DESIGN OF A PWR POWER CONTROLLER USING MODEL PREDICTIVE CONTROL OPTIMIZED BY A GENETIC ALGORITHM

MAN GYUN NA* and IN JOON HWANG
Department of Nuclear Engineering, Chosun University
375 Seosuk-dong, Dong-gu, Gwangju 501-759, Republic of Korea
*Corresponding author. E-mail : magyna@chosun.ac.kr

Received June 22, 2005
Accepted for Publication August 17, 2005

In this study, the core dynamics of a PWR reactor is identified online by a recursive least-squares method. Based on the identified reactor model consisting of the control rod position and the core average coolant temperature, the future average coolant temperature is predicted. A model predictive control method is applied to designing an automatic controller for the thermal power control of PWR reactors. The basic concept of the model predictive control is to solve an optimization problem for a finite future at current time and to implement as the current control input only the first optimal control input among the solutions of the finite time steps. At the next time step, this procedure for solving the optimization problem is repeated. The objectives of the proposed model predictive controller are to minimize both the difference between the predicted core coolant temperature and the desired temperature, as well as minimizing the variation of the control rod positions. In addition, the objectives are subject to the maximum and minimum control rod positions as well as the maximum control rod speed. Therefore, a genetic algorithm that is appropriate for the accomplishment of multiple objectives is utilized in order to optimize the model predictive controller. A three-dimensional nuclear reactor analysis code, MASTER that was developed by the Korea Atomic Energy Research Institute (KAERI), is used to verify the proposed controller for a nuclear reactor. From the results of a numerical simulation that was carried out in order to verify the performance of the proposed controller with a 5%/min ramp increase or decrease of a desired load and a 10% step increase or decrease (which were design requirements), it was found that the nuclear power level controlled by the proposed controller could track the desired power level very well.

KEYWORDS : Genetic Algorithm, Model Predictive Control, Nuclear Reactor Power Control, Recursive Parameter Estimation

1. INTRODUCTION

The power and temperature of a nuclear reactor should be properly controlled in order to maintain the performance of the reactor's operating condition as well as to maximize the thermal efficiency of an entire nuclear power plant. However, power plants are highly complex, nonlinear, time-varying, and constrained systems. To illustrate, a plant's characteristics vary with the operating power levels; as well, ageing effects in plant performance, and changes in the nuclear core reactivity with fuel burnup gradually degrade system performance. Furthermore, if a load-following operation is desired, the daily load cycles can change plant performance significantly. Advanced power tracking control of nuclear reactors has not been accepted mainly due to the safety concerns stemming from imprecise knowledge about the time-varying parameters, nonlinearity, and modeling uncertainty. However, rapid and smooth power maneuvering has its benefits in view of the economical and safe operation of reactors and the importance of a load-following strategy.

The model predictive control methodology has received much attention as a powerful tool for the control of industrial process systems [1-7]. The basic concept of the model predictive control is to solve an optimization problem for a finite future at current time. Once a future input trajectory has been chosen, only the first element of that trajectory is applied as the input to the plant. In other words, at a given time, the behavior of the process over a prediction horizon is considered, and the process output related to changes in a manipulated variable is predicted by using a mathematical design model. The changes of the manipulated variables are selected such that the predicted output has certain desirable characteristics. However, only the first computed change in the manipulated variable is implemented, and at each subsequent instant, the procedure is repeated. This method has many advantages over a conventional infinite horizon control, as it is possible to handle input-and state (or output) constraints in a systematic manner during the design and implementation of the control. In particular, it is a suitable control strategy for nonlinear time varying systems because of the model predictive concept. Recently, problems with the control of uncertain dynamical systems have been of considerable interest to control engineers. The model predictive control method has been applied to a nuclear engineering field by Na [8] for the first time.

The model predictive control (MPC) method has been applied to a nuclear power control [9] but the referenced study does not address some of the constraints (for example, maximum and/or minimum inputs) systematically. This paper thoroughly addresses the multiple constraints related to control input using a genetic algorithm.

In this study, a model predictive control method is applied to design an automatic controller for a thermal power control for PWR reactors. The desired average coolant temperature is usually programmed according to the desired reactor power. Although the objective of this controller is to control the average coolant temperature, it is usually called a power controller. The PWR reactor core dynamics are identified online by a recursive least-squares method. Based on the identified reactor model consisting of the control rod position and the core average coolant temperature, the future average coolant temperature is predicted. The objective function for the model predictive control is minimized by a genetic algorithm. A three-dimensional nuclear reactor analysis code that was developed by the Korea Atomic Energy Research Institute (KAERI) is used to verify the proposed controller for a nuclear reactor.

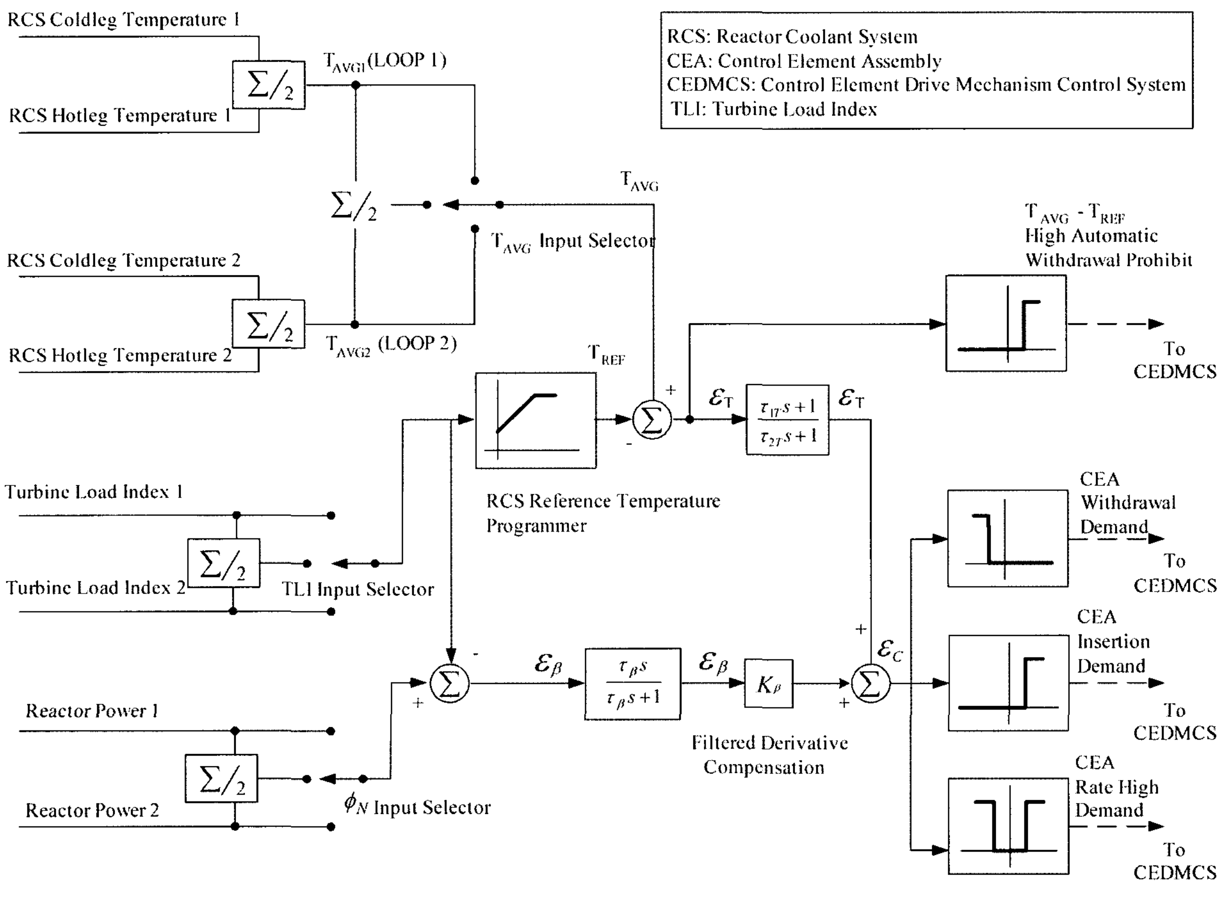


Fig. 1. Conventional Reactor Regulating System

2. CONVENTIONAL POWER CONTROL SYSTEM FOR PWRS

Existing pressurized water reactors (PWRs) have two kinds of control mechanisms: control rods and chemical shim. In many of these reactors, the control rod drives are interconnected electrically so that several control rods move simultaneously in response to a signal from the reactor operator. For example, for Korea Standard Nuclear Power Plants (KSNPs), a total of 73 control rods are divided into two shutdown control rod banks, five regulating control rod banks, and two partial-strength control rod banks. These are groups that move independently. KSNPs can be controlled in part by varying the concentration of boric acid (H_3BO_3) in the coolant because of the strong absorption of neutrons by the boron. The control rods provide a reactivity control for a fast shutdown and for compensating for reactivity changes due to the temperature changes that accompany changes in power. The chemical shim is used to keep the reactor critical during xenon transients and to compensate for the depletion of fuel and the buildup of fission product over the life of the reactor core.

The conventional reactor regulating system controls the average temperature of the reactor core according to the reference temperature that is proportional to the turbine load in order to maximize the plant thermal efficiency. The conventional reactor regulating system is described in Fig. 1. The conventional control method generates a control signal using a temperature deviation channel (the difference between the reference average coolant temperature and the average coolant temperature) and a power mismatch channel (the difference between the turbine load and the nuclear power). In other words, the conventional controller generates the insertion or withdrawal speed of the reactor control rods using the error signals obtained by compensating and filtering these two channels. Finally, the control rod drive mechanism control system moves the control rod assembly groups according to the received signals. This conventional method has the advantages of an easy implementation and a well-proven technology. However, with the aim of optimizing the reactor power control performance, techniques for the optimal power control of nuclear reactors have been studied extensively over the past two decades [10-13]. However, it had proven to be very difficult to design optimized controllers for nuclear systems due to the variations in nuclear system parameters and modeling uncertainties.

3. MODEL PREDICTIVE CONTROLLER

3.1 Model Predictive Control Concept

The model predictive control method solves an optimization problem for a finite future at current time and to implement the first optimal control input as the current control input. This procedure is then repeated at each subsequent instant. Fig. 2 illustrates this basic concept [3]. Looked at another way, for any assumed set of present and future control moves, the future behavior of the process outputs can be predicted over a prediction horizon N, and the M present and future control moves ($M \leq N$) are computed to minimize a quadratic objective function. Although M control moves are calculated, only the first control move is implemented. At the next time step, new values of the measured output are obtained, the control horizon is shifted forward by one step, and the same calculations are repeated. The purpose of taking new measurements at each time step is to compensate for unmeasured disturbances and model inaccuracy, both of which cause the measured system output to be different from the one predicted by the model. At every time instant, model predictive control requires the on-line solution of an optimization problem to compute optimal control inputs over a fixed number of future time instants, known as the time horizon. The basic idea of model predictive control is to calculate a sequence of future control signals in such a way that it minimizes a multistage cost function defined over a prediction horizon.

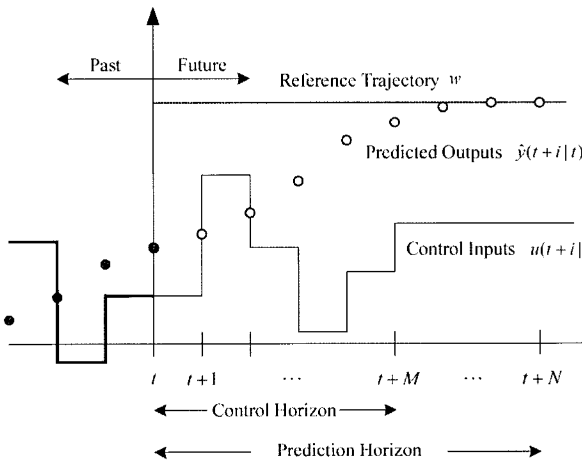


Fig. 2. Basic Concept of a Model Predictive Control Method

Also, in order to achieve fast responses and prevent excessive control efforts, the associated performance index for deriving an optimal control input is represented by the following quadratic function:

$$J = \frac{1}{2}\sum_{j=1}^{N} Q[\hat{y}(t+j|t) - w(t+j)]^2 + \frac{1}{2}\sum_{j=1}^{M} R[\Delta u(t+j-1)]^2 , \quad (1)$$

$$\text{subject to constraints} \begin{cases} \Delta u(t+j-1)=0 \text{ for } j>M, \\ u_{\min} \le u(t) \le u_{\max}, \\ -du_{\max} \le \Delta u(t) \le du_{\max}. \end{cases}$$

where Q and R weight the reactor coolant temperature (system output) error and the control rod position (control input) change between time step at certain future time intervals, respectively, and w is a setpoint (desired average coolant temperature) or reference sequence for the output signal. $\hat{y}(t+j|t)$ is an optimum j-step-ahead prediction of the system output (average coolant temperature) based on data up to time t; that is, the expected value of the output at time t if the past input and output and the future control sequence are known. N and M are termed the prediction horizon and the control horizon, respectively. The prediction horizon represents the limit of the instant in which it is desired for the output to follow the reference sequence. In order to obtain control inputs, the predicted outputs have first to be calculated as a function of the past values of inputs and outputs and of future control signals. The constraint, $\Delta u(t+j-1)=0$ for $j>M$, means that there is no variation in the control signals after a certain interval $M<N$, which is the control horizon concept.

3.2 Output Prediction

The process to be controlled is described by the following Controlled Auto-Regressive and Integrated Moving Average (CARIMA) model, which is widely used as a mathematical model of controller design methods:

$$A(q^{-1})y(t) = B(q^{-1})u(t-1) + \frac{1}{\Delta}C(q^{-1})\xi(t), \quad (2)$$

where y is an output (average coolant temperature), u is a control input (control rod position), ξ is a stochastic random noise sequence with a zero mean value, q^{-1} is the backward shift operator, e.g., $q^{-1}y(t)=y(t-1)$, and Δ is defined as $\Delta=1-q^{-1}$. In Eq. (2), $A(q^{-1})$ and $C(q^{-1})$ are monic polynomials as a function of the backward shift operator q^{-1}, and $B(q^{-1})$ is a polynomial. For example, the polynomial $B(q^{-1})$ is expressed as follows:

$$B(q^{-1}) = b_0 + b_1 q^{-1} + b_2 q^{-2} + \ldots + b_{nB} q^{-nB}, \quad (3)$$

where $b_0, b_1, \cdots, b_{nB}$ are coefficients and nB is the order of the polynomial.

The process output at time $t+j$ can be predicted from the measurements of the output and input up to the time step t. The optimal prediction is derived by solving a Diophantine equation, whose solution can be found by an efficient recursive algorithm. In this derivation, the most common case of $C(q^{-1})=1$ will be considered. The j-step-ahead output prediction of a process is derived below.

Multiplying Eq. (2) by $\Delta E_j(q^{-1})$ from the left gives

$$y(t+j)-E_j(q^{-1})\xi(t+j) = F_j(q^{-1})y(t)+E_j(q^{-1})B(q^{-1})\Delta u(t+j-1), \quad (4)$$

where $E_j(q^{-1})$ and $F_j(q^{-1})$ are polynomials satisfying

$$1 = E_j(q^{-1})\tilde{A}(q^{-1}) + q^{-j}F_j(q^{-1}), \quad (5)$$

$$E_j(q^{-1}) = e_{j,0} + e_{j,1}q^{-1} + \cdots + e_{j,j-1}q^{-(j-1)}, \quad (6)$$

$$F_j(q^{-1}) = f_{j,0} + f_{j,1}q^{-1} + f_{j,2}q^{-2} + \cdots + f_{j,nA}q^{-nA}, \quad (7)$$

$$\tilde{A}(q^{-1}) = A(q^{-1})\Delta. \quad (8)$$

Eq. (5) is termed the Diophantine equation and there exist unique polynomials $E_j(q^{-1})$ and $F_j(q^{-1})$ of order $j-1$ and nA, respectively, such that $e_{j,0}=1$. By taking the expectation operator and considering that $E\{\xi(t)\}=0$, the optimal j-step-ahead prediction of $\hat{y}(t+j|t)$ satisfies

$$\hat{y}(t+j|t) = F_j(q^{-1})y(t) + G_j(q^{-1})\Delta u(t+j-1), \quad (9)$$

where

$$G_j(q^{-1}) = E_j(q^{-1})B(q^{-1}),$$
$$\hat{y}(t+j|t) = E\{y(t+j)|t\}.$$

$\hat{y}(t+j|t)$ denotes the estimated value of the output at time step $t+j$ based on all of the data up to time step t. The output prediction can be easily extended to the nonzero mean noise case by adding the term $E_j(q^{-1})E\{\xi(t)\}$ to the output prediction $\hat{y}(t+j|t)$.

By dividing the polynomial, $G_j(q^{-1})$, as in the following equation:

$$G_j(q^{-1}) = \bar{G}_j(q^{-1}) + q^{-j}\tilde{G}_j(q^{-1}) \quad \text{with } \delta(\bar{G}_j(q^{-1})) < j,$$

the prediction equation, Eq. (9), can now be written as

$$\hat{y}(t+j|t) = \bar{G}_j(q^{-1})\Delta u(t+j-1) + \tilde{G}_j(q^{-1})\Delta u(t-1) + F_j(q^{-1})y(t), \quad (10)$$

where $\delta(\cdot)$ denotes the order of a polynomial. The last two terms on the right side of Eq. (10) consist of past values of the process input and output variables, and correspond to the response of the process if the control input signals are kept constant. On the other hand, the first term on the right side consists of future values of the control input signal and corresponds to the response obtained when the initial conditions are zero $y(t-j)=0$, $\Delta u(t-j-1)=0$ for $j>0$

[14]. Eq. (10) can be rewritten as

$$\hat{y}(t+j\,|\,t)=\overline{G}_j(q^{-1})\Delta u(t+j-1)+f_j\,, \tag{11}$$

where

$$f_j=\widetilde{G}_j(q^{-1})\Delta u(t-1)+F_j(q^{-1})y(t)\,. \tag{12}$$

Following this, a set of N j-step-ahead output predictions can be expressed as

$$\hat{\mathbf{y}}=\overline{\mathbf{G}}\Delta\mathbf{u}+\mathbf{f}\,, \tag{13}$$

where

$$\hat{\mathbf{y}}=\left[\hat{y}(t+1|t)\;\;\hat{y}(t+2|t)\;\cdots\;\hat{y}(t+j|t)\;\cdots\;\hat{y}(t+N|t)\right]^T,$$
$$\Delta\mathbf{u}=\left[\Delta u(t)\;\;\Delta u(t+1)\;\cdots\;\Delta u(t+j)\;\cdots\;\Delta u(t+N-1)\right]^T,$$
$$\mathbf{f}=\left[f_1\;\;f_2\;\cdots\;f_j\;\cdots\;f_N\right]^T,$$
$$\overline{\mathbf{G}}=\begin{bmatrix} g_0 & 0 & \cdots & 0 & \cdots & 0\\ g_1 & g_0 & \cdots & 0 & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots\\ g_{j-1} & g_{j-2} & \cdots & g_0 & \cdots & 0\\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots\\ g_{N-1} & g_{N-2} & \cdots & \cdots & \cdots & g_0\end{bmatrix},$$
$$\overline{G}_j(q^{-1})=\sum_{i=0}^{j-1}g_i q^{-i}\,.$$

If all initial conditions are zero, the response **f** is zero. If a unit step is applied to the first input at time t ; that is, $\Delta u=[1\;0\cdots0]^T$, the expected output sequence $[\hat{y}(t+1)\;\hat{y}(t+2)\cdots\hat{y}(t+N)]^T$ is equal to the first column of the matrix $\overline{\mathbf{G}}$. Specifically, the first column of the matrix $\overline{\mathbf{G}}$ can be calculated as the step response of the plant when a unit step is applied to the first control signal.

If the control signal is kept constant after the first M control moves ($\Delta u(t+j-1)=0$ for $j>M$), because of the model predictive control concept the set of predictions affecting the objective function can be expressed as

$$\hat{\mathbf{y}}=\overline{\mathbf{G}}_s\Delta\mathbf{u}_s+\mathbf{f}\,, \tag{14}$$

where

$$\overline{\mathbf{G}}_s=\begin{bmatrix} g_0 & 0 & \cdots & 0\\ g_1 & g_0 & \cdots & 0\\ & & \ddots & \vdots\\ \vdots & \vdots & & g_0\\ & & & \vdots\\ g_{N-1} & g_{N-2} & \cdots & g_{N-M}\end{bmatrix},$$
$$\Delta\mathbf{u}_s=\left[\Delta u(t)\;\;\Delta u(t+1)\;\cdots\;\Delta u(t+M-1)\right]^T.$$

The objective function of Eq. (1), including the summation form (Σ), can be rewritten as the following matrix-vector form:

$$J=\frac{1}{2}(\hat{\mathbf{y}}-\mathbf{w})^T\widetilde{\mathbf{Q}}(\hat{\mathbf{y}}-\mathbf{w})+\frac{1}{2}\Delta\mathbf{u}_s^T\widetilde{\mathbf{R}}\Delta\mathbf{u}_s\,, \tag{15}$$

$$\text{subject to constraints}\begin{cases}\Delta u(t+j-1)=0\;\text{ for } j>M\,,\\ u_{\min}\le u(t)\le u_{\max}\,,\\ -du_{\max}\le\Delta u(t)\le du_{\max}\,.\end{cases} \tag{16}$$

where $\widetilde{\mathbf{Q}}=diag\,(Q,\cdots,Q)$ is a diagonal matrix consisting of N diagonal elements, Q, and $\widetilde{\mathbf{R}}=diag\,(R,\cdots,R)$ is a diagonal matrix consisting of M diagonal elements, R. Usually, $\widetilde{\mathbf{Q}}=\mathbf{I}_{N\times N}$ and $\widetilde{\mathbf{R}}=\omega\times\mathbf{I}_{M\times M}$ are used and ω is known as an input-weighting factor.

3.3 Recursive Parameter Estimation

The process model is estimated recursively every time step to reflect the time-varying conditions of the plant, including the fuel burnup, control rod movement and other conditions. Eq. (9) can be expressed as the following inner product of the parameter vector $\hat{\theta}(t)$ and the measurement vector $\varphi(t)$:

$$\hat{y}(t+1)=F_1(q^{-1})y(t)+G_1(q^{-1})\Delta u(t)=\hat{\boldsymbol{\theta}}^T(t)\cdot\boldsymbol{\varphi}(t)\,, \tag{17}$$

where

$$\hat{\boldsymbol{\theta}}^T(t)=\left[\hat{a}_1(t)\,\hat{a}_2(t)\cdots\hat{a}_{nA}(t)\;\hat{b}_0(t)\;\hat{b}_1(t)\cdots\hat{b}_{nB}(t)\right],$$
$$\boldsymbol{\varphi}^T(t)=\left[-y(t)-y(t-1)\cdots-y(t-nA+1)\,\Delta u(t)\,\Delta u(t-1)\cdots\Delta u(t-nB)\right].$$

The parameter vector $\hat{\theta}(t)$ is estimated using a recursive least-squares method as follows:

$$\hat{\boldsymbol{\theta}}(t)=\hat{\boldsymbol{\theta}}(t-1)+\boldsymbol{\Sigma}(t)\boldsymbol{\varphi}(t-1)\left[y(t)-\hat{\boldsymbol{\theta}}^T(t-1)\cdot\boldsymbol{\varphi}(t-1)\right], \tag{18}$$

$$\boldsymbol{\Sigma}(t)=\frac{1}{\lambda(t)}\left[\boldsymbol{\Sigma}(t-1)-\frac{\boldsymbol{\Sigma}(t-1)\boldsymbol{\varphi}(t-1)\boldsymbol{\varphi}^T(t-1)\boldsymbol{\Sigma}(t-1)}{\lambda(t)+\boldsymbol{\varphi}^T(t-1)\boldsymbol{\Sigma}(t-1)\boldsymbol{\varphi}(t-1)}\right], \tag{19}$$

where the covariance matrix $\Sigma(0)>0$ and $0<\lambda(t)\le 1$. A forgetting factor $\lambda(t)$ is usually used to account for the exponential decay of the past data while tracking a slow drift in parameters. $\lambda(t)$ is calculated from the following equation:

$$\lambda(t)=\lambda_0\lambda(t-1)+(1-\lambda_0)\quad\text{with}\quad\lambda_0\le 1\;\text{ and }\;\lambda(0)\le 1\,. \tag{20}$$

The parameters estimated by Eqs. (18) through (20) are used to predict the future outputs over prediction horizon N.

4. OPTIMIZATION OF THE MODEL PREDICTIVE CONTROLLER BY A GENETIC ALGORITHM (GA)

The objective function of Eq. (15) can be solved by linear matrix inequality (LMI) techniques. In this study, a genetic algorithm is used to minimize the objective function with multiple objectives. The genetic algorithm is known to be useful for solving multiple objective functions. Compared to the conventional optimization methods that move from one point to another, genetic algorithms simultaneously start from many points and climb many peaks in parallel. Accordingly, genetic algorithms are less susceptible to becoming stuck at local minima than are conventional search methods [15, 16].

In the genetic algorithm, the term chromosome refers to a candidate solution that minimizes a cost function. As the generation proceeds, populations of chromosomes are iteratively altered by biological mechanisms inspired by natural evolution mechanisms such as selection, crossover, and mutation. The genetic algorithms require a fitness function that assigns a score to each chromosome (candidate solution) in the current population; additionally, they maximize the fitness function value. The fitness function evaluates the extent to which each candidate solution is suitable for the specified objectives. The genetic algorithm starts with an initial population of chromosomes, which represent possible solutions of the optimization problem. For each chromosome, the fitness function is computed. New generations are produced by the genetic operators known as selections, crossovers and mutations. The algorithm stops after the maximum allowed time has passed.

In the following description [17], a chromosome will be represented by s_l of which elements consist of present and future control inputs. The chromosome will have the following structure:

$$s_l = \left[u_l(t) \quad u_l(t+1) \quad \cdots \quad u_l(t+M-1) \right] . \tag{21}$$

Assuming the number L of chromosomes which will constitute the initial population, the crossover probability P_c and the mutation probability P_m, the algorithm proceeds according to the following steps:

Step 1 (initial population): Set the number of iterations *iter* =1. Generate an initial population consisting of L chromosomes of Eq. (21). The values are allocated randomly, but they should satisfy both input and input move constraints of Eq. (16). For this purpose, a simple procedure is used, as follows:

(a) Read the measured value of the input variable at the previous time point t-1, which has already been implemented.

(b) Select the current input value using the following equations:

$$u_l(t) = u(t-1) + r \cdot \Delta u_{\max} . \tag{22}$$

$$\text{If } u_l(t) \ge u_{\max}, \text{ set } u_l(t) = u_{\max} . \tag{23}$$

$$\text{If } u_l(t) \le u_{\min}, \text{ set } u_l(t) = u_{\min} . \tag{24}$$

c) Select the rest of the input moves using the following equations:

$$u_l(t+i) = u_l(t+i-1) + r \cdot \Delta u_{\max}, \quad 1 \le i \le M-1 . \tag{25}$$

$$\text{If } u_l(t+i) \ge u_{\max}, \text{ set } u_l(t+i) = u_{\max}, \quad 1 \le i \le M-1 . \tag{26}$$

$$\text{If } u_l(t+i) \le u_{\min}, \text{ set } u_l(t+i) = u_{\min}, \quad 1 \le i \le M-1 . \tag{27}$$

In the above equations, r is a random number between -1 and 1. A new random number r is generated each time Eq. (22) or Eq. (25) is used.

Step 2 (fitness function evaluation): Evaluate the objective function of Eq. (15) for all the chosen chromosomes. Following this, invert the objective function values and find the total fitness of the population as follows:

$$F = \sum_{l=1}^{L} \frac{1}{J_l(t)} , \tag{28}$$

where $J_l(t)$ is the objective function value for the l-th chromosome and the inversion of $J_l(t)$ is a fitness value of the l-th chromosome. Next, calculate the normalized fitness value of each chromosome, which indicates the selection probability P_l calculated by

$$p_l = \frac{(1/J_l(t))}{F}, \quad 1 \le l \le L . \tag{29}$$

Step 3 (selection operation): Calculate the cumulative probability q_l for each chromosome using the following equation:

$$q_l = \sum_{n=1}^{l} p_n , \quad 1 \le l \le L . \tag{30}$$

For l=1, . . ., L, generate a random number r between 0 and 1. Select the chromosome for which $q_{l-1} \le r \le q_l$. At this point of the algorithm, a new population of chromosomes has been generated. The chromosomes with high fitness values have more of a chance to be selected.

Step 4 (crossover operation): For each chromosome s_l, generate a random number r between 0 and 1. If r is lower than p_c, this particular chromosome will undergo the process of a crossover, otherwise it will remain unchanged. Mate selected chromosomes, and for each selected pair generate a random integer number z between 0 and M-1. The crossing point is the position indicated by the random number. Two new chromosomes are produced by interchanging all of the members of the parents following the crossing point. Graphically, the crossover operation can be represented as shown below, assuming that the crossover operation is applied to the parent chromosomes s_l and s_{l+1}:

$$\begin{aligned} s_l &= \left[u_l(t) \quad u_l(t+1) \quad \cdots \quad u_l(t+z-1) \;\middle|\; u_l(t+z) \quad \cdots \quad u_l(t+M-1) \right] \\ s_{l+1} &= \left[u_{l+1}(t) \quad u_{l+1}(t+1) \quad \cdots \quad u_{l+1}(t+z-1) \;\middle|\; u_{l+1}(t+z) \quad \cdots \quad u_{l+1}(t+M-1) \right] \\ &\Downarrow \text{ crossover operation} \\ s_l^{new} &= \left[u_l(t) \quad u_l(t+1) \quad \cdots \quad u_l(t+z-1) \;\middle|\; u_{l+1}(t+z) \quad \cdots \quad u_{l+1}(t+M-1) \right] \\ s_{l+1}^{new} &= \left[u_{l+1}(t) \quad u_{l+1}(t+1) \quad \cdots \quad u_{l+1}(t+z-1) \;\middle|\; u_l(t+z) \quad \cdots \quad u_l(t+M-1) \right] \end{aligned}$$

The above operation might produce infeasible offspring if the input values at the cross point do not satisfy the input move constraints. This situation is avoided by the following correction mechanism for an input variable, which modifies the values of the input parameters after the cross position so that the input move constraints are satisfied. At first, for one of the produced chromosomes s_l^{new},
if

$$u_{l+1}(t+z) - u_l(t+z-1) > \Delta u_{\max}, \tag{31}$$

then

$$\Delta = u_{l+1}(t+z) - u_l(t+z-1) - \Delta u_{\max}, \tag{32}$$

$$u_{l+1}(t+z+i) = u_{l+1}(t+z+i) - \Delta, \quad 0 \le i \le M-1-z. \tag{33}$$

if

$$u_{l+1}(t+z) - u_l(t+z-1) < -\Delta u_{\max}, \tag{34}$$

then

$$\Delta = u_l(t+z-1) - u_{l+1}(t+z) - \Delta u_{\max}, \tag{35}$$

$$u_{l+1}(t+z+i) = u_{l+1}(t+z+i) + \Delta, \quad 0 \le i \le M-1-z. \tag{36}$$

A similar set can be written for the chromosome s_{l+1}^{new}.

Step 5 (mutation operation): For each member of each chromosome s_l, $u_l(t+i)$, generate a random number r between 0 and 1. If r is lower than p_m, this particular member of the chromosome will undergo the process of mutation, otherwise it will remain unchanged. For the selected members, define the upper and lower bounds as follows:

$$\begin{aligned} b_u &= \min(\Delta u_{\max} + u(t-1),\ \Delta u_{\max} + u_l(t+i+1),\ u_{\max}) \\ b_l &= \max(-\Delta u_{\max} + u(t-1),\ -\Delta u_{\max} + u_l(t+i+1),\ u_{\min}) \end{aligned} \quad \text{if } i = 0, \tag{37}$$

$$\begin{aligned} b_u &= \min(\Delta u_{\max} + u_l(t+i-1),\ \Delta u_{\max} + u_l(t+i+1),\ u_{\max}) \\ b_l &= \max(-\Delta u_{\max} + u_l(t+i-1),\ -\Delta u_{\max} + u_l(t+i+1),\ u_{\min}) \end{aligned} \quad \text{if } 0 < i < M-1, \tag{38}$$

$$\begin{aligned} b_u &= \min(\Delta u_{\max} + u_l(t+i-1),\ u_{\max}) \\ b_l &= \max(-\Delta u_{\max} + u_l(t+i-1),\ u_{\min}) \end{aligned} \quad \text{if } i = M-1. \tag{39}$$

The above bounds define the region of values of $u_l(t+i)$ which will produce a feasible solution. This definition is followed by the generation of a random binary number b. Based on the value of b, $u_l(t+i)$ is modified by the following equations:

$$u_l(t+i) = u_l(t+i) + \left(b_u - u_l(t+i)\right)\left(1 - r^{(1-iter/iter_{\max})}\right) \quad \text{if } b = 0, \tag{40}$$

$$u_l(t+i) = u_l(t+i) - \left(u_l(t+i) - b_l\right)\left(1 - r^{(1-iter/iter_{\max})}\right) \quad \text{if } b = 1. \tag{41}$$

where r is a random number between 0 and 1, *iter* is the number of iterations performed so far and $iter_{\max}$ is the expected final number of iterations.

Step 6 (repeat or stop): If the maximum allowed time has not expired, set $iter = iter + 1$ and return the algorithm to Step 2. Otherwise, stop the algorithm and select the chromosome that produced the lowest value of the objective function throughout the entire procedure.

The above simplified genetic algorithm makes it possible to calculate the optimal control in real time.

5. APPLICATION TO NUCLEAR REACTOR POWER CONTROL

Fig. 3 shows the schematic block diagram of the model predictive controller combined with a parameter estimation algorithm. In this study, the developed power controller was applied to a three-dimensional reactor model (MASTER code) [18]. MASTER (Multipurpose Analyzer for Static and Transient Effects of a Reactor), developed by KAERI, is a nuclear analysis and design code which can simulate the Pressurized Water Reactor (PWR) and Boiling Water Reactor (BWR) cores in three-dimensional geometry. MASTER was designed to have a variety of capabilities, such as a static nuclear reactor core design, a transient nuclear reactor core analysis and operational support. The MASTER code was written in FORTRAN, and the proposed control algorithm in MATLAB [19]. Visual C++ is in charge of the variable transfer between the MASTER code and the control algorithm.

At first, a reactor power controller was designed with the use of the model predictive control optimized by a genetic algorithm and applied to the Yonggwang Unit 3 Nuclear Power Plant (YGN-3) modeled by the MASTER

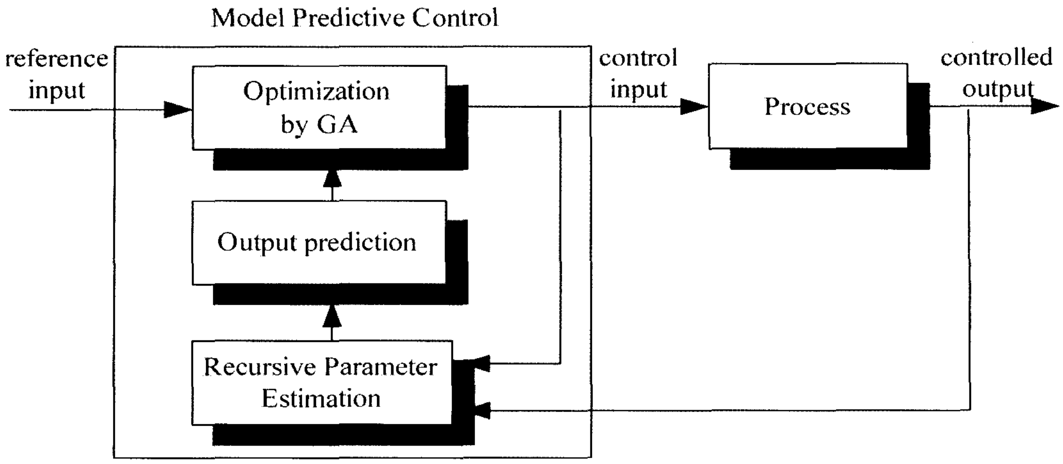


Fig. 3. Schematic Block Diagram of a Proposed Model Predictive Controller

code. The thermal power of YGN-3 is regulated by five regulating control rod banks and is also regulated by changing the concentration of boric acid, which absorbs neutrons at a high rate and is dissolved in coolant. In this work, it is assumed that the power level is controlled solely by the regulating control rod banks, R5, R4, R3, R2, and R1, and that the boric acid concentration is not changed for the short period of a few hours over which the depletion of nuclear fuel is insignificant.

A leading insertion bank of the regulating control banks is R5. The regulating control rod banks are inserted in the order of R5 first, then R4, R3, R2 and finally R1. They are overlapped (228.6cm) with one another (see Fig. 4), and these regulating banks are withdrawn in the opposite order in the case of a withdrawal. For example, in case they are inserted from the top position (381 cm), when the R5 control rod bank is inserted first and approaches a 152.4 cm (=381-228.6 in case of 228.6cm overlap) axial position, the R4 control rod bank goes into the reactor core together with the R5 control rod bank. As shown in Fig. 4, as all of the control rods will not go down below the bottom of the reactor core, the positions of all of the regulating control rod banks can be described by the pseudo position of the regulating control rod bank R5, the control input.

The model predictive controller for the power level control is subject to constraints as follows:

$$\Delta u(t+j-1)=0 \text{ for } j>M,$$
$$-914.4\,cm \le u(t) \le 381\,cm,$$
$$-1.27\times T \le \Delta u(t) \le 1.27\times T,$$

where T is a sampling time of 5 sec.

The pseudo-position of the R5 regulating control bank (the control input) is higher than four overlap lengths (914.4cm) below the bottom of the reactor core and is also lower than the top position of the reactor core (see Fig. 4). The maximum speed of the regulating control rods is 30 inches/min ($1.27T$ *cm/sec*). The optimal control input could be obtained by solving the minimization objective function of Eq. (15) using a genetic algorithm.

Most of the computation time is taken by the calculation time of the reactor dynamics; that of the controller is especially insignificant. The sampling time is five sec. The one-step simulation time is usually below 2 sec in a 3.0 GHz PC, including the reactor dynamics simulation time. The computation time of the controller by itself is approximately 0.2 sec. Therefore, it is possible to accomplish a real-time performance even in low-power computing environments.

Fig. 5 shows the simulation results. The desired power is initially 70%, and increases to 90% by the ramp from 0.13hr, and decreases to 80% by the ramp from 1.20hr. In addition, it increases from 80% to 90% by the step at 2.25hr and decreases from 90% to 80% by the step at 3.23hr. Figs. 5(a) and 5(b) show the responses of the nuclear power level and average coolant temperature. It is shown that the average coolant temperature and the power level follow the desired values for these readings very well. Fig. 5(c) shows the positions of the regulating control rod banks. This figure describes the overlapped positions well. As described above, the boric acid concentration did not change during the simulation time of 5hr (see Fig. 5(d)). Fig. 5(e) shows the trend of the best fitness function value, which was affected by the magnitudes of the estimated output error and the control input move.

Figs. 5(f) and (g) show the parameters related to $A(q^{-1})$ and $B(q^{-1})$ that are estimated recursively at every time step. It was assumed that all of the parameters of the polynomials $A(q^{-1})$ and $B(q^{-1})$ were 0.1 at the beginning of the simulation. In this study, these assumed values were estimated by the parameter estimation algorithm for the initial 250 sec, which denotes fifty time steps, as the sampling period was five sec. This was done by exciting the core dynamics with initial small random movements of the R5 regulating control bank. Subsequently, the parameters were continuously adapted according to the changing operating conditions of the control rods and the reactor power. Figs. 5(f) and (g) show that the reactor dynamics changed according to the power level and the control rod positions.

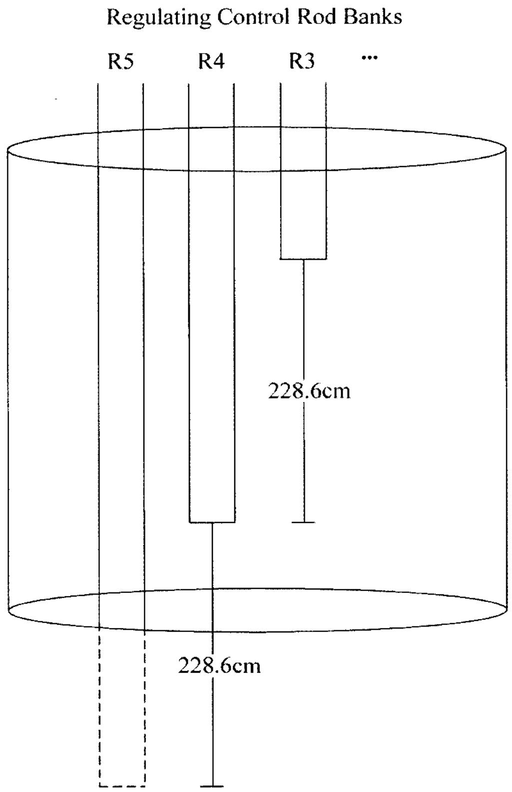
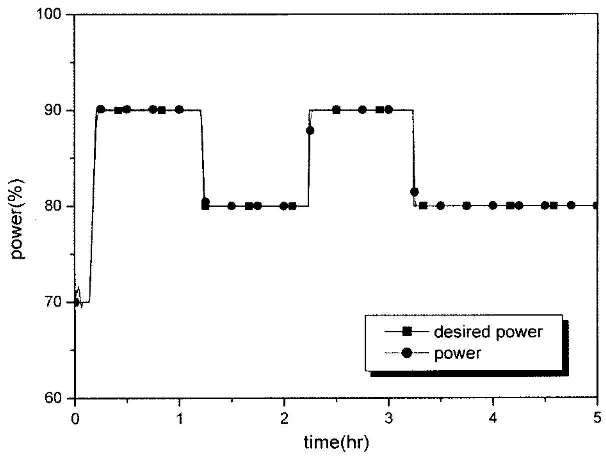
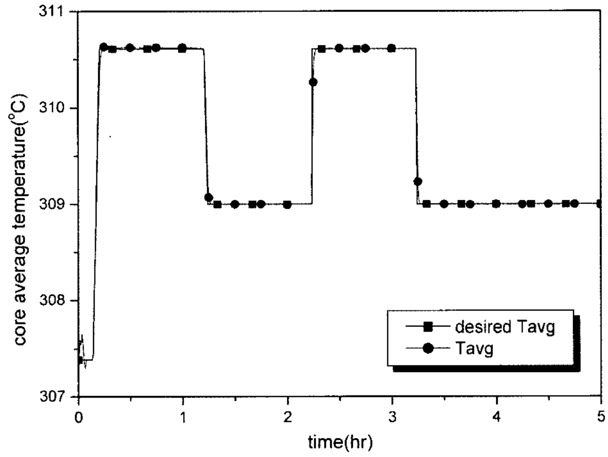


Fig. 4. Overlapped Positions of the Regulating Control Rod Banks

A conventional proportional-integral (PI) controller was additionally designed in order to compare the performances of the power level response with the proposed model predictive controller as optimized by the GA. As shown in Fig. 6, the existing PI controller showed an inferior performance compared to the proposed model predictive controller. In addition, if nonlinear characteristics are strong

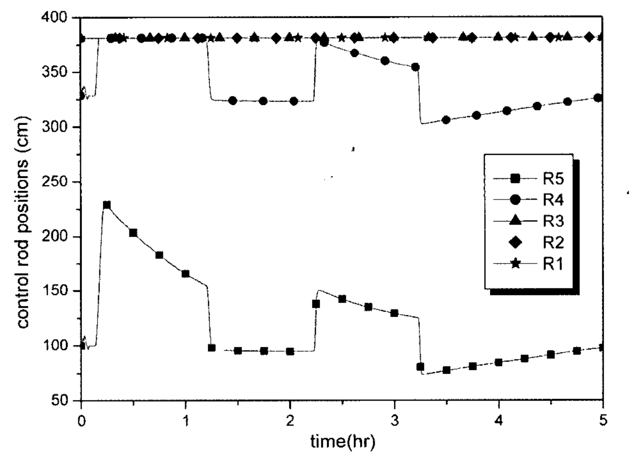


(a) power level

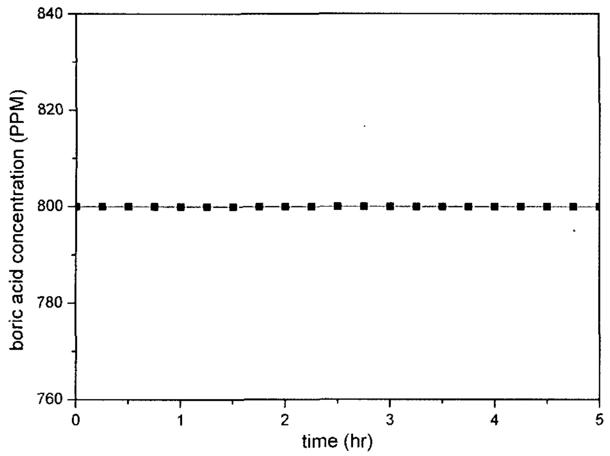


(b) average coolant temperature

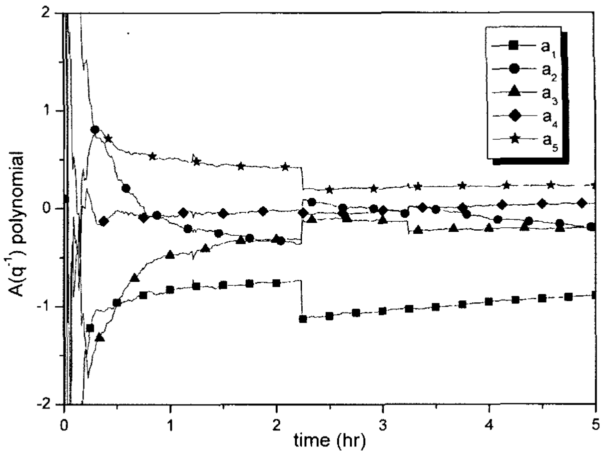
Fig. 5. Power Control Performance by the Proposed Controller



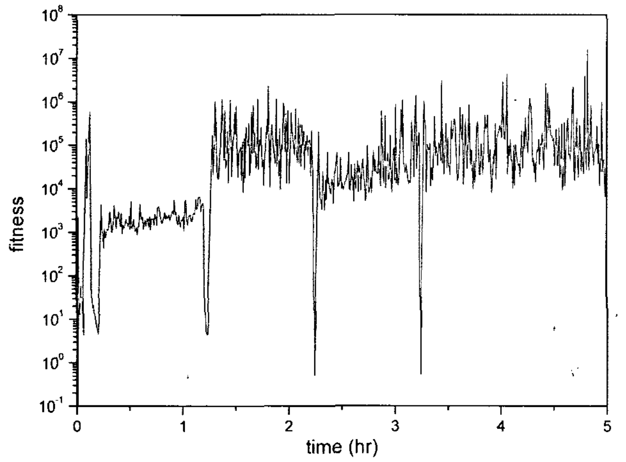
(c) regulating control rod bank position



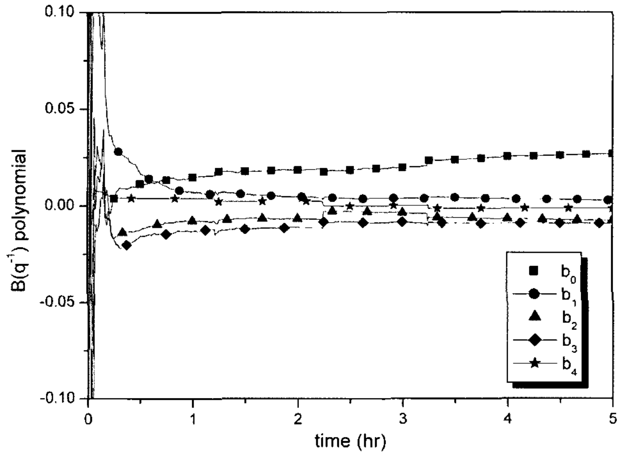
(d) boric acid concentration



(f) $A(q^{-1})$ polynomial



(e) best fitness value of the genetic algorithm



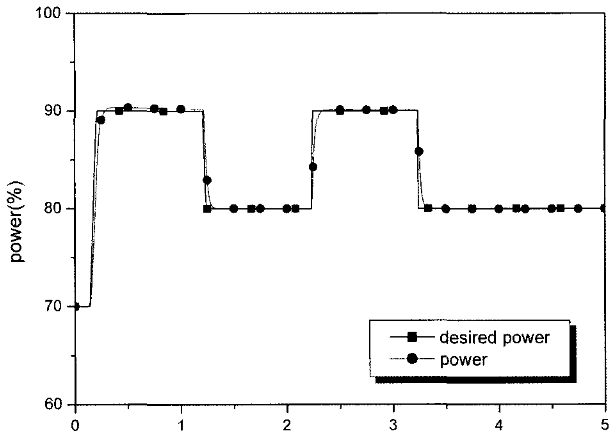
(g) $B(q^{-1})$ polynomial

Fig. 5. Continued

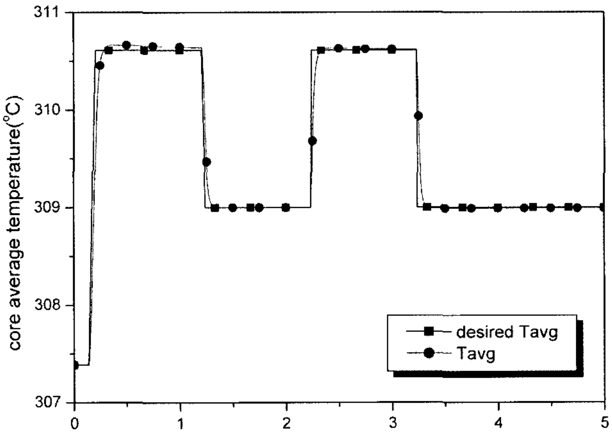
on account of the nuclear fuel usage and a boric acid concentration change (which are not considered in this work) it is expected that the proposed model predictive controller will display a much better performance than the PI controller because it is optimized at each time step.

6. CONCLUSIONS

In this study, the model predictive controller optimized by a GA was developed to control the nuclear power in pressurized water reactors. The developed controller has been applied to YGN-3, which was modeled by the MASTER code. Additionally, a controller design model used for designing the model predictive controller was estimated at every time step by applying a recursive parameter estimation algorithm to reflect the time-varying condition. It was known that the proposed controller controls the control rod position so that the average coolant temperature tracks very well its setpoint change according to load. In addition, the reactor power tracks the demand load very well. From these numerical simulation results, both the performances of the proposed controller for the 5%/min ramp increase or decrease of a desired load as well as its 10% step increase or decrease (which are design requirements) are proved to be satisfactory.

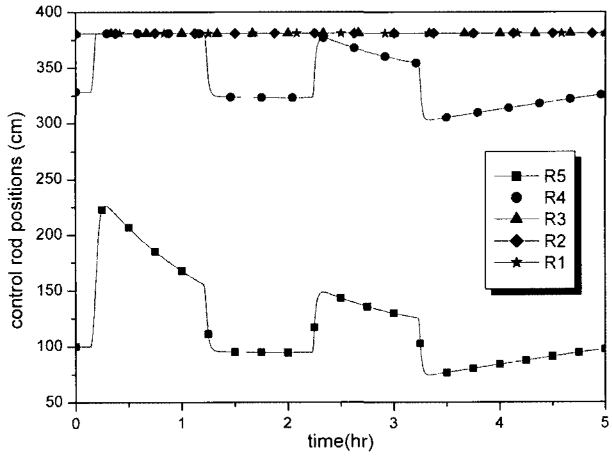


(a) power level

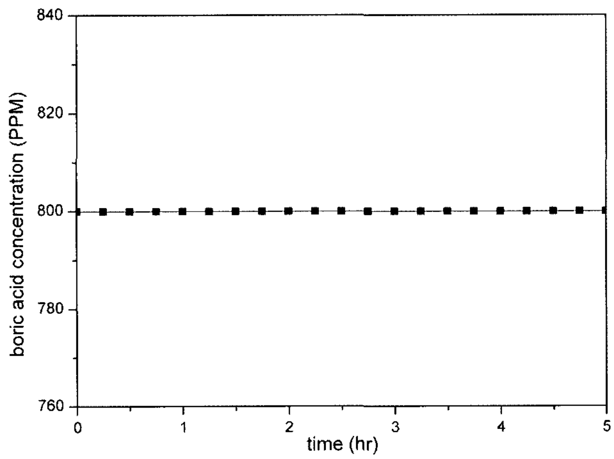


(b) average coolant temperature

Fig. 6. Power Level Control Performance by an Existing PI Controller



(c) regulating control rod bank position



(d) boric acid concentration

Fig. 6. Continued

REFERENCES

[1] W. H. Kwon and A. E. Pearson, "A Modified Quadratic Cost Problem and Feedback Stabilization of a Linear System," *IEEE Trans. Automatic Control*, **22**, p. 838 (1977).

[2] J. Richalet, A. Rault, J. L. Testud, and J. Papon, "Model Predictive Heuristic Control: Applications to Industrial Processes," *Automatica*, **14**, p. 413 (1978).

[3] C. E. Garcia, D. M. Prett, and M. Morari, "Model Predictive Control: Theory and Practice – A Survey," *Automatica*, **25**, p. 335 (1989).

[4] D. W. Clarke, and R. Scattolini, "Constrained Receding-Horizon Predictive Control," *IEE Proceedings-D*, **138**, p. 347 (1991).

[5] M. V. Kothare, V. Balakrishnan, and M. Morari, "Robust Constrained Model Predictive Control Using Linear Matrix Inequality," *Automatica*, **32**, p. 1361 (1996).

[6] J. W. Lee, W. H. Kwon, and J. H. Lee, "Receding Horizon Tracking Control for Time-Varying Discrete Linear Systems," *Intl. J. Control*, **68**, p. 385 (1997).

[7] J. W. Lee, W. H. Kwon, and J. Choi, "On Stability of Constrained Receding Horizon Control with Finite Terminal Weighting Matrix," *Automatica*, **34**, p. 1607 (1998).

[8] M. G. Na, "Design of a Receding Horizon Control System for Nuclear Reactor Power Distribution," *Nucl. Sci. Eng.*, **138**, p. 305 (2001).

[9] Man Gyun Na, Sun Ho Shin, and Whee Cheol Kim, "A Model Predictive Controller for Nuclear Reactor Power," *J. Korean Nucl. Soc.*, **35**, p. 399 (2003).

[10] N. Z. Cho and L. M. Grossman, "Optimal Control for Xenon Spatial Oscillations in Load Follow of a Nuclear Reactor," *Nucl. Sci. Eng.*, **83**, p. 136 (1983).

[11] P.P. Niar and M. Gopal, "Sensitivity-Reduced Design for a Nuclear Pressurized Water Reactor," *IEEE Trans. Nucl. Sci.*, **NS-34**, p. 1834 (1987).
[12] C. Lin, J.-R. Chang, and S.-C. Jenc, "Robust Control of a Boiling Water Reactor," *Nucl. Sci. Eng.*, **102**, p. 283 (1989).
[13] M. G. Park and N. Z. Cho, "Time-Optimal Control of Nuclear Reactor Power with Adaptive Proportional-Integral Feedforward Gains," *IEEE Trans. Nucl. Sci.*, **40**, p. 266 (1993).
[14] E. F. Camacho and C. Bordons, *Model Predictive Control*, Springer-Verlag, London (1999).
[15] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley, Reading, MA (1989).
[16] M. Mitchell, *An Introduction to Genetic Algorithms*, MIT Press, Cambridge, MA (1996).
[17] H. Sarimveis and G. Bafas, "Fuzzy Model Predictive Control of Non-linear Processes Using Genetic Algorithms," *Fuzzy Sets and Systems*, **139**, p. 59 (2003).
[18] B. O. Cho, H. G. Joo, J. Y. Cho and S. Q. Zee., "MASTER: Reactor Core Design and Analysis Code," *Proc. 2002 Int. Conf. New Frontiers of Nuclear Technology: Reactor Physics (PHYSOR 2002)*, Seoul, Korea, Oct. 7-10, 2002.
[19] MathWorks, *MATLAB 5.3 (Release 11)*, The MathWorks, Natick, Massachusetts (1999).