DMQEA: Dual Multiobjective Quantum-inspired Evolutionary Algorithm

Si-Jung Ryu[1], Jong-Hwan Kim[1] and Ki-Baek Lee[2]
[1]*Department of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon 305-701, Korea*
[2]*Department of Electric Engineering, Kwangwoon University, Seoul 139-701, Korea*
{johkim, sjryu}@rit.kaist.ac.kr, kblee@kw.ac.kr

Keywords: Multiobjective evolutionary algorithm, quantum-inspired evolutionary algorithm, preference-based evolutionary algorithm

Abstract: This paper proposes dual multiobjective quantum-inspired evolutionary algorithm (DMQEA) with the dual-stage of dominance check by introducing secondary objectives in addition to primary objectives. The secondary objectives are to maximize global evaluation values and crowding distances of the solutions in the external global population obtained for the primary objectives and the previous archive obtained from the secondary objectives-based nondominated sorting. By employing the secondary objectives for sorting the solutions in each generation, DMQEA can induce the balanced exploration of the solutions in terms of user's preference and diversity to generate preferable and diverse nondominated solutions in the archive. To demonstrate the effectiveness of the proposed DMQEA, empirical comparisons with MQEA, MQEA-PS, and NSGA-II are carried out for benchmark functions.

1 INTRODUCTION

Multiobjective evolutionary algorithms (MOEAs) are designed to solve multiobjective optimization problems to get Pareto-optimal solutions while maintaining as diverse a distribution as possible. These are well-known two goals, proximity to Pareto-optimal front and diversity preservation, in ideal multiobjective optimization. Much research has been conducted to enhance the solution quality and diversity (Laumanns et al., 2002; Cui et al., 2001; Bosman and Thierens, 2003; Kim et al., 2009; Deb et al., 2002; Lee and Kim, 2012).

The other issue is how to select a preferable solution among the widely distributed solutions in the Pareto-optimal front for the application of the real world problem. To solve this issue, preference-based solution selection algorithm (PSSA) was proposed (Kim et al., 2012). It selects a solution considering user's preference for each objective, which is represented by the fuzzy measures. In PSSA, global evaluation value of a candidate solution is calculated by the fuzzy integral of the partial evaluation values with respect to the fuzzy measures. The solution with the highest global evaluation value is selected out of the candidate solutions.

Based on PSSA, multiobjective quantum-inspired evolutionary algorithm with preference-based selection (MQEA-PS) was proposed (Kim et al., 2012). In each archive generation process, MQEA-PS employs PSSA in MQEA for preference-based sorting for the solutions in the external global population and the previous archive. It means that the nondominated solutions in the archive are obtained by preference-based sorting instead of dominance-based sorting, whereas the internal subpopulations are sorted by fast nondominated sorting. In this way, the solutions that reflect user's preference for each objective can be obtained in the archive. Furthermore, for considering the diversity of the solutions as well as user's preference, crowding distance sorting after the preference-based sorting in the archive generation process is developed (Ryu et al., 2012). However, the solutions are lack of the proximity to the Pareto front.

In this paper, we propose dual multiobjective quantum-inspired evolutionary algorithm (DMQEA) by introducing secondary objectives in addition to primary objectives that are given objectives in the problem. The proposed DMQEA has the dual-stage of dominance check respectively for the primary and secondary objectives. In the first stage, the dominated solutions with respect to primary objectives are culled out by primary objectives-based nondominated sorting (PONS). In the second stage, nondominated sorting is applied for the secondary objectives for the generation of archive, which is called secondary objectives-based nondominated sorting (SONS). The secondary objectives are to maximize global evaluation values and crowding distances of the solutions in the previous archive and the external global population obtained for the primary objectives. The archive consists of first-tier solutions obtained from the SONS.

By employing SONS in each generation, DMQEA can induce the balanced exploration of the solutions in terms of user's preference and diversity to produce preferable and diverse nondominated solutions in the archive. The effectiveness of the proposed DMQEA is demonstrated through statistical comparisons with MQEA, MQEA-PS, and NSGA-II for benchmark functions. The experimental results confirm that the proposed DMQEA generates the solutions with larger hypervolume while maintaining user's preference compared to the existing two algorithms, MQEA and MQEA-PS.

The rest of this paper is organized as follows: quantum-inspired evolutionary algorithm (QEA), preference-based solution selection algorithm (PSSA), and crowding distance are briefly described in Section II. Section III proposes dual multiobjective evolutionary algorithm (DMQEA). The experimental results are presented in Section IV and concluding remarks follow in Section V.

2 PRELIMINARIES

2.1 QEA

Quantum-inspired evolutionary algorithm (QEA) is an evolutionary algorithm, which employs the probabilistic mechanism inspired by the concept and principles of quantum computing, such as a quantum bit and superposition of states (Han and Kim, 2002; Han and Kim, 2004). Building block of classical digital computer is represented by two binary states, '0' or '1', which is a finite set of discrete and stable state. In contrast, QEA utilizes a novel representation, called a Q-bit representation, for the probabilistic representation that is based on the concept of qubits in quantum computing (Hey, 1999). Quantum system enables the superposition of such state as follows:

$$\alpha|0\rangle + \beta|1\rangle \quad (1)$$

where α and β are the complex numbers satisfying $|\alpha|^2 + |\beta|^2 = 1$.

Qubit is shown in Fig. 1, which can be illustrated as a unit vector on the two dimensional space as follows:

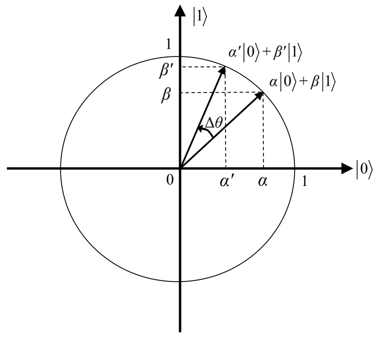

Figure 1: Qubit described in two-dimensional space.

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad (2)$$

where $|\alpha|^2 + |\beta|^2 = 1$. Q-bit individual is defined as a string of Q-bits as follows:

$$\mathbf{q}_j^t = \left[\begin{array}{c|c|c|c} \alpha_{j,m-1}^t & \alpha_{j,m-2}^t & \cdots & \alpha_{j,0}^t \\ \beta_{j,m-1}^t & \beta_{j,m-2}^t & \cdots & \beta_{j,0}^t \end{array} \right] \quad (3)$$

where m is the string length of Q-bit individual, and $j = 1, 2, ..., n$ for the population size n. The population of Q-bit individuals at generation t is represented as follows:

$$Q(t) = \{\mathbf{q}_1^t, \mathbf{q}_2^t, \cdots, \mathbf{q}_n^t\}. \quad (4)$$

Since Q-bit individual represents the linear superposition of all possible states probabilistically, various individuals are generated during the evolutionary process. The procedure of QEA and the overall structure for single-objective optimization problems are described in (Han and Kim, 2002). To solve multiobjective optimization problems, multiobjective quantum-inspired evolutionary algorithm (MQEA) is also developed (Kim et al., 2006).

2.2 PSSA

Preference-based solution selection algorithm (PSSA) selects a solution among the obtained nondominated solutions considering user's preference (Kim et al., 2012). The nondominated solutions cannot be directly compared against each other, and therefore a multicriteria decision making (MCDM) algorithm is required to evaluate them. In PSSA, the global evaluation value of a candidate solution is calculated by the fuzzy integral, as an MCDM algorithm, of the partial evaluation values with respect to the fuzzy measures. The fuzzy measures represent the degrees of consideration for objectives, and the partial evaluation value indicates a normalized objective function value. Overall procedure of global evaluation is summarized in Algorithm 1. Each step in the algorithm is briefly described in the following and detail procedure of global evaluation is described in (Kim et al., 2012).

1. Calculate λ-fuzzy measures

Objectives are defined as criteria in multi-objective problem. λ-fuzzy measure represents the degree of consideration for each criterion. To get the values of λ-fuzzy measures, a pairwise comparison matrix (P) is initially defined by user for representing preference degrees between criteria. Secondly, the normalized weights of P are calculated by adding each value in the row of the pairwise comparison matrix and dividing it by the total sum of the values in the row. Lastly, λ-fuzzy measures are obtained using the normalized weights (Bajwa et al., 2008).

2. Compute global evaluation value

First, the value of partial evaluation of each solution is calculated by normalizing the objective function value to 1. Global evaluation of each and every solution is performed by the Choquet fuzzy integral of the partial evaluation values with respect to the λ-fuzzy measures, which are obtained from the previous steps.

Algorithm 1 Procedure of global evaluation

- l: No. of the solutions
- m: No. of the objectives
- C: A set of objectives $C = \{c_1, c_2, ..., c_m\}$
- P: A power set of C
- $f_j(\mathbf{x}_k)$: j-th Objective value of $\mathbf{x}_k$
- $h_j(\mathbf{x}_k)$: j-th partial evaluation value of $\mathbf{x}_k$
- $e(\mathbf{x}_k)$: Global evaluation value of $\mathbf{x}_k$

1. Calculate λ-fuzzy measures g's of $P(C)$
   - Make a pairwise comparison matrix P
   - Calculate normalized weights of
   - Calculate λ-fuzzy measures of $P(C)$.
2. Compute global evaluation value e

   for $k = 1$ to l **do**
       for $j = 1$ to m **do**
           $h_j(\mathbf{x}_k)$ = Normalize($f_j(\mathbf{x}_k)$)
       end for
   end for
   for $k = 1$ to l **do**
       $e(\mathbf{x}_k) = \int h \circ g$
   end for

2.3 Crowding Distance

The crowding distance estimates the density of solutions surrounding a particular solution in the population (Deb et al., 2002). The crowding distance is aimed to uniformly select the solutions in the front, making the solutions in the most dense areas less likely to be selected. The crowding distance is defined by the average distance of the closest points on either side of the point for each objective. Therefore, the crowding distance is inversely proportional to the density of solutions. Boundary points for each objective have the maximum crowding distance, and they are always selected. Calculation of crowding distance is described in Algorithm 2.

Algorithm 2 Crowding distance assignment

- l: No. of the solutions
- m: No. of the objectives
- $f_j(\mathbf{x}_k)$: j-th objective value of $\mathbf{x}_k$
- $\mathbf{x}_k.CD$: Crowding distance of the solution $\mathbf{x}_k$

1. Initialization

   for $k = 1$ to l **do**
       $\mathbf{x}_k.CD = 0$
   end for
2. Calculate the crowding distances

   for $j = 1$ to m **do**
       for $k = 1$ to l **do**
           Calculate the objective value $f_j(\mathbf{x}_k)$
       end for
       Sort the solutions using objective value $f_j(\mathbf{x}_k)$, $\mathbf{x}_k = \text{sort}(\mathbf{x}_k)$
       $\mathbf{x}_1.CD = \mathbf{x}_l.CD = \infty$
       for $k = 2$ to $l - 1$ **do**
           $\mathbf{x}_k.CD = \mathbf{x}_k.CD + |f_j(\mathbf{x}_{k+1}) - f_j(\mathbf{x}_{k-1})|$
       end for
   end for

3 DMQEA

Dual multiobjective quantum-inspired evolutionary algorithm (DMQEA) has the dual-stage of dominance check for the primary and secondary objectives. Primary objectives are the given objectives of the problem. The secondary objectives are to maximize both the global evaluation values and crowding distances of the solutions in the external global population obtained for the primary objectives and the

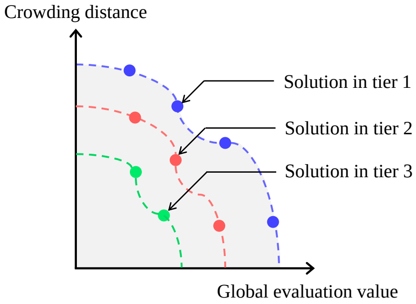

Figure 2: Secondary objectives-based nondominated sorting

previous archive. In each archive generation process, the secondary objectives are employed for sorting the solutions, which is called secondary objectives-based nondominated sorting (SONS). By the proposed SONS, the archive stores first-tier solutions.

3.1 SONS

SONS is to sort the solutions with the secondary objectives for maximizing the global evaluation value (GEval) and crowding distance (CD). The SONS is performed for the solutions in the external global population obtained for the primary objectives and the previous archive. It means that in DMQEA, the solutions are sorted by SONS that checks the dominance relationship with respect to GEval and CD. By SONS, the solutions that are not dominated by any other solutions could be obtained as first-tier solutions that are stored in the archive.

The proposed SONS is depicted in Fig. 2. GEval and CD of every solution in the external global population and the previous archive are calculated as explained in the previous section. Note that the global evaluation value of a solution is calculated by the fuzzy integral of the partial evaluation values with respect to the fuzzy measures representing the user's preference for objectives. The solutions with higher values of GEval and CD are better in terms of user's preference and diversity. For example, in the figure, blue points are classified as first-tier solutions to be stored in the archive. The solutions in lower tiers are discarded because they are dominated by the first-tier solutions.

3.2 Procedure of DMQEA

In an archive generation process, MQEA employs dominance-based sorting for primary objectives of the solutions in the external global population and the previous archive. Most of them are nondominated by the other solutions because primary objectives-based nondominated sorting (PONS) or fast nondominated sorting is already performed in each subpopulation. It means that the dominance-based sorting for the primary objectives might be an ineffective operation in selecting solutions to be stored in the archive. Indeed, in experiments, the external global population almost consists of nondominated solutions. To solve this problem, DMQEA employs SONS in the archive generation process. By SONS, each solution is classified into the corresponding tier and the solutions in the first tier are stored in the archive. These are used for reference solutions through the global random migration process. The overall procedure of DMQEA is summarized in Algorithm 3, and depicted in Fig. 3. Each step is described in detail in the following.

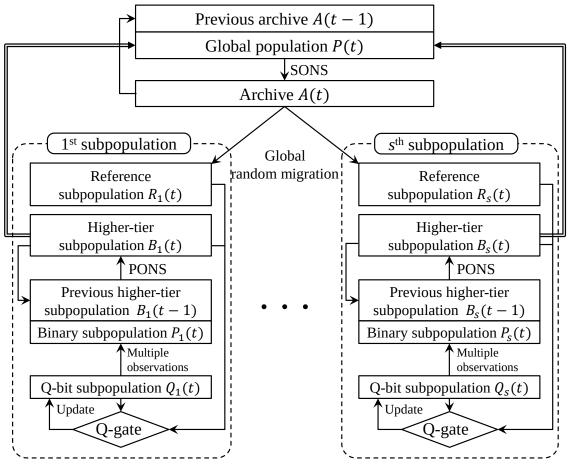

Figure 3: Overall procedure of DMQEA, where PONS: Primary objectives-based nondominated sorting, SONS: Secondary objectives-based nondominated sorting

1. Initialize $Q_k(t)$ and generate archive $A(t)$
$Q_k(0)$ including $\mathbf{q}_j^0$, which consists of α_{ji}^0 and β_{ji}^0, is initialized with $1/\sqrt{2}$, where $i = 0,1,\ldots,m-1$, $j = 1,2,\ldots,n$, and $k = 1,2,\ldots,s$. Note that m is the string length of Q-bit individual, n is the subpopulation size, and s is the number of subpopulations. It means that one Q-bit individual, $\mathbf{q}_j^0$, represents the linear superposition of all possible states with same probability. Binary solutions in $P_k(0)$ are produced by multiple observing the states of $Q_k(0)$, where $P_k(0) = \{\mathbf{x}_1^0, \mathbf{x}_2^0, \ldots, \mathbf{x}_n^0\}$ and $\mathbf{x}_j^0 = \{x_{j,m-1}^0, x_{j,m-2}^0, \ldots, x_{j0}^0\}$, $j = 1,2,\ldots,n$. A bit of one binary solution, x_{ji}^0, has a value either '0' or '1' according to the probability either $|\alpha_{ji}^0|^2$ or $|\beta_{ji}^0|^2$, where $i = 0,1,\ldots,m-1$, $j = 1,2,\ldots,n$, as follows:

$$x_{ji}^0 = \begin{cases} 0 & \text{if } \mathrm{rand}[0,1] \geq |\beta_{ji}^0|^2 \\ 1 & \text{if } \mathrm{rand}[0,1] < |\beta_{ji}^0|^2. \end{cases} \quad (5)$$

Algorithm 3 Procedure of DMQEA

- $P_k(t) = \{\mathbf{x}_1^t, \mathbf{x}_2^t, ..., \mathbf{x}_n^t\}$
- $\mathbf{x}_j^t = \{x_{j,m-1}^t, x_{j,m-2}^t, ..., x_{j0}^t\}$
- $Q_k(t) = \{\mathbf{q}_1^t, \mathbf{q}_2^t, \cdots, \mathbf{q}_n^t\}$
- $\mathbf{q}_j^t = \left[\begin{array}{c|c|c|c} \alpha_{j,m-1}^t & \alpha_{j,m-2}^t & \cdots & \alpha_{j,0}^t \\ \beta_{j,m-1}^t & \beta_{j,m-2}^t & \cdots & \beta_{j,0}^t \end{array} \right]$
- $R_k(t) = \{\mathbf{r}_1^t, \mathbf{r}_2^t, ..., \mathbf{r}_n^t\}$
- s = No. of subpopulations
- n = Size of subpopulation
- m = Q-bit string length

1. Initialize $Q_k(t)$ and generate archive $A(t)$

```
t = 0
for k = 1 to s do
    for j = 1 to n do
        for i = 0 to m − 1 do
            α^t_ji = β^t_ji = 1/√2
        end for
        Make P_k(t) by multiple observing the states of Q_k(t)
        for each objective do
            Evaluate the objective value from x^t_j
        end for
        Copy all solutions in P_k(t) into P(t)
        Store first-tier solutions of P(t) by SONS in the archive A(t)
    end for
end for
```

2. Generate global population $P(t)$

```
t = t + 1
for k = 1 to s do
    for j = 1 to n do
        Make P_k(t) by multiple observing the states of Q_k(t)
        for each objective do
            Evaluate the objective value from x^t_j
        end for
    end for
    Run PONS for P_k(t) ∪ B_k(t − 1)
    Store n higher-tier solutions of P_k(t) ∪ B_k(t − 1) into B_k(t)
end for
Store all solutions in every B_k(t) into P(t)
```

3. Update archive $A(t)$

```
for each solution in A(t − 1) ∪ P(t) do
    Evaluate GEval and CD
end for
Run SONS
Store the first-tier solutions into the archive A(t)
```

4. Migrate and update $Q_k(t)$

```
for k = 1 to s do
    for j = 1 to n do
        Select a solution in A(t) randomly
        Store it into r^t_j
        Update q^t_j using Q-gates referring to the solutions in r^t_j
    end for
end for
```

5. Go back to Step 2 and repeat

Multiple observation is performed on each and every Q-bit individual in subpopulations, $\mathbf{q}_j^0$ in $Q_k(0)$, $k = 1, 2, ..., s$. Each binary solution in $P_k(0)$ is decoded to a real number if necessary, and its objective value is calculated. All solutions in each binary subpopulation $P_k(0)$ are copied to the external global population $P(0)$ and store first tier solutions of $P(0)$ by SONS in the archive $A(t)$.

2. Generate global population $P(t)$

Binary solutions are generated by multiple observations of Q-bit individuals in Q-bit subpopulation $Q_k(t)$. Each bit of binary solution x_{jl}^t, $l = 1, 2, ..., o$, where o is the observation index is obtained. $\mathbf{x}_j^t$ is assigned by the best among the observed binary solutions x_{jl}^t, $l = 1, 2, ..., o$, from the multiple observations. And then, evaluation is performed to $P_k(t)$, where $k = 1, 2, ..., s$. Therefore, objective values of all solutions in each subpopulation are obtained. The solutions in the previous higher-tier subpopulation and the current binary subpopulation $P_k(t) \cup B_k(t-1)$ are sorted by PONS to select n solutions in order from the first tier to the lower tiers. The n higher-tier solutions form $B_k(t)$, where $B_k(t) = \{\mathbf{b}_1^t, \mathbf{b}_2^t, ..., \mathbf{b}_n^t\}$ that is to become the previous higher-tier subpopulation in the next generation. To update Q-bit individuals corresponding to higher-tier subpopulation later, Q-bit subpopulation $Q_k(t)$ is rearranged by replacing each $\mathbf{q}_j^t$ in the subpopulation by the Q-bit individual that has generated $\mathbf{b}_j^t$. All higher-tier solutions in each subpopulation $B_k(t)$ are copied to the external global population $P(t)$.

3. Update archive $A(t)$

Global evaluation values are calculated by the fuzzy integral and crowding distance is also calculated. The fuzzy integral reflects how much a user prefers the solution, and crowding distance denotes the density of the solutions. SONS with GEval and CD for the solutions in the external global population and the previous archive is performed. The nondominated solutions in the first tier are stored into the archive $A(t)$. The size of the archive might be different each generation.

4. Migrate and update $Q_k(t)$

The solutions in the archive $A(t)$ are randomly selected n times and they are globally migrated to each reference subpopulation $R_k(t)$, where $R_k(t) = \{\mathbf{r}_1^t, \mathbf{r}_2^t, ..., \mathbf{r}_n^t\}$. Note that the solutions in $R_k(t)$ are employed as references to update Q-bit individuals, each of which is corresponding to the solution in the higher-tier subpopulation. Global random migration procedure occurs at every generation. In the update process of Q-bit individuals, the rotation gate is employed. $\mathbf{r}_j^t$ and $\mathbf{b}_j^t$ in each subpopulation are compared bit-by-bit to decide the update directions of Q-bit individuals in the rotation gate $U(\Delta\theta)$, which is defined as follows:

$$\mathbf{q_j^t} = U(\Delta\theta) \cdot \mathbf{q_j^{t-1}} \tag{6}$$

with

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

where $\Delta\theta$ is the rotation angle of each Q-bit as shown in Fig. 1. Note that crossover and mutation operators are not used in QEA.

5. Go back to Step 2 and repeat

Go back to Step 2 and repeat until a termination condition is satisfied.

4 EXPERIMENTAL RESULTS

4.1 Experimental Settings

The proposed DMQEA was compared with MQEA, MQEA-PS, and NSGA-II. To evaluate the performance of algorithms, we employed seven DTLZ functions as benchmark functions. The number of variables for each DTLZ function was set to 9 for DTLZ1, 16 for DTLZ2 to DTLZ6, and 26 for DTLZ7. Parameters for DMQEA, MQEA-PS, MQEA, and NSGA-II were equally set and given in Table 1. Belief measure ($\xi = 0.25$) for MQEA-PS and DMQEA was used. As the preferred objectives, two objectives among the five objectives in DTLZ functions were selected. The preference degrees or the degrees of consideration for five objectives was set as $f_1 : f_2 : f_3 : f_4 : f_5 = 1 : 10 : 1 : 10 : 1$. The normalized weights from the pairwise comparison matrix were calculated as (0.0435, 0.435, 0.0435, 0.435, 0.0435).

Table 1: Parameter setting of MQEA, MQEA-PS, and DMQEA for DTLZ functions

Algorithms	Parameters	Values
MQEA, MQEA-PS, DMQEA	The population size ($N = n \cdot s$)	100
	No. of generations	3000
	Subpopulation size(n)	25
	No. of subpopulations (s)	4
	No. of multiple observations	10
	The rotation angle ($\Delta\theta$)	0.23π
NSGA-II	The population size (N)	100
	No. of generations	3000
	Mutation probability	0.1

4.2 Performance Metrics

Two performance metrics, the size of dominated space and the diversity, were employed to evaluate the performances of MQEA, MQEA-PS, DMQEA, and NSGA-II (Zitzler, 1999). The size of dominated space, $\vec{\mathcal{S}}$, is defined by the hypervolume of the finally obtained global population. The quality of the obtained global population is high if this space is large. The diversity, $\vec{\mathcal{D}}$, is to evaluate the spread of nondominated solutions, which is defined as follows (Li et al., 2004):

$$\vec{\mathcal{D}} = \frac{\sum_{k=1}^{n}(f_k^{(max)} - f_k^{(min)})}{\sqrt{\frac{1}{|N_0|}\sum_{i=1}^{|N_0|}(d_i - \bar{d})^2}} \tag{7}$$

where N_0 is the set of nondominated solutions, d_i is the minimal distance between the i-th solution and the nearest neighbor, and $\bar{d}$ is the mean value of all d_i. $f_k^{(max)}$ and $f_k^{(min)}$ represent the maximum and minimum objective function values of the k-th objective, respectively. A larger value means a better diversity of the nondominated solutions.

4.3 Results

The proposed DMQEA generated the optimized solutions concentrated on the selected preferred objectives, f_2 and f_4. The hypervolume and diversity of MQEA, MQEA-PS, DMQEA, and NSGA-II are summarized in Tables 2. The results in Table 2 are averaged ones by repeating the simulation 50 times.

For statistical analysis, t-test was employed to statistically compare the performance metrics of algorithms. The t-test is a statistical hypothesis test in which the test statistic follows a t distribution if the null hypothesis $\mathcal{H}_0$ is supported. If the null hypothesis $\mathcal{H}_0$ is rejected, the alternative hypothesis is supported. t-test was used to determine whether two comparison groups were significantly different from each other. The t-test was carried out with the two-tailed test. Tables 3 show t-value (and p-value) for the hypervolume with MQEA, MQEA-PS, and NSGA-II.

Table 2: Comparisons of hypervolume and diversity among MQEA, MQEA-PS, and DMQEA for seven DTLZ functions

[Average hypervolume of nondominated solutions]

Problem	NSGA-II	MQEA	MQEA-PS	DMQEA
DTLZ1	99999	99255	99740	99748
DTLZ2	99998	99796	97139	99202
DTLZ3	66773	N/A	71911	79015
DTLZ4	99999	95119	92309	94898
DTLZ5	99158	98578	95877	98388
DTLZ6	96169	67967	89411	72915
DTLZ7	63202	10907	38574	40500

[Average diversity of nondominated solutions]

Problem	NSGA-II	MQEA	MQEA-PS	DMQEA
DTLZ1	103.95	144.99	92.13	75.99
DTLZ2	132.95	71.05	93.96	79.19
DTLZ3	104.36	55.05	60.93	52.73
DTLZ4	132.53	80.62	105.24	137.61
DTLZ5	179.29	129.39	278.32	139.29
DTLZ6	143.12	73.35	65.26	63.46
DTLZ7	164.78	115.96	136.27	135.55

As shown in Table 3, the proposed DMQEA had larger hypervolume than MQEA for DTLZ1, DTLZ3, DTLZ6, and DTLZ7. For the fair comparison of hypervolume, the size of the obtained global populations for three algorithms are set to the same value. In comparison with MQEA-PS, DMQEA had larger hypervolume for all DTLZ functions except for DTLZ1 and DTLZ6. It means DMQEA found more optimized solutions close to the Pareto-optimal front. However, in comparison with NSGA-II, DMQEA had better performance only for DTLZ3. This is because the proposed DMQEA generated the optimized solutions concentrated on the selected preferred objectives, f_2 and f_4. Due to the property of the hypervolume, DMQEA that generates the dense solutions in a small region has a lower value of hypervolume compared to NSGA-II for DTLZ1, DTLZ2, DTLZ5.

Table 4 shows the result for the diversity and Table 5 presents average objective values of preferred solutions finally selected by PSSA among the solutions obtained from MQEA-PS and DMQEA, respectively. As Table 4 shows, DMQEA has a lower value of diversity compared with MQEA, MQEA-PS, and NSGA-II, and Table 5 shows DMQEA generated the solutions that effectively reflect preferred objectives. It means that DMQEA could generate the solutions emphasized on preference objectives, f_2, f_4 in dense area. In other word, the proposed DMQEA could find more optimized solutions for the preferred objectives compared with the other algorithms.

Table 3: The hypothesis test on $\mathcal{S}$ of the three algorithms

	$\mathcal{H}_0 : \overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{MQEA}} = 0$		
	t-value (p-value)	Reject	$\mathcal{H}_1$
DTLZ1	6.731 (0.000)	NO	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{MQEA}} > 0$
DTLZ2	-14.524 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{MQEA}} < 0$
DTLZ3	67.767 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{MQEA}} > 0$
DTLZ4	-0.557 (0.580)	NO	N/A
DTLZ5	-7.233 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{MQEA}} < 0$
DTLZ6	8.871 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{MQEA}} > 0$
DTLZ7	57.875 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{MQEA}} > 0$

	$\mathcal{H}_0 : \overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{MQEA-PS}} = 0$		
	t-value (p-value)	Reject	$\mathcal{H}_1$
DTLZ1	0.252 (0.802)	NO	N/A
DTLZ2	15.134 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{MQEA-PS}} > 0$
DTLZ3	3.367 (0.001)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{MQEA-PS}} > 0$
DTLZ4	6.287 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{MQEA-PS}} > 0$
DTLZ5	12.742 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{MQEA-PS}} > 0$
DTLZ6	-37.855 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{MQEA-PS}} < 0$
DTLZ7	6.709 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{MQEA-PS}} > 0$

	$\mathcal{H}_0 : \overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{NSGA-II}} = 0$		
	t-value (p-value)	Reject	$\mathcal{H}_1$
DTLZ1	-10.460 (0.800)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{NSGA-II}} < 0$
DTLZ2	-17.689 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{NSGA-II}} < 0$
DTLZ3	4.771 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{NSGA-II}} > 0$
DTLZ4	-25.450 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{NSGA-II}} < 0$
DTLZ5	-31.727 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{NSGA-II}} < 0$
DTLZ6	-160.96 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{NSGA-II}} < 0$
DTLZ7	-136.51 (0.000)	YES	$\overline{\mathcal{S}}_{\text{DMQEA}} - \overline{\mathcal{S}}_{\text{NSGA-II}} < 0$

Table 4: The hypothesis test on $\mathcal{D}$ of the three algorithms

	$\mathcal{H}_0 : \overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{MQEA}} = 0$		
	t-value (p-value)	Reject	$\mathcal{H}_1$
DTLZ1	-5.966 (0.000)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{MQEA}} < 0$
DTLZ2	9.492 (0.000)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{MQEA}} > 0$
DTLZ3	-0.445 (0.658)	NO	N/A
DTLZ4	3.794 (0.000)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{MQEA}} > 0$
DTLZ5	3.328 (0.002)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{MQEA}} > 0$
DTLZ6	-10.177 (0.000)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{MQEA}} < 0$
DTLZ7	3.439 (0.001)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{MQEA}} > 0$

	$\mathcal{H}_0 : \overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{MQEA-PS}} = 0$		
	t-value (p-value)	Reject	$\mathcal{H}_1$
DTLZ1	-2.959 (0.004)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{MQEA-PS}} < 0$
DTLZ2	-11.512 (0.000)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{MQEA-PS}} < 0$
DTLZ3	-2.002 (0.051)	NO	N/A
DTLZ4	1.997 (0.051)	NO	N/A
DTLZ5	-9.052 (0.000)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{MQEA-PS}} < 0$
DTLZ6	-0.644 (0.522)	NO	N/A
DTLZ7	-0.119 (0.906)	NO	N/A

	$\mathcal{H}_0 : \overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{NSGA-II}} = 0$		
	t-value (p-value)	Reject	$\mathcal{H}_1$
DTLZ1	-3.871 (0.000)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{NSGA-II}} < 0$
DTLZ2	-63.8581 (0.000)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{NSGA-II}} < 0$
DTLZ3	1.773 (0.082)	NO	N/A
DTLZ4	4.527 (0.000)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{NSGA-II}} < 0$
DTLZ5	-4.076 (0.000)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{NSGA-II}} < 0$
DTLZ6	-65.703 (0.000)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{NSGA-II}} < 0$
DTLZ7	-6.241 (0.000)	YES	$\overline{\mathcal{D}}_{\text{DMQEA}} - \overline{\mathcal{D}}_{\text{NSGA-II}} < 0$

5 CONCLUSION

In this paper, dual multiobjective quantum-inspired evolutionary algorithm (DMQEA) was proposed by introducing secondary objectives in addition to primary objectives. The proposed DMQEA had the dual-stage of dominance check for the primary and secondary objectives. The secondary objectives are to maximize global evaluation values and crowding distances of the solutions. The global evaluation of a solution was carried out by the fuzzy in-

Table 5: The objective function values of preferred solutions each selected by PSSA among the solutions obtained from MQEA-PS and DMQEA, respectively

[Average objective values of a preferred solution finally selected by PSSA among the solutions obtained from MQEA-PS]

	f_1	f_2	f_3	f_4	f_5
DTLZ1	0.0599	0.0002	0.1402	0.0002	0.4596
DTLZ2	0.0000	0.0000	0.0000	0.0000	1.0000
DTLZ3	0.0021	0.0002	0.0027	0.1232	5.5938
DTLZ4	1.0000	0.0000	0.0000	0.0000	1.0000
DTLZ5	0.0000	0.0000	0.0000	0.0000	1.0000
DTLZ6	0.0002	0.0002	0.0002	0.0003	1.8851
DTLZ7	0.6302	0.0022	0.6022	0.0005	9.2351

[Average objective values of a preferred solution finally selected by PSSA among the solutions obtained from DMQEA]

	f_1	f_2	f_3	f_4	f_5
DTLZ1	0.0685	0.0005	0.1401	0.0005	0.5994
DTLZ2	0.0000	0.0000	0.0000	0.0000	1.0003
DTLZ3	0.0000	0.0000	0.0000	0.0000	4.4475
DTLZ4	1.0251	0.0000	0.0000	0.0000	1.0000
DTLZ5	0.0000	0.0000	0.0000	0.0000	1.0001
DTLZ6	0.0004	0.0002	0.0004	0.0006	4.1782
DTLZ7	0.1235	0.0017	0.3517	0.0026	10.7801

tegral of the partial evaluation values with respect to the fuzzy measures representing user's preference for objectives. By employing the secondary objectives-based nondominated sorting in each archive generation process, DMQEA could generate the preferable and diverse solutions. For the performance comparisons among MQEA, MQEA-PS, DMQEA, and NSGA-II, seven DTLZ functions were used as benchmark functions, and hypervolume and diversity were employed as performance metrics. The experimental results confirmed that the proposed DMQEA was able to generate more optimized solutions for the preferred objectives compared with the other algorithms.

ACKNOWLEDGEMENTS

This work was supported by the Technology Innovation Program, 10045252, Development of robot task intelligence technology, funded by the Ministry of Trade, Industry Energy (MOTIE, Korea).

REFERENCES

Bajwa, G., Choo, E., and Wedley, W. (2008). Effectiveness analysis of deriving priority vectors from reciprocal pairwise comparison matrices. *Asia-Pacific Journal of Operational Research*, 25(3):279–299.

Bosman, P. A. and Thierens, D. (2003). The balance between proximity and diversity in multiobjective evolutionary algorithms. evolutionary computation. *IEEE Transactions on Evolutionary Computation*, 7(2):174–188.

Cui, X., Li, M., and Fang, T. (2001). Study of population diversity of multiobjective evolutionary algorithm based on immune and entropy principles. In *Proc. of IEEE Congress on Evolutionary Computation*, pages 1316–1321.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197.

Han, K.-H. and Kim, J.-H. (2002). Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. *IEEE Transactions on Evolutionary Computation*, 6(6):580–593.

Han, K.-H. and Kim, J.-H. (2004). Quantum-inspired evolutionary algorithms with a new termination criterion, hε gate, and two phase scheme. *IEEE Transactions on Evolutionary Computation*, 8(2):156–169.

Hey, T. (1999). Quantum computing: an introduction. *Computing and Control Engineering Journal*, 10(3):105–112.

Kim, J.-H., Han, J.-H., Kim, Y.-H., Choi, S.-H., and Kim, E.-S. (2012). Preference-based solution selection algorithm for evolutionary multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 16(1):20–34.

Kim, J.-H., Kim, Y.-H., Choi, S.-H., and Park, I.-W. (2009). Evolutionary multi-objective optimization in robot soccer system for education. *IEEE Computational Intelligence Magazine*, 4(1):31–41.

Kim, Y.-H., Kim, J.-H., and Han, K.-H. (2006). Quantum-inspired multiobjective evolutionary algorithm for multiobjective 0/1 knapsack problems. In *Proc. of IEEE Congress on Evolutionary Computation*, pages 2601–2606.

Laumanns, M., Thiele, L., Deb, K., and Zitzler, E. (2002). Combining convergence and diversity in evolutionary multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 10(3):263–282.

Lee, K.-B. and Kim, J.-H. (2012). Multiobjective particle swarm optimization with preference-based sort and its application to path following footstep optimization for humanoid robots. *IEEE Transactions on Evolutionary Computation*, 17(6):755–766.

Li, H., Zhang, Q., Tsang, E., and Ford, J. (2004). Hybrid estimation of distribution algorithm for multiobjective knapsack problem. *Evolutionary Computation in Combinatorial Optimization*, pages 145–154.

Ryu, S.-J., Lee, K.-B., and Kim, J.-H. (2012). Improved version of a multiobjective quantum-inspired evolutionary algorithm with preference-based selection. In *Proc. of IEEE World Congress on Computational Intelligence*, pages 1672–1678.

Zitzler, E. (1999). *Evolutionary algorithms for multiobjective optimization: methods and applications*. Berichte aus der Informatik, Shaker Verlag, Aachen-Maastricht.