right) u(\delta_i(X)) \ , \qquad (11)$$

where

$$c_i = \frac{\overline{\delta_{iN}^2}}{\overline{\delta_{iN}^2} + \Delta\overline{\delta_{iN}^2}} \tilde{c}_i \ .$$

Since $\tilde{c}_i$ is a user-specified constant representing the relative importance of the i'th constraint, c_i is also regarded as a free parameter. For the sake of interest, however, it will be shown later that the SA optimization process has a weak dependence on c_i. Thus, it does not really matter what value to use for c_i in the new MOSA optimization formulation based on the discontinuous penalty function.

II.B. Two-Objective Loading Pattern Optimization

The multiobjective optimization formulation presented above is applicable to problems of both multiple objectives with $m > 1$ and a single objective with $m = 1$. To illustrate the simplest example of the multiobjective optimization formulation and also to examine the viability of the MOSA algorithm incorporating the concept of dominance as well as the penalty function method, we consider the reload design problem of the cycle 4 core of YGN4 as the dual-objective LP optimization problem aiming at finding the best LP to have the longest cycle length (L_c) and the lowest possible radial peaking factor (F_r), given the fresh and once- and twice-burned FAs. The choice of the lowest F_r as another objective parameter here is intended to secure the thermal margin as large as possible because it can turn into an economic benefit.

Table I lists the parameters of both the design constraints and the objectives. F_r usually plays as a design constraint, particularly in the single-objective optimization problem, but it is treated here as one of the objective parameters. Representing the cycle length to be maximized in the form like constraints, namely, Eq. (4), in conformity with the penalty function method needs a little precaution. Because the penalty function method is designed to minimize the involved objective parameters, one treats here the negative value of the cycle length as the objective parameter. The constrained optimization condition that the cycle length should meet the required minimum cycle length ($L_{c,\min}$), e.g., $L_c(X) \geq L_{c,\min}$, can be expressed equivalently as

$$-L_c(X) \leq -L_{c,\min} \ . \tag{12}$$

The dual-objective minimization problem is then stated as follows:

Find X that minimizes $\mathbf{F}(X)$

$$= [J(X), -L_C(X), F_r(X)]^T \ . \tag{13}$$

$J(X)$ in the three-element objective function vector $\mathbf{F}(X)$ is given in Eq. (7) with the individual penalty functions $J_i(X)$ defined in Eq. (11).

III. SCREENING TECHNIQUE ASSISTED BY TWO-DIMENSIONAL NEUTRONICS EVALUATION MODEL

As mentioned in Sec. I, the SA optimization has a major drawback of long computing time because it requires evaluation of tens of thousands of trial LPs. To get around this, one is tempted to adopt simpler and faster evaluation models such as 2-D models,[2] neural network–assisted models, etc.[10] In most cases, however, the best LPs from these simplified models turn out to be far from being optimal when they are reevaluated by the 3-D design code. To serve the reload core designers' needs, it is most desirable to adopt the 3-D core model used for the neutronic design. The issue in this case is how to alleviate the computational burden. The ST is a way to resolve this issue by screening out a considerable portion of 3-D evaluation of trial LPs and thus reducing the SA computation time.

The ST takes advantage of a predetermined distribution function on $\delta J(X) = J^{3\text{-D}}(X) - J^{2\text{-D}}(X)$, the difference between the 3-D constrained objective function, $J^{3\text{-D}}(X)$, and its 2-D counterpart, $J^{2\text{-D}}(X)$, for X. To obtain the distribution function for $\delta J(X)$, one needs to sample and evaluate a set of LPs that supposedly represent the whole range of LPs subject to the SA optimization calculation. Suppose that one has obtained the distribution with the mean $\overline{\delta J}$ and the standard deviation σ and also $e_\kappa = \Pr\{|\delta J(X) - \overline{\delta J}| \leq \kappa\sigma\}$, which is the probability of $\delta J(X)$ lying within $\pm\kappa\sigma$ ($\kappa = 1, 2,$

TABLE I

Design Constraints and Objectives for Cycle 4 Core of YGN4

i	Symbol α_i	Design Parameters	Design Limit	Target Limit
1	F_r	Integrated radial pin peaking factor	1.55	1.50
2	F_q	3-D pin peaking factor	2.58	2.20
3	B_D	Pin discharge burnup (GWd/tonne U)	54	54
4	α_Z	Hot zero power MTC (pcm/°C)	5	5
5	α_F	Hot full power MTC (pcm/°C)	0	0
6	L_C	Cycle length (effective full-power day)	383	393

or 3) from $\overline{\delta J}$. If one defines $J_{\max}^{\text{3-D}}(X) = J^{\text{2-D}}(X) + \overline{\delta J} + \kappa\sigma$ and $J_{\min}^{\text{3-D}}(X) = J^{\text{2-D}}(X) + \overline{\delta J} - \kappa\sigma$, it follows that the probability of $J^{\text{3-D}}(X)$ lying between $J_{\min}^{\text{3-D}}(X)$ and $J_{\max}^{\text{3-D}}(X)$ is $100e_\kappa\%$, namely,

$$\Pr(J_{\min}^{\text{3-D}} \le J^{\text{3-D}}(X) \le J_{\max}^{\text{3-D}}) = 100e_\kappa\% \qquad (14)$$

or

$$\Pr(J^{\text{3-D}}(X) < J_{\min}^{\text{3-D}} \text{ or } J^{\text{3-D}}(X) > J_{\max}^{\text{3-D}}) = 100(1 - e_\kappa)\% \ . \qquad (15)$$

The value of e_κ depends on the actual distribution function of $\delta J(X)$ and the κ value. For example, $e_2 = 0.956$ and $e_3 = 0.997$ in the case that $\delta J(X)$ is normally distributed. If one uses e_2 in Eq. (15) in this case, one sees that $\Pr(J^{\text{3-D}}(X) < J_{\min}^{\text{3-D}})$ or $\Pr(J^{\text{3-D}}(X) > J_{\max}^{\text{3-D}})$ is only 2.2% because of the symmetry of the normal distribution. The ST makes the most of this result to devise an approximate acceptance or rejection criterion that does not require 3-D evaluation for a trial LP sampled in the course of SA optimization.

In order to show how the ST works, let us consider the acceptance probability of a new LP X_{new} produced from the current LP, $X_{cur}, \exp(-\Delta J/C)$. In the SA calculation, this probability is simulated by the pseudorandom number ξ, and X_{new} is either accepted when $\exp(-\Delta J/C) \le \xi$ or rejected when $\exp(-\Delta J/C) > \xi$. Note that the acceptance and the rejection criteria can be rewritten as

SA criteria

$$= \begin{cases} X_{new} \text{ is accepted if } J(X_{new}) \le J(X_{cur}) - C\ln\xi \\ X_{new} \text{ is rejected if } J(X_{new}) > J(X_{cur}) - C\ln\xi \ . \end{cases}$$
$$(16)$$

Obviously, 3-D evaluation of the penalty functions of all the LPs to be tested by the SA criteria above is costly. As a way to avoid the necessary 3-D evaluation for $J(X_{new})$, the ST uses $J_{\max}^{\text{3-D}}(X_{new})$ to implement the acceptance criterion while it uses $J_{\min}^{\text{3-D}}(X_{new})$ to implement the rejection criterion. Instead of Eq. (16), the ST accepts X_{new} without 3-D calculation when the following condition is satisfied:

$$J_{\max}^{\text{3-D}}(X_{new}) < J(X_{cur}) - C\ln\xi \ , \qquad (17a)$$

where

$$J_{\max}^{\text{3-D}}(X_{new}) = J^{\text{2-D}}(X_{new}) + \overline{\delta J} + \kappa\sigma \ . \qquad (17b)$$

On the other hand, the ST rejects X_{new} without 3-D calculation when the following condition is met:

$$J_{\min}^{\text{3-D}}(X_{new}) > J(X_{cur}) - C\ln\xi \ , \qquad (18a)$$

where

$$J_{\min}^{\text{3-D}}(X_{new}) = J^{\text{2-D}}(X_{new}) + \overline{\Delta J} - \kappa\sigma \ . \qquad (18b)$$

The acceptance criterion, Eq. (17), and the rejection criterion, Eq. (18), require only the 2-D evaluation of X_{new}. If X_{new} meets Eq. (17), it is accepted without the 3-D evaluation for it. Likewise, if X_{new} satisfies Eq. (18), it is rejected without the 3-D evaluation for it. In other words, the above criteria screen out the 3-D evaluations for some of the trial LPs that may appear in the SA optimization process. Consequently, they can reduce the SA computing time. We observe that the acceptance criterion screens out effectively the 3-D evaluations of the trial LPs at the earlier stages of the SA algorithm when the cooling temperature C is high, while the rejection criterion does at the later stages of the SA algorithm when the cooling temperature C is low. Note that the acceptance probability of X_{new} at the high-temperature earlier stages of SA is close to 1, and thus, a greater portion of LPs sampled are accepted as X_{new}, while the rejection probability of X_{new} at the low-temperature later stages of SA is close to 1, and thus, a greater portion of LPs sampled are rejected. As a matter of course, when X_{new} is accepted by Eq. (17), it replaces X_{cur} as the SA algorithm dictates. Unless the penalty function value of X_{new} is too low, e.g., $< J_{\lim}$, it is approximated by $J^{\text{3-D}}(X_{new}) \approx \tilde{J}(X_{new}) = J^{\text{2-D}}(X_{new}) + \overline{\delta J}$. On the contrary, if $\tilde{J}(X_{new}) < J_{\lim}$, a 3-D calculation is performed to determine $J^{\text{3-D}}(X_{new})$ more accurately. The decision on acceptance of X_{new} in this case is made once more according to the normal acceptance criterion for three dimensions, Eq. (17). The algorithm of screening as well as the dominance check explained so far is given in a pseudocode form in Fig. 2.

The adoption of the ST may have an unspecified effect on the best LP search process by the SA algorithm by probably accepting or rejecting LPs that could not be accepted or rejected if the 3-D model is used to evaluate them. In order to avoid this effect as far as one can, one may set e_κ as high as practicable so that $\Pr(J^{\text{3-D}}(X) < J_{\min}^{\text{3-D}})$ or $\Pr(J^{\text{3-D}}(X) > J_{\max}^{\text{3-D}})$ becomes small enough to justify the use of the modified acceptance and rejection criteria, Eqs. (15) and (16), with high confidence. Considering that the increasingly higher e_κ results in increasingly higher confidence, but increasingly lower efficiency, of the ST, we here adopt $e_2 = 0.954$ as a compromise. $e_2 = 0.954$ can be derived exactly when the normal distribution is assumed for $\delta J(X)$. It is arbitrarily set here regardless of the actual distribution function for $\delta J(X)$. It must be noted that utilization of Eqs. (14), (15), or (16) for the ST does not need the distribution function for $\delta J(X)$ but the numerical value for e_κ.

1. Choose initial SA parameters.
2. Reset stage parameters: $n_s := 0$ and $n_a := 0$.
3. If $n_s > N_{smp}$ or $n_a > N_{acp}$, go to step 4.

    3.1 Generate X_{new} from X_{cur}.

    3.2 Perform a 2D calculation and evaluate $J^{2D}(X_{new})$.

    3.3 Determine J_{acp}, $J_{min}(X_{new})$, and $J_{max}(X_{new})$.

    3.4 If $J_{min}(X_{new}) > J_{acp}$, go to 3.12.

    3.5 If $J_{max}(X_{new}) < J_{acp}$, then

$$\tilde{J}(X_{new}) = J_{2D}(X_{new}) + \overline{\Delta J}$$
$$\tilde{J}(X_{new}) > J_{\lim}, \quad J(X_{new}) = \tilde{J}(X_{new}) \text{ and go to 3.11.}$$

    3.6 Perform a 3D calculation and evaluate $J(X_{new})$.

    3.7 Update $\overline{\Delta J}$ and σ.

    3.8 If $J(X_{new}) \neq 0$, then

        If X_{new} is to be accepted, then go to 3.11, else go to 3.12.

    3.9 If X_{new} dominates anyone in the candidate set, then

        update the candidate set with X_{new} and go to 3.11

    3.10 If $J(X_{cur}) = 0$, go to 3.12.

    3.11 $X_{cur} = X_{new}, J(X_{cur}) = J(X_{new}), n_a := n_a + 1$

    3.12 $n_s := n_s + 1$ and go to 3.1

4. Terminate if $n_a = 0$ or other stopping criterion is satisfied.
5. Reduce the artificial temperature C and go to step 2.

Fig. 2. Pseudocode for SA with dominance check and screening.

IV. APPLICATION TO REALISTIC LOADING PATTERN SEARCH

The new MOSA algorithm with the discontinuous penalty function and the 2-D–based ST described above were implemented in the Unified Nodal Code for Advanced Reactor Design and Simulation[15] (UNCARDS). To examine its effectiveness in a realistic LP search problem, UNCARDS is applied for the reload design of the cycle 4 core of YGN4, which is a former Combustion-Engineering PWR loaded with 177 FAs (Ref. 13). Shown in Fig. 3 is the actual FA LP for the cycle 4 core of YGN4. It is taken here as the reference LP, the perfor-

Fig. 3. Reference fuel LP of cycle 4 core of YGN4.

FA Type	Enrichment	No. of BP	No. of FA
B1	2.36/1.30	8	1
D0	3.35/2.85	0	8
E0	4.11/3.59	0	16
E1	4.10/3.59	8	4
E2	3.60/3.12	8	28
F0	4.11/3.62	0	12
F1	4.10/3.62	8	16
F2	4.10/3.61	12	36
G0	4.07/3.60	0	16
G1	4.07/3.59	8	8
G2	4.08/3.58	12	32

mance of which will be compared with the optimum or the near-optimum LPs generated from the MOSA algorithm presented above. The FAs designated by G0, G1, and G2 are fresh assemblies with 0, 8, 12 burnable poison rods (BPRs), respectively, while the remaining FAs are once- and twice-burned ones. Table I shows the design limits and constraints actually used for the reference LP and the target limits imposed in this study.

IV.A. Input Parameters of New Multiobjective Simulated Annealing Algorithm and Neutronics Evaluation

The new MOSA algorithm requires user-specified input parameters such as the initial acceptance ratio χ, the number of LPs generated and accepted per stage N_{smp} and N_{acp}, the stopping criterion ε_s, and the probabilities of generating new LP generation logics from a current LP; binary exchanges of FAs (P_1), addition of BPRs in fresh FAs (P_2), and rotation (P_3). Typical values for these parameters have been suggested in previous work.[4] The same values of 0.99, 500, and 50 are used here for χ, N_{smp}, and N_{acp}, respectively. ε_s is set to 0.001, while P_1, P_2, and P_3 are set to 0.7, 0.3, and 0.0, respectively. In addition to these parameters, the parameters such as mean squared deviation $\overline{\delta_{iN}^2}$ and the relative penalty importance parameters c_i and $J_{\lim}$ need to be set. Since $\overline{\delta_{iN}^2}$ can be determined only through evaluations of a certain number of LPs, 100 LPs are arbitrarily generated initially to determine $\overline{\delta_{iN}^2}$. The specification of c_i's requires care because they determine the relative contribution of the individual penalty functions to the overall penalty function and thus may affect the result of the MOSA optimization. After an extensive sensitivity analysis to be discussed in Sec. IV.B, they are all set to 1.0. $J_{\lim}$ can affect the screening efficiency, if it is set to an unnecessarily high value. In this study, it is set to 2.

The 3-D evaluation of LPs by UNCARDS is performed with four radial nodes per FA and 26 axial nodes, while the 2-D evaluations are conducted with one node per assembly. The number of burnup steps is 15. The cycle length is measured in the core average cycle burnup in MWd/kg at the end of the cycle, namely, when the critical soluble boron concentration reaches 10 ppm.

IV.B. Sensitivity Analysis

In the discontinuous penalty function method presented in Sec. II, the penalty coefficients c_i's are free parameters that must be specified for the MOSA calculations. In order to determine these coefficients properly, the sensitivity of MOSA optimization results to these values needs to be examined first. Among the six coefficients, c_1 and c_6 are chosen for the sensitivity analysis because of the following reasons. First, they are the

penalty coefficients related directly to the dual objectives in this study and therefore are considered more important than the others. Second, the MTC and PDB constraints can be readily satisfied whereas the peaking and cycle length constraints are more difficult to meet. Third, the 3-D peaking factor and radial peaking factor are interrelated. For the required sensitivity analysis, four different values are assigned to each penalty coefficient, which comes up with 16 different cases in terms of the pair of values for (c_1, c_6) as listed in Table II.

For the application of the ST, the probability parameter e_2 is arbitrarily set to 0.954. This implies that the probability that δJ is within the 2σ ranges around $\overline{\delta J}$ is 95.4% no matter which distribution δJ may actually assume. To validate this, we calculated δJ for about 3000 independent LPs for each case of Table II and examined the fraction of LPs whose δJ value is within $\overline{\delta J} \pm 2\sigma$. As shown in Fig. 4, the fraction of LPs whose δJ is within the 2σ ranges of $\overline{\delta J}$ turns out to be $>$95.4% for all 16 cases. Thus, this ensures that the lower and upper bounds of δJ given by Eq. (14) are met.

TABLE II

A Pair of Parameters (c_1, c_6) for Sensitivity Analysis by Case Number

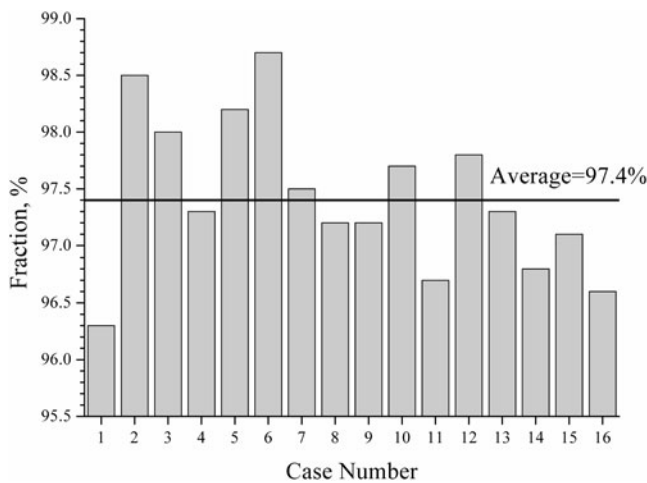| c_1	c_6			
	0.1	1	10	100
0.1	1	2	3	4
1	5	6	7	8
10	9	10	11	12
100	13	14	15	16

Fig. 4. Fraction of LPs whose δJ's are within 2σ ranges around $\overline{\delta J}$.

For each of the 16 cases, the dual-objective LP optimization calculations are performed, and 16 sets of the final candidate LPs are obtained as summarized in Table III, which lists the number of LP's sampled (N_{gen}), the number of LPs subject to 3-D evaluations ($N_{3\text{-}D}$), the number of LPs accepted (N_a), the screening efficiency (η_{ST}), the number of feasible LPs (N_{fea}), the number of candidate LPs (N_{can}), and the turnaround time (T_{hr}). It is observed that all the LPs in the final candidate set satisfy all the design constraints imposed: namely, the values of the penalty function are all zero, and only those LPs that are nondominating mutually remain in the final candidate set.

The screening efficiency turns out to range from 51 to 74%, and the turnaround time varies from 6.8 to 13 h on a Linux cluster consisting of 35 Xeon CPUs (eleven 3.0-GHz Xeon 3.0 CPUs and twenty-four 2.33-GHz Xeon 64 CPUs). A final candidate LP set is always produced for each of the 16 cases although the number of LPs in each candidate set differs depending on the cases. For further examination, the $L_c - F_r$ plots of all the LPs in the 16 candidate sets are shown in Fig. 5. From comparison of the candidate LP sets in terms of the cycle length and the peaking factor of the individual LPs, it seems that cases 7, 11, or 12 give rise to better LPs while case 10 has worse LPs than those in any other candidate sets. The better LPs here refer to those that have relatively longer cycle lengths but relatively lower peaking factors and are located in the lower right region of Fig. 5.

By observing that $c_1 = 1$ and $c_6 = 10$ for the best case (case 7) and $c_1 = 10$ and $c_6 = 1$ for the worst case (case 10), one may conjecture that the penalty coefficients may affect the optimum results of the dual-objective SA calculations. To ascertain whether this is a

TABLE III

Summary of MOSA Optimization for the 16 Cases for the YGN4 Cycle 4

Case	N_{gen}	$N_{3\text{-}D}$	N_a	η_{ST}	N_{fea}	N_{can}	T_h
1	15 881	6763	4203	0.57	565	6	11.4
2	11 222	3959	2980	0.65	496	8	6.8
3	11 099	5451	3493	0.51	341	1	9.1
4	15 240	4465	2968	0.71	402	2	7.7
5	12 233	4058	3247	0.67	548	3	7.1
6	13 854	4697	3081	0.66	293	6	7.8
7	10 112	4506	3320	0.55	216	6	7.8
8	15 783	5228	3245	0.67	372	1	8.7
9	11 376	4580	3281	0.60	288	1	7.9
10	15 881	4193	3060	0.74	448	5	7.0
11	13 524	6078	2998	0.55	353	2	10.3
12	16 289	5936	3422	0.64	282	5	10.1
13	13 162	5044	3114	0.62	399	2	8.8
14	19 605	5277	3423	0.73	915	11	8.2
15	17 476	7878	3370	0.55	320	4	13.0
16	14 703	6333	3402	0.57	269	1	10.8
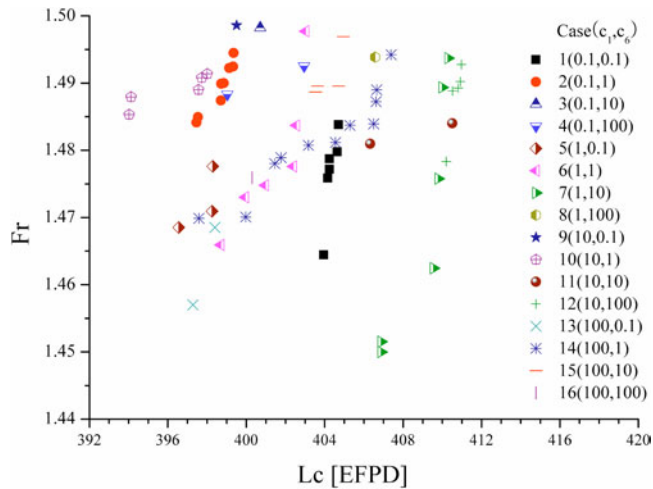
Fig. 5. L_C-F_r plot of LPs in the final candidate sets from 16 different cases in (c_1, c_6).

Fig. 7. Behavior of discontinuous penalty function with different c_i.

pure coincidence or a real trend, the SA optimization calculations for cases 7 and 10 are redone by employing different random number sequences because the stochastic effect resulting from different random numbers may affect the SA outcome. Twelve candidate sets or a total of 55 candidate LPs are obtained from six independent SA runs for each of cases 7 and 10, including the two original sets shown already in Fig. 5. Figure 6 shows the $L_c - F_r$ plots of the 55 LPs in the 12 candidate sets. The $L_c - F_r$ plots show that in contrast to the observation in Fig. 5, case 7 can generate worse candidate LPs while case 10 can produce better candidate LPs comparable to those plotted for this case in Fig. 5, depending on the
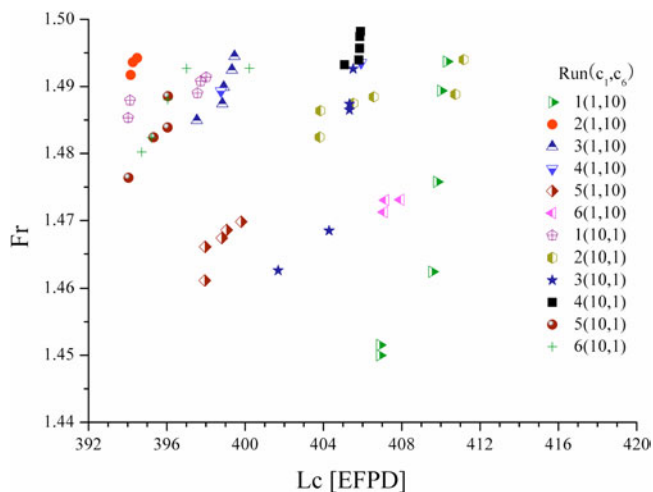

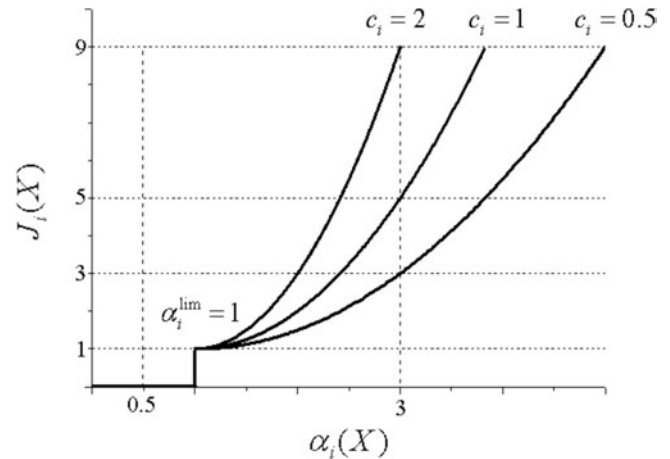Fig. 6. L_C-F_r plot of LPs in 12 final candidate sets from 12 independent runs for cases 7 and 10.

random number sequence. This indicates clearly that the outcome of the dual-objective SA optimization depends less on the values of the penalty coefficients represented here by c_1 and c_6 but rather strongly on the stochastic effect.

The weak dependence on the penalty coefficients can be better explained by using Fig. 7, which illustrates the dependence of the discontinuous penalty function $J(X)$ on one of the penalty coefficients c_i. The three curves in Fig. 7 are obtained with the three different c_i values, but they share the same discontinuity point and the same horizontal line below this point. Consider now two LPs, a feasible LP X, and a nonfeasible LP X'. The discontinuous penalty function value of X is always zero regardless of the value of c_i while that of X' is 3, 5, and 9, respectively, for the three different c_i values. Obviously, the acceptance probability of new LPs above the nonfeasible region depends on c_i values. Therefore, the moves from a nonfeasible LP like X' to another nonfeasible LP are affected by c_i values. The effect depends on where the moves occur and becomes increasingly weaker as the region of moves approaches the feasible region. Suppose now that X is produced as a new LP from X' in the course of the MOSA calculation. This is the move that reduces the discontinuous function value. Though the magnitude of reduction in the discontinuous function values depends on the c_i value, X is always accepted, regardless of the c_i values. Once X is produced, the acceptance probabilities for the moves from X to other feasible LPs around X are not affected because the moves do not change the discontinuous penalty function values no matter what values c_i may have. Only the stochastic effect is dominant in the feasible region and can lead the search to the nonfeasible region. Although the value of c_i can affect the path leading to the feasible region, it is not very significant unless the feasible region is very narrow. Finer tuning in the feasible region is performed

stochastically by dominance check. This is the merit of the new MOSA algorithm based on the discontinuous penalty function and dominance check.

IV.C. Optimization Calculation for Cycle 4 Core of YGN4

With all the necessary input parameters for the MOSA algorithm as specified above, ten independent MOSA optimization runs are conducted. Each MOSA run is done with different random number sequences to examine the stochastic effect. The results are summarized in Table IV. From the results of ten MOSA runs, it is estimated that the average CPU time is ~8.1 h. The average screening efficiency is ~66% with approximately 95 770 LPs out of approximately 144 000 LPs generated in ten MOSA runs screened out.

On the ten-run average, a total of 372 LPs out of 14 400 LPs are found to be feasible solutions; among them 5 remain as candidate LPs. Actually, the ten optimization runs produced a total of 54 candidate LPs, which are displayed in $L_c - F_r$ plots in Fig. 8. From an additional dominance check over them, five final candidate LPs are finally obtained. Shown in Table V are the core performance parameters of the 15 candidate LPs selected by the order of cycle length. It should be noted that all the final five LPs are included in Table V and that they are better than the reference one in that the cycle length is longer by >20 days while the peaking is lower by >2.5%. Surprisingly though, it turns out that all of the five final candidate LPs are produced from the third SA run. Figure 9 displays the number of the feasible LPs and that of candidate LPs in the candidate LP set as a function of the stage number of the third optimization run. Figure 10 illustrates the evolution of the candidate LP set with the progress of the stages. The first candidate
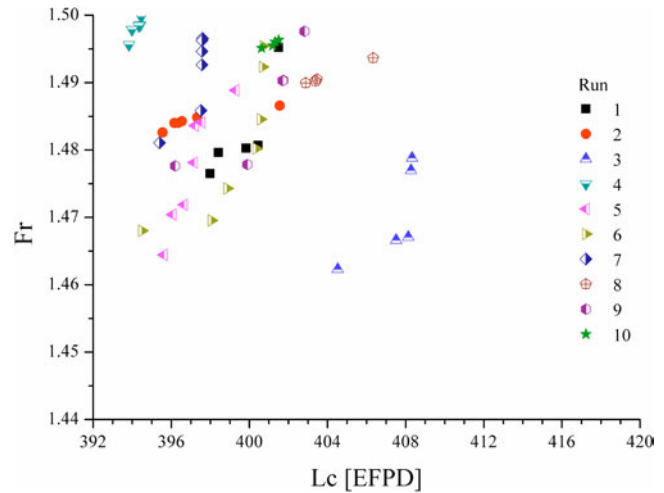

Fig. 8. L_C-F_r plot of 54 candidate LPs in the ten optimization runs.

LP appears at the 31st stage and is removed by the second candidate LP, which is generated at the 32nd stage because it is dominated by the second LP. The second candidate LP is then removed at the 33rd stage by three mutually nondominating candidate LPs, all of which dominate the second one. The candidate LP set continues to be updated with the progress of stages until updating it ends up with the final five LPs at the last stage of number 39. Judging from the standpoint of the intended dual objectives of the problem, the final five LPs show clearly a certain degree of the trade-off between the two objectives. This means that our new MOSA has reached the trade-off surface of the optimum solution to the LP problem. Among the final five LPs shown in Fig. 10, the one marked with a circle seems to be the best in terms of the cycle length and the radial peaking factor. This LP corresponds to the third one listed in Table V and is shown in Fig. 11.

Table V shows that the total number of fresh FAs loaded in the five final candidate LPs and the reference LP have the same number of fresh LPs but that the splits of the three types of fresh FAs are different. This is attributed to the fact that in the previous ten MOSA optimization runs, the number of BPRs in the fresh FA was treated as a variable to be determined. In order to compare the reference LP and the MOSA-derived optimum LP on the equal condition, five additional MOSA optimization runs are conducted using the same FAs used for the reference LP of the cycle 4 core of YGN4. The five runs have produced 37 candidate LPs from which 13 LPs are finally obtained through the dominance check. Figure 12 presents a comparison of these 13 candidate LPs and the previous 5 LPs in terms of the cycle length and the radial peaking factor. All 13 candidates exceed the reference LP in terms of the two core performance parameters, which clearly demonstrates the capability of

TABLE IV

Summary of Ten Optimization Runs for Cycle 4 Core of YGN4

Run	N_{gen}	$N_{3\text{-}D}$	N_a	η_{ST}	N_{fea}	N_{can}	T_h
1	17 236	5 959	3 654	0.65	424	5	9.5
2	11 398	4 089	3 155	0.64	502	6	6.9
3	10 888	3 427	2 649	0.69	445	5	5.8
4	17 694	4 668	2 918	0.74	411	5	7.7
5	13 982	4 882	3 352	0.65	400	7	8.0
6	12 600	4 136	3 209	0.67	348	7	7.3
7	12 401	5 231	3 183	0.58	315	6	8.8
8	14 393	5 210	3 470	0.64	304	4	8.6
9	17 155	4 949	3 374	0.71	366	5	8.5
10	16 249	5 676	3 696	0.65	208	4	9.8
Total	143 996	48 227	32 660	6.62	3 723	54	80.9
Average (σ)	14 400 (2553)	4 823 (770)	3 266 (320)	0.66 (0.04)	372 (83)	5 (1.1)	8.1 (1.2)

TABLE V

Core Characteristic Parameters of the 15 Candidate LPs

| Candidate Index	F_r	F_q	B_D	α_Z	α_F	L_C	Number of Fresh FAs		
							G0	G1	G2
1	1.479	1.775	47.2	3.64	-10.72	408.3	20	16	20
2	1.477	1.770	47.4	3.71	-10.62	408.3	20	16	20
3	1.467	1.756	47.5	3.84	-10.53	408.1	20	16	20
4	1.467	1.753	47.4	3.69	-10.74	407.5	20	12	24
5	1.494	1.781	48.3	4.20	-10.21	406.3	24	16	16
6	1.462	1.754	47.1	3.27	-11.23	404.5	20	12	24
7	1.490	1.808	48.1	3.77	-10.55	403.5	24	20	12
8	1.490	1.799	48.1	3.76	-10.68	403.4	24	16	16
9	1.489	1.796	48.2	3.87	-10.46	402.9	24	20	12
10	1.497	1.786	48.3	3.81	-10.53	402.8	20	16	20
11	1.490	1.779	48.2	3.69	-10.67	401.7	20	16	20
12	1.490	1.780	48.3	3.63	-10.57	401.7	20	16	20
13	1.486	1.802	48.7	4.16	-9.81	401.6	36	8	12
14	1.495	1.793	46.1	3.73	-10.7	401.5	16	28	12
15	1.496	1.782	48.8	2.84	-11.67	401.5	12	24	20
Reference	1.517	1.801	47.2	2.02	-12.82	386.0	16	8	32
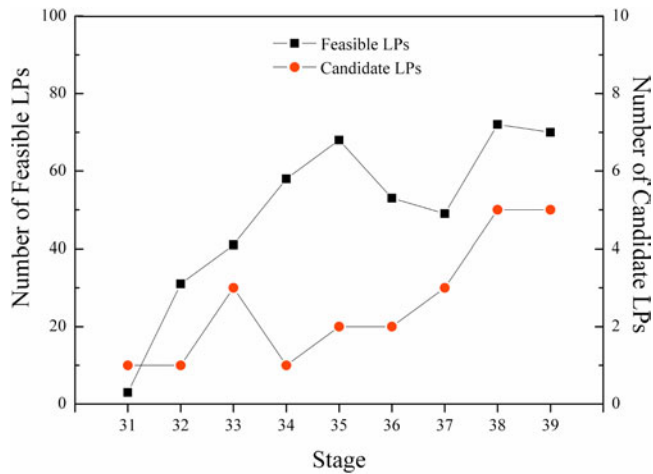
Fig. 9. Number of feasible and candidate LPs versus stage number for the third optimization run.
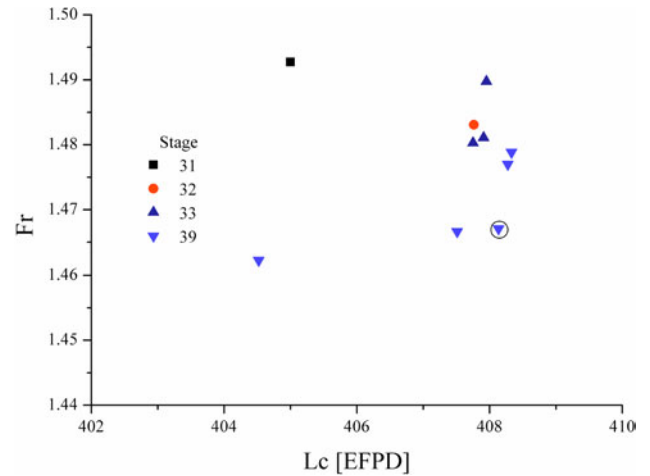

Fig. 10. Evolution of candidate LP set in the optimization run 3.

the MOSA optimization scheme proposed here. On the other hand, they are inferior to the previous five LPs. This is simply the consequence of the increased degree of freedom in the former case to allow different burnable poison loading.

V. CONCLUSION

In this paper we have presented a 2-D–based ST as well as a new MOSA algorithm that is based on the

discontinuous penalty function. In principle, the new MOSA algorithm and the ST can be applicable to both boiling water reactors and PWRs. But, their effectiveness has been examined only for a dual-objective reload design problem of the cycle 4 core of the YGN4 PWR plant. The discontinuous penalty function $J(X)$ formulated here for the new MOSA algorithm is very different from Engrand's G function. The former is constructed as the sum of discontinuous penalty functions $J_i(X)$ of similar magnitude that represent either the individual objectives or the constraints, while the latter as the sum of
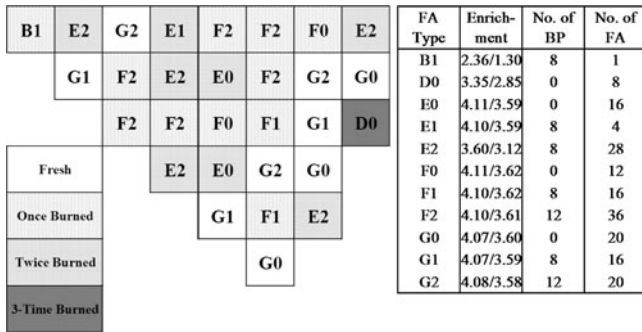
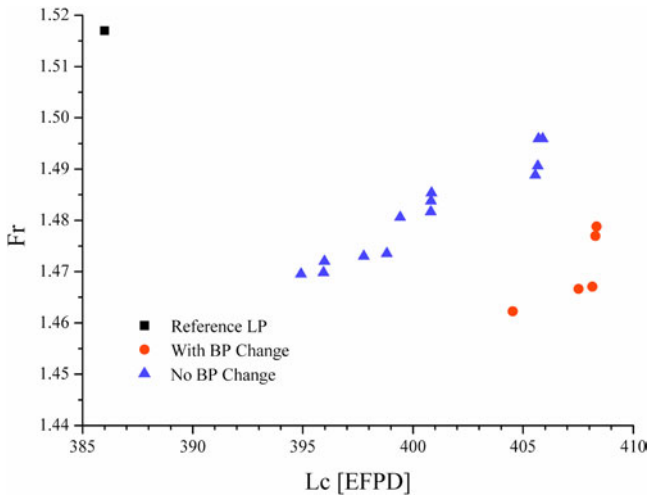| FA Type	Enrichment	No. of BP	No. of FA
B1	2.36/1.30	8	1
D0	3.35/2.85	0	8
E0	4.11/3.59	0	16
E1	4.10/3.59	8	4
E2	3.60/3.12	8	28
F0	4.11/3.62	0	12
F1	4.10/3.62	8	16
F2	4.10/3.61	12	36
G0	4.07/3.60	0	20
G1	4.07/3.59	8	16
G2	4.08/3.58	12	20

Fig. 11. Final candidate LP of YGN4 cycle 4.

Fig. 12. L_C-F_r plot of candidate LPs obtained with and without burnable poison change option.

logarithms of the individual objective functions of dissimilar magnitude. The role of two functions appears to be similar because they are used to determine the acceptance probability of trial solutions produced in the course of MOSA optimization calculations.

Unlike the G function, however, $J(X)$ is the composite objective function to be minimized in accordance with the given multiple-objective minimization problem. These differences between the two functions make the new MOSA algorithm distinct from Engrand's. In particular, the construction of $J(X)$ by the individual penalty functions $J_i(X)$'s of similar magnitude and discontinuity gives advantageous features to the new MOSA algorithm over the existing MOSA algorithms. It makes the new MOSA algorithm avoid the shortcoming of Engrand's, which is ascribed to the dissimilar magnitude of the individual objective functions forming his G function and thus results in much less sensitive dependence on the penalty coefficients and objective weighting coefficients. It also makes the new MOSA algorithm much simpler than the improved MOSA algorithm,[6,7] which uses individual annealing temperatures for each objective in order to get around the shortcoming of Engrand's method. The new MOSA algorithm does not have the drawbacks of Parks and Suppapitnarm such as the risk of committing the underflow or overflow errors in acceptance probability computation, faster cooling of one objective over the other objectives or premature cooling, constraint violation of accepted solutions, etc. It must be noted that to overcome these drawbacks, Keller had to correct the formula for acceptance probability, adjust annealing rate parameters for individual objectives, adopt a two-stage annealing schedule for the primary objectives and adaptive penalty control algorithm, use termination criterion, etc., at the cost of complexity and empiricism.

The 2-D–based ST, which is designed to avoid the expensive 3-D calculations for unfavorable LPs that are frequently encountered during the course of SA, is very effective, as demonstrated by the solution of the new MOSA algorithm to the dual-objective reload core design problem. The average screening efficiency turns out to be ~66%, which amounts to a speedup factor of ~3. As a result, the new MOSA algorithm can produce a candidate LP set with the average turnaround time of ~8 h on a LINUX cluster loaded with 35 Xeon CPUs .

The resulting LPs obtained from repeated MOSA optimization runs for the cycle 4 core of YGN4 turn out to be better than the reference LP regardless of whether the same constituents of fresh FAs are used or not. The cycle length of the newly found LP is longer by >20 days than the reference LP while the peaking is lower by >2.5%. From these observations, it is safely concluded that the new MOSA optimization method proposed here is an effective method and works well in practice to produce a set of candidate LPs meeting all the imposed constraints while satisfying the mutually exclusive objectives with affordable computing resources and time.

REFERENCES

1. A. KIRKPATRICK, C. D. GELATT, Jr., and M. P. VECHI, "Optimization by Simulated Annealing," *Science*, **220**, 671 (1983).

2. D. J. KROPACZEK and P. J. TURINSKY, "In-Core Nuclear Fuel Management Optimization for Pressurized Water Reactor Using Simulated Annealing," *Nucl. Technol.*, **95**, 9 (1991).

3. J. G. STEVENS, K. S. SMITH, K. R. REMPE, and T. J. DOWNAR, "Optimization of Pressurized Water Reactor Shuffling by Simulated Annealing with Heuristics," *Nucl. Sci. Eng.*, **121**, 67 (1995).

4. H. C. LEE, H. J. SHIM, and C. H. KIM, "Parallel Computing Adaptive Simulated Annealing Scheme for Fuel Assembly Loading Pattern Optimization In PWRs," *Nucl. Technol*, **135**, 39 (2001).

5. P. ENGRAND, "A Multi-Objective Optimization Approach Based on Simulated Annealing and Its Application to Nuclear Fuel Management," *Proc. 5th Int. Conf. Nuclear Engineering (ICONE5)*, Nice, France, May 26–30, 1997 (1997).

6. G. T. PARKS and A. SUPPAPITNARM, "Multiobjective Optimization of PWR Reload Core Designs Using Simulated Annealing," *Proc. Int. Conf. Mathematics and Computation, Reactor Physics, and Environmental Analysis in Nuclear Applications (M&C '99)*, Madrid, Spain, September 27–30, 1999.

7. P. M. KELLER "FORMOSA-P Constrained Multiobjective Simulated Annealing Methodology," *Proc. Int. Mtg. Mathematical Methods for Nuclear Applications (MC2001)*, Salt Lake City, Utah, September 9–13, 2001.

8. K. I. SMITH, R. M. EVERSON, J. E. FIELDSEND, C. MURPHY, and R. MISRA, "Dominance-Based Multi-Objective Simulated Annealing," *IEEE Trans. Evolutionary Comput.*, **12**, 323 (2008).

9. G. T. PARKS, "Multiobjective Pressurized Water Reactor Reload by Nondominated Genetic Algorithm Search," *Nucl. Sci. Eng.*, **124**, 178 (1996)

10. C. S. JANG, H. J. SHIM, and C. H. KIM, "Optimization Layer by Layer Networks for In-Core Fuel Management Optimization Computations in PWRs," *Ann. Nucl. Energy*, **28**, 1115 (2001).

11. T. K. PARK, H. C. LEE, H. K. JOO, and C. H. KIM, "Screening Technique for Loading Pattern Optimization by Simulated Annealing," *Trans. Am. Nucl. Soc.*, **93**, 578 (2005).

12. T. K. PARK, H. C. LEE, H. K. JOO, and C. H. KIM, "Improvement of Screening Efficiency in Loading Pattern Optimization by Simulated Annealing," *Trans. Am. Nucl. Soc.*, **96**, 578 (2007).

13. "The Nuclear Design Report for Yonggwang Nuclear Power Plant Unit 4 Cycle 4," KNF-Y4C4-98030, Korea Electric Power Corporation and Korea Nuclear Fuel Corporation (1998).

14. P. KELLER and P. J. TURINSKY, "FORMOSA-P Multiple Adaptive Penalty Function Methodology," *Trans. Am. Nucl. Soc.*, **75**, 341 (1996).

15. H. C. LEE, K. Y. CHUNG, and C. H. KIM, "Unified Nodal Method for Solution to the Space-Time Kinetics Problems," *Nucl. Sci. Eng.*, **147**, 1 (2004).