

Multi-objective Quantum-inspired Evolutionary Algorithm-based Optimal Control of Two-link Inverted Pendulum

In-Won Park, Bum-Joo Lee, Ye-Hoon Kim, Ji-Hyeong Han and Jong-Hwan Kim

Abstract— This paper proposes a method to generate an optimal trajectory of nonlinear dynamical system and concurrently optimize multiple performance criteria. As the dimensionality of system increases, it is difficult to find values of cost/reward function of conventional optimal controllers. In order to solve this problem, the proposed method employs iterative linear quadratic regulator and multi-objective quantum-inspired evolutionary algorithm to generate various optimal trajectories that satisfy multiple performance criteria. Fuzzy measure and fuzzy integral are also employed for global evaluation by integrating the partial evaluation of each solution over criteria with respect to user's degree of consideration for each criterion. Effectiveness of the proposed method is verified by computer simulation carried out for the problem of stabilizing two-link inverted pendulum model.

I. INTRODUCTION

Learning optimal controllers for high-dimensional and nonlinear dynamical systems in continuous state, action, and time spaces have received a great deal of attention in recent times. When learning controllers having discretized state and action space, one is confronted with the dimensionality of domain and the computational complexity. To cope with the problem, trajectory-based techniques, such as differential dynamic programming (DDP) [1], [2] and iterative linear quadratic regulator (ILQR) [3], were proposed to compute a locally optimal feedback control law.

Controllers that generate an optimal trajectory is required for the performance criteria, such as rise time, settling time, and percentage overshoot, to be ensured simultaneously. Depending on the initialization of cost/reward function of DDP and ILQR, these performance criteria significantly change as the dynamical systems become more complex. In this way, many real-world optimization problems involve several objectives that conflict each other at the same time, which are known as multi-objective optimization problems (MOPs).

Multi-objective evolutionary algorithm (MOEA) utilizes the concept of Pareto-optimal solution to obtain a set of nondominated solutions, where the preferred solution is selected and applied in real-world application. Strength Pareto evolutionary algorithm (SPEA) [4] and its improved version, SPEA2 [5], were proposed to approximate the Pareto-optimal set for MOPs by employing a refined fitness assignment

In-Won Park, Ye-Hoon Kim, Ji-Hyeong Han and Jong-Hwan Kim are with the Department of Electrical Engineering, KAIST, 373-1 Guseong-dong, Yuseong-gu, Daejeon, 305-701, Republic of Korea (phone: +82-42-350-8048; email: {iwpark, yhkim, jhhan, johkim}@rit.kaist.ac.kr).

Bum-Joo Lee is with the Information & Electronics Research Institute, KAIST, 373-1 Guseong-dong, Yuseong-gu, Daejeon, 305-701, Republic of Korea (phone: +82-42-350-8048; email: leebumjoo@gmail.com).

and an enhanced archive truncation technique. Nondominated sorting genetic algorithm (NSGA) [6] and its improved version, NSGA-II [7], were proposed to efficiently maintain diversity using a nondominated sorting and a crowding distance assignment.

This paper aims at proposing a method, which generates an optimal trajectory of nonlinear dynamical system and optimizes a set of solutions to satisfy the multiple performance criteria by using ILQR and MOEA, respectively. Multi-objective quantum-inspired evolutionary algorithm (MQEA) [8], [9] is used as MOEA, which is an extended version of quantum-inspired evolutionary algorithm (QEA) for MOPs [10], [11]. Due to the inherent probabilistic mechanism of QEA, it starts with a global search scheme and automatically changes to a local search scheme as the generation progresses, leading to a good balance between exploration and exploitation [12]. The quality of solutions is improved during the multiple observation process of MQEA, while preserving diversity.

Among the set of nondominated solutions obtained from MQEA, a preferred solution has to be selected based on the partial evaluation over each objective and the user's preference for each objective. In this paper, fuzzy measure is employed to represent the user's preference as the degree of consideration and fuzzy integral is to calculate the value of global evaluation of each solution considering the partial evaluation and the user's preference. Utilizing fuzzy measure and fuzzy integral performs well for MOPs, where the performance criteria are dependent on one another.

To verify the effectiveness of proposed method, computer simulation is carried out for the problem of stabilizing a two-link inverted pendulum model. The two-link inverted pendulum model is a nonlinear dynamical system, which represents the dominant dynamics of humanoid robot (upper and lower body) in double support phase. Simulation results confirm that the proposed method has generated various link trajectories that satisfy multiple performance criteria.

This paper is organized as follows: Section II briefly reviews iterative linear quadratic regulator, which is used as an optimal controller. Section III presents the basis of QEA and the structure of MQEA to deal with multi-objectives. Section IV presents the method of selecting a preferred solution among nondominated solutions using fuzzy measure and fuzzy integral. Section V and Section VI describe the optimal control problem to be solved and its simulation results, respectively. Concluding remarks follow in Section VII.

II. OPTIMAL CONTROL METHOD

A. Problem Definition

We consider the discrete-time nonlinear dynamical systems, $x_{k+1} = g(x_k, u_k)$, with states $x_k \in \mathbb{R}^{n_x}$ and controls $u_k \in \mathbb{R}^{n_u}$. The cost function is expressed in the quadratic form as follows:

$$J = \frac{1}{2}(x_N - x^*)^T Q_f (x_N - x^*) + \frac{1}{2} \sum_{k=0}^{N-1} (x_k^T Q x_k + u_k^T R u_k) \quad (1)$$

where x_N and x^* represent the final state and given target, respectively. Q_f and Q are the state cost-weighting matrices and R is the control cost-weighting matrix, where all these matrices are symmetric and positive definite. The control objective is to find the optimal trajectory of states by using ILQR, which minimizes the cost function over a time step, N .

B. Iterative Linear Quadratic Regulator

ILQR [3] is an iterative scheme, which starts with a nominal control sequence, u_k , and a corresponding nominal trajectory, x_k , to calculate the sequence of optimal control input by linearizing the system dynamics around u_k and x_k . If δu_k and δx_k are deviations from nominal u_k and x_k , the nonlinear dynamical system can be linearized as follows:

$$\delta x_{k+1} = A_k \delta x_k + B_k \delta u_k \quad (2)$$

where A_k and B_k are evaluated Jacobians along x_k and u_k with respect to x and u , respectively. The optimal control improvement, δu_k , is found by solving the state equation (2), the costate equation

$$\delta \lambda_k = A_k^T \delta \lambda_{k+1} + Q(x_k + \delta x_k) \quad (3)$$

and the stationary condition

$$0 = R(u_k + \delta u_k) + B_k^T \delta \lambda_{k+1} \quad (4)$$

where $\delta \lambda_k$ is the Lagrange multiplier. Every iteration is comprised of two sweeps of trajectory: a backward and a forward sweep.

In the backward sweep, unknown sequences of S_k and v_k are calculated to determine the Lagrange multiplier as follows:

$$\delta \lambda_k = S_k \delta x_k + v_k \quad (5)$$

with the boundary conditions,

$$S_N = Q_f, \quad v_N = Q_f(x_N - x^*). \quad (6)$$

Given the final condition of S_k and v_k , the sequences of S_k and v_k are calculated by substituting (5) into (2) and (3) as follows:

$$S_k = A_k^T S_{k+1} [I - B_k (B_k^T S_{k+1} B_k + R)^{-1} B_k^T S_{k+1}] A_k + Q \quad (7)$$

and

$$v_k = A_k^T v_{k+1} - A_k^T S_{k+1} [I - B_k (B_k^T S_{k+1} B_k + R)^{-1} B_k^T S_{k+1}] B_k R^{-1} B_k^T v_{k+1} - A_k^T S_{k+1} [I - B_k (B_k^T S_{k+1} B_k + R)^{-1} B_k^T S_{k+1}] B_k u_k + Q x_k. \quad (8)$$

In the forward sweep of ILQR iteration, the optimal control improvement, δu_k , is calculated by substituting (2) and (5) into (4) as follows:

$$\delta u_k = - (R + B_k^T S_{k+1} B_k)^{-1} B_k^T S_{k+1} A_k \delta x_k - (R + B_k^T S_{k+1} B_k)^{-1} B_k^T v_{k+1} - (R + B_k^T S_{k+1} B_k)^{-1} R u_k. \quad (9)$$

Then an improved control sequence can be found as, $u_k^* = u_k + \delta u_k$, which results in a new nominal trajectory, x_k^* .

C. Optimizing the Cost-weighting Matrix

The key objective of optimal control problems is to find the optimal trajectory of states from starting point to target within a minimal time frame. However, performance criteria, such as rise time, settling time, and percentage overshoot of state trajectories, vary according to the values of state and control cost-weighting matrices. Thus, optimization method is used to find the values of Q_f , Q , and R that minimize these performance criteria.

In the case of single-objective evolutionary algorithm, the fitness of solution is evaluated by summing up the values of each fitness. However, multi-objective evolutionary algorithm is more suitable for satisfying multiple performance criteria simultaneously because they conflict with each other. As an example, there exists no solution, which satisfies both fastest rise time and lowest percentage overshoot.

The objective functions in this MOP are defined as follows:

$$\begin{aligned} f_1 &= K_r \cdot t_r, \\ f_2 &= K_s \cdot t_s, \\ f_3 &= K_{po} \cdot \%PO \end{aligned} \quad (10)$$

where f_1 , f_2 , and f_3 correspond to the objective function of rise time (t_r), settling time (t_s), and percentage overshoot ($\%PO$), respectively. Note that K_r , K_s , and K_{po} are constants that balance the objective function values. Multi-objective quantum-inspired evolutionary algorithm (MQEA) described in Section III is used to find the nondominated solutions of optimal trajectory generated by the ILQR.

III. MULTI-OBJECTIVE OPTIMIZATION METHOD

A. Quantum-inspired Evolutionary Algorithm

Instead of binary, numeric or symbolic representation, quantum-inspired evolutionary algorithm (QEA) [10] utilizes Q-bit as a probabilistic representation, which is based on the concept of qubits in quantum computing [13]. QEA handles the balance between exploration and exploitation more easily as compared to conventional genetic algorithms. QEA also explores the search space with a small number of individuals and exploits a global solution in the search space within a short span of time.

Q-bit is the smallest unit of information used in QEA, which is defined as the pair of numbers, (α, β) , satisfying $|\alpha|^2 + |\beta|^2 = 1$. Note that $|\alpha|^2$ and $|\beta|^2$ represent probabilities of Q-bit being found in ‘0’ and ‘1’ state, respectively. Q-bit may be in either ‘0’ or ‘1’ state, or in the linear superposition of the two. Q-bit individual is defined as a string of Q-bits as follows:

$$\mathbf{q}_j^t = \left[\begin{array}{c|c|c|c} \alpha_{j1}^t & \alpha_{j2}^t & \cdots & \alpha_{jm}^t \\ \beta_{j1}^t & \beta_{j2}^t & \cdots & \beta_{jm}^t \end{array} \right] \quad (11)$$

where m is the number of Q-bits. QEA maintains the population of Q-bit individuals, $Q(t) = \{\mathbf{q}_1^t, \mathbf{q}_2^t, \dots, \mathbf{q}_n^t\}$, at generation t , where n is the size of population and \mathbf{q}_j^t , $j = 1, 2, \dots, n$, is the Q-bit individual.

During the evolutionary process, more diverse individuals are generated because the Q-bit individual represents linear superposition of all possible states probabilistically. The detailed procedure of QEA and its structure for single-objective optimization problems are described in [10].

B. Multi-objective Quantum-inspired Evolutionary Algorithm

Multi-objective quantum-inspired evolutionary algorithm (MQEA) enhances the proximity of nondominated solutions to Pareto-optimal front [8], [9]. MQEA combines the framework of QEA and MOEA such as NSGA-II to minimize each objective function stated in Section II.C. Algorithm 1 summarizes the overall procedure of MQEA, where each step is described in detail as follows.

i) The first step is to initialize α_i^0 and β_i^0 of \mathbf{q}_j^0 in $Q_k(0)$ with $1/\sqrt{2}$, where $i = 1, 2, \dots, m$, m is the string length of Q-bit individual, $j = 1, 2, \dots, n$, n is the size of subpopulation, and $k = 1, 2, \dots, s$, s is the number of subpopulation. In other words, one Q-bit individual, \mathbf{q}_j^0 , represents linear superposition of all possible states with the same probability.

ii) Binary solutions, $P_k(0) = \{\mathbf{x}_1^0, \mathbf{x}_2^0, \dots, \mathbf{x}_n^0\}$, are evaluated by observing the states of $Q_k(0)$ at generation $t = 0$. One binary solution, \mathbf{x}_j^0 , is formed by selecting either 0 or 1 for each bit using the probability, either $|\alpha_i^0|^2$ or $|\beta_i^0|^2$, $i = 1, 2, \dots, m$, of \mathbf{q}_j^0 as follows:

$$x_i^0 = \begin{cases} 0 & \text{if rand}[0,1] \geq |\beta_i^0|^2 \\ 1 & \text{if rand}[0,1] < |\beta_i^0|^2 \end{cases} \quad (12)$$

Algorithm 1 Pseudo code for MQEA

```

begin
     $t \leftarrow 0$ 
    i) initialize  $Q_k(t)$ 
    ii) make  $P_k(t)$  by observing the states of  $Q_k(t)$ 
    iii) evaluate  $P_k(t)$ 
    iv) store all solutions in  $P_k(t)$  into  $P(t)$  and
        nondominated solutions in  $P(t)$  to  $A(t)$ 
    while (not termination condition) do
         $t \leftarrow t + 1$ 
    v) make  $P_k(t)$  by observing the states of  $Q_k(t-1)$ 
    vi) evaluate  $P_k(t)$ 
    vii) run the fast nondominated sort and crowding
        distance sort assignment  $P_k(t) \cup P_k(t-1)$ 
    viii) form  $P_k(t)$  by the first  $n$  individuals in the
        sorted population of size  $2n$ 
    ix) store all solutions in every  $P_k(t)$  into  $P(t)$ 
    x) form  $A(t)$  by nondominated solutions in
         $A(t-1) \cup P(t)$ 
    xi) migrate randomly selected solutions in  $A(t)$ 
        to every  $R_k(t)$ 
    xii) update  $Q_k(t)$  using Q-gates referring to the
        solutions in  $R_k(t)$ 
end

```

Since α_i^0 and β_i^0 of \mathbf{q}_j^0 in $Q_k(0)$ are initialized with the same value, binary solutions are randomly generated.

iii) The fitness of each binary solution in $P_k(0)$, \mathbf{x}_j^0 , is evaluated.

iv) Solutions of the global population, $P(0) = \{\mathbf{x}_1^0, \mathbf{x}_2^0, \dots, \mathbf{x}_n^0, \dots, \mathbf{x}_N^0\}$, are filled with all solutions in every $P_k(0)$, where N ($n \cdot s$) is the size of global population in MQEA. Then, the nondominated solutions in $P(0)$ are copied to the archive, $A(0) = \{\mathbf{a}_1^0, \mathbf{a}_2^0, \dots, \mathbf{a}_l^0\}$, where l is the size of current archive satisfying $l \leq N$.

v), vi) Binary solutions, $P_k(t)$, are formed by observing multiple states of $Q_k(t-1)$ as in step ii), and then the fitness of each binary solution is evaluated. Based on the dominance check, \mathbf{x}_j^t is replaced by the best \mathbf{x}_{jo}^t , where o is the observation index.

vii) The procedure of fast nondominated sorting is as follows: nondominated front is found and temporarily saved to search the next nondominated front. This procedure is repeated until all individuals are ranked.

The calculation of normalized crowding distance estimates the density of each individual. This density information is utilized to select individuals in the population for the next generation. The crowding distance of individual refers to the average side length of the cuboid that has the vertices of the nearest neighbors.

viii) Superior (high ranked) n individuals in a generation survive such that survived individuals form $P_k(t)$. Q-bit individuals in $Q_k(t)$ are also rearranged according to the corresponding individuals in $P_k(t)$, where $P_k(t)$ becomes

the parent population in the next generation. $Q_k(t)$ will be updated by the strategy of step xii).

ix) $P(t)$ is filled with all solutions in every $P_k(t)$.

x) Only the nondominated solutions in $(A(t-1) \cup P(t))$ form $A(t)$. If the number of nondominated solutions is larger than the size of global population, $A(t)$ is filled with random solutions among the nondominated ones in $(A(t-1) \cup P(t))$ until l is equal to N .

xi) Compared to the QEA, the replacement method maintains multiple global best solutions, which provides diversity for multi-objective optimization problems. Note that the global random migration process occurs in every generation, i.e. global migration period is one.

xii) The fitness values of \mathbf{r}_j^t and \mathbf{x}_j^t in each subpopulation are compared to decide the updating direction of Q-bit individuals. Instead of crossover and mutation, the rotation gate $U(\Delta\theta)$ is employed as an update operator for Q-bit individuals, which is defined as follows [10]:

$$\mathbf{q}_j^t = U(\Delta\theta) \cdot \mathbf{q}_j^{t-1}, \quad (13)$$

$$U(\Delta\theta) = \begin{bmatrix} \cos(\Delta\theta) & -\sin(\Delta\theta) \\ \sin(\Delta\theta) & \cos(\Delta\theta) \end{bmatrix} \quad (14)$$

where $\Delta\theta$ is the rotation angle of each Q-bit.

IV. PREFERRED SOLUTION SELECTION METHOD

Among numerous nondominated solutions generated from MQEA, one solution must be selected in order to be applied in real-world application. These solutions cannot be directly compared against each other because they are optimized to multiple performance criteria. Thus, preference-based solution selection algorithm (PSSA) is used to calculate a value of global evaluation of each nondominated solution by considering the user's degree of consideration and partial evaluation over each objective.

In PSSA, the user's preference degree of each objective is represented by fuzzy measure [14], [15] and then global evaluation of each nondominated solution is calculated by using fuzzy integral. Algorithm 2 summarizes the overall procedure of PSSA, where each step is described in detail as follows.

i) The objectives of MOP are defined as the performance criteria of fuzzy integral.

ii) λ -fuzzy measure is used to represent the user's preference degree of each criterion, which is classified as belief measure, plausibility measure, and probability measure. λ is determined by the interaction degree, ξ , where $\lambda = \frac{(1-\xi)^2}{\xi^2} - 1$. Depending on the value of λ , fuzzy measure is considered as belief measure ($\lambda > 0$), plausibility measure ($\lambda < 0$), and probability measure ($\lambda = 0$).

To find the values of λ -fuzzy measures of the power set, $P(C)$ of a set of performance criteria, C , a pairwise comparison matrix is initially defined by user for representing relative preference degree of each criterion [16]. Secondly, the normalized weights are calculated by adding each row of

Algorithm 2 Pseudo code for PSSA

$C = \{c_1, c_2, \dots, c_n\}$: a set of performance criteria
 n : the number of performance criteria
 $P(C)$: a power set of C
 m : the number of nondominated solutions
begin
i) define the set of objectives in MOP as C
ii) calculate λ -fuzzy measures of $P(C)$
iii) normalize the nondominated solutions to get the partial evaluation value
iv) calculate the global evaluation value using fuzzy integral for λ -fuzzy measures and partial evaluation values
v) select the one with the highest global evaluation value as the preferred solution
end

pairwise comparison matrix and dividing it by the total sum of the related matrix. Lastly, λ -fuzzy measures are calculated using the normalized weights [15].

iii) The value of partial evaluation of each nondominated solution over each objective is calculated by normalizing the objective function value to 1.

iv) The value of global evaluation of each nondominated solution is calculated by using fuzzy integral for the λ -fuzzy measure and the value of partial evaluation, which are obtained from previous steps [14].

v) The one with the highest value of global evaluation is selected as the preferred solution.

V. OPTIMAL CONTROL PROBLEM

The control objective is to stabilize a two-link inverted pendulum shown in Fig. 1 into an upright position with the minimum rise time, settling time, and overshoot percentage. The two-link inverted pendulum model has the following dynamic equation:

$$\tau = M(\theta)\ddot{\theta} + V(\theta, \dot{\theta}) + G(\theta) \quad (15)$$

where τ is the control input, and M , V , and G represent mass matrix, centrifugal and Coriolis matrix, and gravity matrix, respectively. When four outputs - two link angles (θ_1 and θ_2) and their angular velocities ($\dot{\theta}_1$ and $\dot{\theta}_2$) - are measured, the model is controlled by two input torques (τ_1 and τ_2).

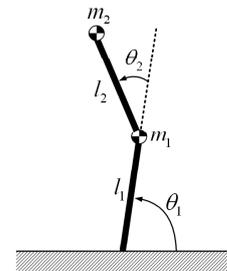


Fig. 1. Schematic drawing of two-link inverted pendulum.

The mass matrix, $M(\theta)$, the centrifugal and Coriolis matrix, $V(\theta, \dot{\theta})$, and the gravity matrix, $G(\theta)$, are defined as follows:

$$M(\theta) = \begin{bmatrix} l_1^2(m_1 + m_2) + 2P_1 + P_2 & P_1 + P_2 \\ P_1 + P_2 & P_2 \end{bmatrix}, \quad (16)$$

$$V(\theta, \dot{\theta}) = \begin{bmatrix} -2P_3\dot{\theta}_1\dot{\theta}_2 - P_3\dot{\theta}_2^2 \\ P_3\dot{\theta}_1^2 \end{bmatrix}, \quad (17)$$

$$G(\theta) = \begin{bmatrix} l_1(m_1 + m_2)g \cos \theta_1 + P_4 \\ P_4 \end{bmatrix} \quad (18)$$

where

$$\begin{aligned} P_1 &= l_1 l_2 m_2 \cos \theta_2, & P_2 &= l_2^2 m_2, \\ P_3 &= l_1 l_2 m_2 \sin \theta_2, & P_4 &= l_2 m_2 g \cos(\theta_1 + \theta_2). \end{aligned} \quad (19)$$

The values of link mass, m , and length, l , are set as 1.0 kg and 1.0 m, respectively. It is assumed that the link is a weightless telescopic limb and mass is concentrated at a single point.

The forward dynamics can be computed based on the equations (15)-(19) as follows:

$$\ddot{\theta} = M^{-1}(\tau - V(\theta, \dot{\theta}) - G(\theta)) \quad (20)$$

and the system can be represented as state space form as follows:

$$\dot{x} = F(x) + G(x)u \quad (21)$$

where the state, x , equals $[\theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2]^T$ and the control, u , equals $[\tau_1, \tau_2]^T$. This state space equation is transformed into discrete-time domain and then ILQR and MQEA are employed to find the values of Q_f , Q , and R that minimize multiple performance criteria.

VI. SIMULATION RESULTS

A. Simulation Environment

Generating optimal trajectories of two-link inverted pendulum was used for algorithm verification. Parameters used in ILQR and MQEA are given in Table I, where the constants related to the objective function, K_r , K_s , and K_{po} , were

TABLE I
PARAMETER SETTINGS OF ILQR, MQEA, AND PSSA.

Algorithm	Parameter	Value
ILQR	Sampling time (sec)	0.01
	Final time step (N)	500
	Number of iterations	5
MQEA	Size of global population ($N = n \cdot s$)	100
	Number of generations	50
	Size of subpopulation (n)	20
	Number of subpopulation (s)	5
	Number of multiple observations	10
PSSA	The rotation angle ($\Delta\theta$)	0.2π
	The interaction degree (ξ)	0.75

TABLE II
PAIRWISE COMPARISON MATRIX AND NORMALIZED WEIGHTS
OF PSSA FOR CASE 1.

	f_1	f_2	f_3	Weights
f_1	1.0	0.5	0.1	0.077
f_2	2.0	1.0	0.2	0.154
f_3	10.0	5.0	1.0	0.769

TABLE III
PAIRWISE COMPARISON MATRIX AND NORMALIZED WEIGHTS
OF PSSA FOR CASE 2.

	f_1	f_2	f_3	Weights
f_1	1.0	5.0	10.0	0.769
f_2	0.2	1.0	2.0	0.154
f_3	0.1	0.5	1.0	0.077

set to 1.0. Values used in ILQR, such as the number of iterations and the final time step, were obtained from previous MATLAB test.

For PSSA, Choquet fuzzy integral was used to calculate the value of global evaluation for all nondominated solutions. The value of interaction degree, ξ , was set to 0.75 and two cases for multiple objectives were considered in the simulation: placing more emphasis on decreasing the percentage overshoot ($f_1 : f_2 : f_3 = 1 : 2 : 10$) for Case 1 and on decreasing the rise time ($f_1 : f_2 : f_3 = 10 : 2 : 1$) for Case 2. In order to find the value of partial evaluation for nondominated solutions, values of objective function were normalized to 1. Corresponding pairwise comparison matrices and normalized weights for Case 1 and Case 2 are shown in Table II and Table III, respectively. Since the objective values are needed to be minimized in this problem, minimum and maximum value were mapped to 1 and 0, respectively.

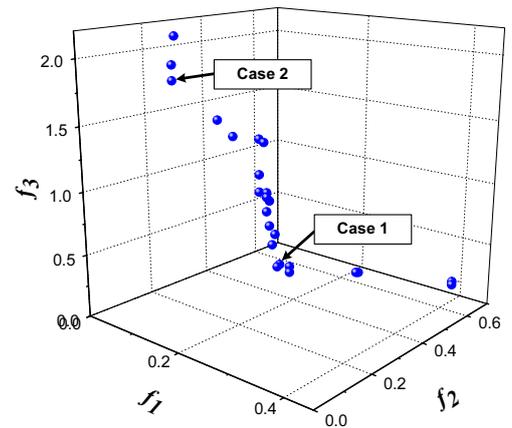


Fig. 2. Nondominated solutions of MQEA in three-objective spaces.

B. Results

Fig. 2 and Fig. 3 show the nondominated solutions obtained using ILQR and MQEA in three-objective spaces and two-objective spaces, respectively. The relationship between rise time (f_1) and settling time (f_2) was roughly linear, but the relationships between rise time and percentage overshoot (f_3), and settling time and percentage overshoot were inversely proportional. In other words, it was impossible to generate an optimal trajectory that minimizes all three performance criteria simultaneously. Note that the maximum and minimum boundary values were given for cost-weighting matrices to enhance simulation efficiency.

Fig. 4 shows the simulation results of two nondominated solutions selected by PSSA among the various nondominated solutions as shown in Fig. 3. The results of first link are represented as a solid line, and second link's as a dotted line. Fig. 4(a)-(b) represent the optimal trajectories of two-link angles and applied torques when percentage overshoot was emphasized (Case 1), whereas Fig. 4(c)-(d) represent results when rise time was emphasized (Case 2).

Table IV shows the objective function values (f_1 , f_2 , and f_3) and the corresponding values of cost-weighting matrices (Q_f , Q , and R) for two simulation cases, where diag represents the diagonal matrix. The rise time was 0.16 seconds, the settling time was 0.34 seconds, and the percentage overshoot was 0.33% for Case 1. However, when the rise time was more optimized in Case 2, it decreased to 0.08 seconds, and the settling time decreased to 0.18 seconds, whereas the percentage overshoot increased to 1.79%. To stabilize the dynamical system in a shorter period of time, maximum applied torques of two-links in Case 2 were higher than those applied in Case 1.

Since the value of λ was -0.89 , the plausibility measure was considered as the fuzzy measure. Due to this plausibility measure in PSSA, the selected solution was the most optimized one for the objective, with the highest degree of consideration, even though it was less optimized for other objectives. In summary, various optimal trajectories of nonlinear dynamical system could be obtained, which depend on the degree of consideration for each objective given by the user's preference.

VII. CONCLUSIONS

This paper proposed a method to generate optimal trajectories of nonlinear dynamical system and to generate multiple nondominated solutions using MOEA in order to satisfy multiple performance criteria. The optimal trajectories of two-link inverted pendulum were optimized by MQEA for three objectives - rise time, settling time, and percentage overshoot. Then, PSSA was employed to select one preferred solution among various nondominated solutions. Simulation results of two-link inverted pendulum confirmed that the proposed scheme could generate optimal trajectories satisfying the user's preference for performance criteria, such as rise time and percentage overshoot.

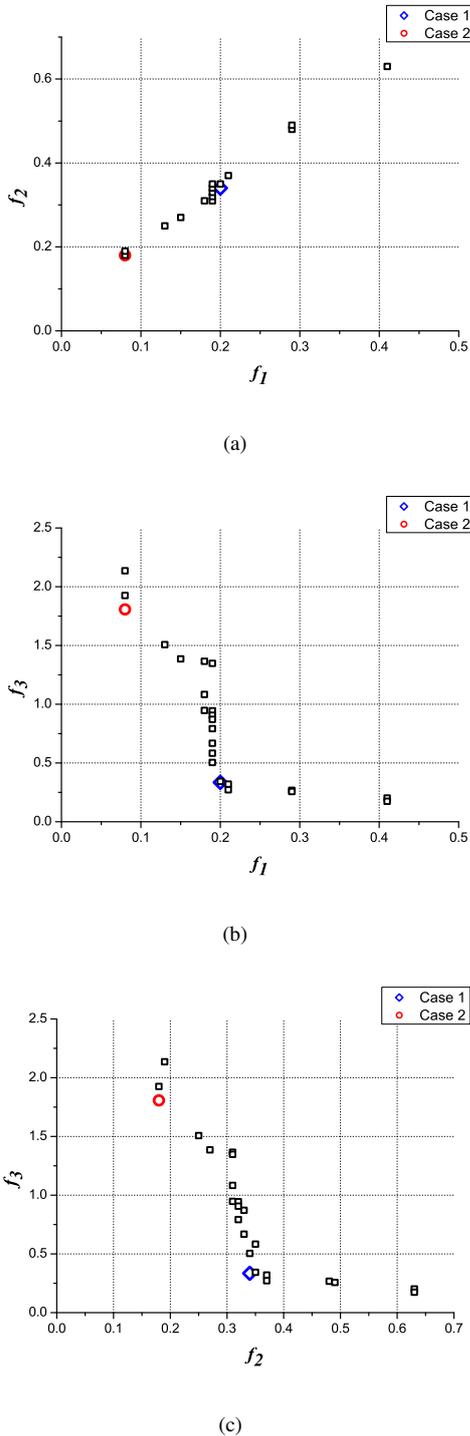


Fig. 3. Nondominated solutions of MQEA in (a) $f_1 - f_2$, (b) $f_1 - f_3$, and (c) $f_2 - f_3$ objective spaces.

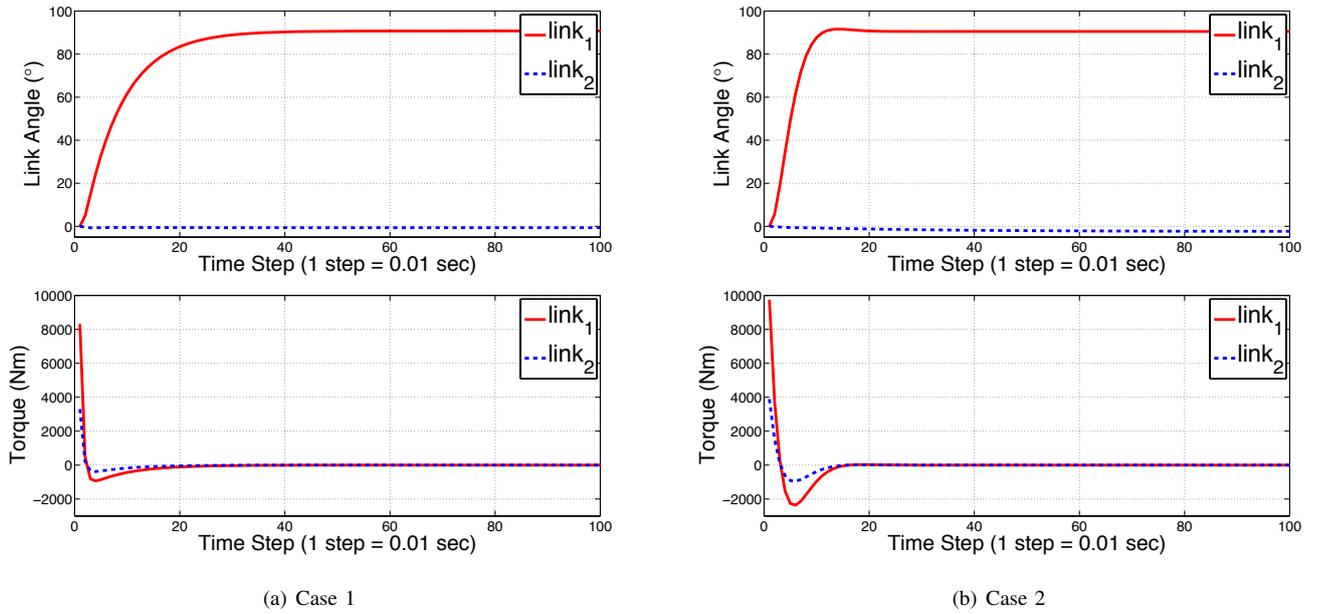


Fig. 4. Link and torque trajectories of two-link inverted pendulum for two simulated cases.

TABLE IV
VALUES OF OBJECTIVE FUNCTION AND COST-WEIGHTING MATRICES.

	f_1	f_2	f_3	Q_f, Q	R
Case 1	0.16	0.34	0.33	diag(230.34, 1.21, 288.74, 1.38)	diag(0.000001, 0.000001)
Case 2	0.08	0.18	1.79	diag(329.78, 0.13, 173.56, 12.34)	diag(0.000001, 0.000021)

ACKNOWLEDGMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2010-0000831).

REFERENCES

- [1] C. G. Atkeson, "Using local trajectory optimizers to speed up global optimization in dynamic programming," *Advances in Neural Information Processing Systems*, pp. 663–670, 1994.
- [2] C. G. Atkeson and J. Morimoto, "Non-parametric representation of a policies and value functions: A trajectory based approach," *Advances in Neural Information Processing Systems 15*, 2003.
- [3] W. Li and E. Todorov, "Iterative linear quadratic regulator design for nonlinear biological movement systems," in *Proc. of Int. Conf. on Informatics in Control, Automation and Robotics*, Setúbal, Portugal, Aug. 2004, pp. 1–8.
- [4] E. Zitzler, and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.
- [5] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the performance of the strength pareto evolutionary algorithm," *Technical Report 103, Computer Engineering and Communication Networks Lab, Swiss Federal Institute of Technology, Zurich*, 2001.
- [6] N. Srinivas and K. Deb, "Multiobjective function optimization using nondominated sorting genetic algorithms," *Evolutionary Computation*, vol. 2, no. 3, pp. 221–248, 1995.
- [7] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [8] Y.-H. Kim, J.-H. Kim and K.-H Han, "Quantum-inspired Multiobjective Evolutionary Algorithm for Multiobjective 0/1 Knapsack Problems, in *Proc. of IEEE Congress on Evolutionary Computation*, Vancouver, Canada, Jul. 2006, pp. 9151–9156.
- [9] Y.-H. Kim and J.-H. Kim, "Multiobjective quantum-inspired evolutionary algorithm for fuzzy path planning of mobile robot," in *Proc. of IEEE Congress on Evolutionary Computation*, Trondheim, Norway, May. 2009, pp. 1185–1192.
- [10] K.-H. Han and J.-H Kim, "Quantum-inspired evolutionary algorithm for a class of combinatorial optimization, *IEEE Transactions on Evolutionary Computation*, vol. 6, pp. 580–593, 2002.
- [11] K.-H. Han and J.-H. Kim, "Quantum-inspired evolutionary algorithms with a new termination criterion, He gate, and two phase scheme," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 2, 2004.
- [12] K.-H. Han and J.-H. Kim, "On the analysis of the quantum-inspired evolutionary algorithm with a single individual," in *Proc. of IEEE Congress on Evolutionary Computation*, vol. 2, Vancouver, Canada, Jul. 2006, pp. 9172–9179.
- [13] T. Hey, Quantum computing: An introduction, *Computing and Control Engineering Journal. Piscataway, NJ: IEEE Press*, vol. 10, no. 3, pp. 105–112, Jun. 1999.
- [14] M. Sugeno, "Theory of fuzzy integrals and its applications," *Doct.Thesis*, Tokyo Institute of Technology, 1974.
- [15] E. Takahagi, "On identification methods of λ -fuzzy measures using weights and λ ," *Journal of Japan Society for Fuzzy Theory and Systems*, vol. 12, no. 5, pp. 665–676, 2000.
- [16] T. L. Saaty, "Decision making with the analytic hierarchy process," *International Journal of Services Sciences*, vol. 1, no. 1, pp. 83–98, 2008.