Chinese Journal of Aeronautics, (2014),27(6): 1477–1487

CSAA

Chinese Society of Aeronautics and Astronautics
& Beihang University

Chinese Journal of Aeronautics

cja@buaa.edu.cn
www.sciencedirect.com

Uncertain multiobjective redundancy allocation problem of repairable systems based on artificial bee colony algorithm

Guo Jiansheng [a,1], Wang Zutong [a], Zheng Mingfa [b,*], Wang Ying [a]

[a] Materiel Management and Safety Engineering College, Air Force Engineering University, Xi'an 710051, China
[b] School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710021, China

Received 18 November 2013; revised 4 August 2014; accepted 20 August 2014
Available online 20 October 2014

KEYWORDS

Artificial bee colony algorithm;
Multiobjective optimization;
Redundancy allocation problem;
Repairable systems;
Uncertainty theory

Abstract Based on the uncertainty theory, this paper is devoted to the redundancy allocation problem in repairable parallel-series systems with uncertain factors, where the failure rate, repair rate and other relative coefficients involved are considered as uncertain variables. The availability of the system and the corresponding designing cost are considered as two optimization objectives. A crisp multiobjective optimization formulation is presented on the basis of uncertainty theory to solve this resultant problem. For solving this problem efficiently, a new multiobjective artificial bee colony algorithm is proposed to search the Pareto efficient set, which introduces rank value and crowding distance in the greedy selection strategy, applies fast non-dominated sort procedure in the exploitation search and inserts tournament selection in the onlooker bee phase. It shows that the proposed algorithm outperforms NSGA-II greatly and can solve multiobjective redundancy allocation problem efficiently. Finally, a numerical example is provided to illustrate this approach.

1. Introduction

Many real system design problems require the use of redundancy to meet high reliability specifications. A redundancy allocation problem (RAP) basically involves the determination of the number of redundancies to be allocated in each subsystem with the purpose of maximizing system reliability, which is a fundamental reliability optimization model and can be formulated as a difficult combinatorial optimization problem. The RAP with single objective has been extensively studied.[1,2] With the development of system design, it has been increasingly recognized that many practical design situations we encounter often involve multiple and conflicting objectives, which should be considered and optimized at the same time. For instance, it is often required to minimize the total system cost while maximizing the system reliability simultaneously. Considering that the decision-makers always require a full consideration of possible trade-offs and an availability-cost report in materiel solution

* Corresponding author. Tel.: +86 15289351366.
E-mail addresses: amisc@163.com (J. Guo), mingfazheng@126.com (M. Zheng).
1 Tel.: +86 29 84786546.
Peer review under responsibility of Editorial Committee of CJA.

ELSEVIER Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.cja.2014.10.014

analysis, researchers investigating RAP in such kind of complex situations often look for the determination of an entire Pareto optimal solution set.[3–5] These studies mentioned are all performed in deterministic environment, in which the reliability values of the components are assumed to be deterministic and known with certainty.

Unfortunately, in many practical situations, it is impossible to determine a fixed number that shows reliability of a component under all conditions. Furthermore, it is often difficult to accurately assess system failure rate and repair rate until the system is deployed or fielded. Thus, the RAP is always indeterministic with vague or imprecise statements, which can be boiled down to problems of observing the parameters themselves, deficiency in history and statistical data, insufficient theory, incomplete knowledge and the subjectivity and preference of human judgment, etc. With the wide application of fuzzy set theory[6] in engineering practice, many scholars consider this kind of indeterminacy as fuzziness and study RAP as a fuzzy programming problem.[7–9] In these literatures the parameters (coefficients) involved are treated as fuzzy variables, and the possibility measure[10] is applied to describing or formulate RAP as an imprecise model.

However, since the possibility measure has no self-duality, one event with possibility measure of 1 may be an impossible event, while one event with possibility measure of 0 may be a certain event.[11] That is to say, it is not reasonable to use the possibility measure to characterize the performance of redundancy system and optimize RAP in such kind of environment. Actually, these types of indeterminacy in RAP mentioned above should be called uncertainty, rather than fuzziness. A lot of surveys have shown that human beings usually overweight unlikely events, and the personal belief degree may have much larger variance than the real frequency.[12] Liu[13] declared that it is inappropriate to apply both probability theory and fuzzy set theory to uncertainty, because both theories may lead to counterintuitive results in this case. In order to deal with such kind of uncertainty problem, Liu[14] founded the uncertainty theory, which is a branch of mathematics based on normality, duality, subadditivity, and product axioms, as a means of handling uncertainty that is due to imprecision rather than randomness. So far, there has been little research on RAP using uncertainty theory, which is indeed one of the most important areas in decision analysis because many real world decision problems involve uncertainty. The RAP with multiple optimization objectives in uncertain environment is called uncertain multiobjective RAP (UMRAP), and the optimization problem under consideration becomes an uncertain multiobjective programming problem.

In this paper, a UMRAP in repairable series-parallel systems is studied, with the aim of maximizing the system availability A_s while minimizing the total cost C_s simultaneously, where the failure rate, repair rate and coefficients in objective functions are considered as uncertain variables. As a consequence, the objectives are also uncertain variables. Since the uncertain variables cannot be compared directly, the equivalent deterministic models should be proposed to remove the uncertain ambiguity first. Different real-life RAPs call for different meanings of deterministic models to satisfy their needs in practical application. Given the fact that the expected value of uncertain variable is widely used in real-life problem, in this paper expected value of availability and cost are considered. Moreover, when the designer is risk averse, the obtained design has to be done with high confidence levels, and the designs with large deviation are not desirable. Therefore, the associated variance of availability and cost are also considered. Thus, the uncertain biobjective RAP presented in this paper can be converted into a deterministic RAP with four objectives. However, for a programming problem with four objectives, the Pareto optimal set obtained always contains hundreds or even thousands of Pareto efficient solutions. It is difficult for designers to find satisfactory and meaningful trade-offs, and to select a preferred final design solution. To reduce the Pareto optimal set and to achieve a smaller practical set that can be easily analyzed by the designers, a new method, which involves breaking the original RAP with four objectives into two biobjective RAPs, is proposed in this paper. That is, optimize the expected value and associated variance of availability and cost respectively, then obtain two Pareto optimal sets in these two biobjective RAPs. It is proved that the intersection of these two Pareto optimal sets is Pareto efficient to the original RAP with four objectives. The solutions in the intersection will guarantee that both the expected value and associated variance are desirable. Considering the uncertain and NP-hard nature in UMRAP, where the size of the problem and thus the computational effort increases exponentially, meta-heuristics and evolutionary algorithms should be widely applied to UMRAP for successful generation of optimal solutions.

The artificial bee colony (ABC) algorithm, a meta-heuristic bionic algorithm based on the intelligent foraging behavior of honey bees proposed by Karaboga in 2005,[15] is a relatively new member of swarm intelligence. ABC algorithm is one of the adaptive meta-heuristic optimization methods inspired by nature, which is distinctly different from its siblings, such as genetic algorithms and ant colony optimization, in that it is a constructive, rather than an improvement, algorithm. It is inspired by the behavior of real honey bees foraging behavior, where the self-organization and division of labor features can be seen clearly. Especially, in ABC algorithm, the possible solutions are represented by the positions of food source, rather than the individuals. The ABC algorithm provides a new idea for the research of meta-heuristic algorithm and becomes one of the important research directions of solving complex optimization problem gradually. So far, due to its simplicity and ease of implementation, the ABC algorithm has been adopted by researchers in a variety of fields, including machines scheduling problem,[16] flexible job-shop scheduling problem,[17] hybrid intelligent problem,[18] etc. And it has been experimentally validated that its effectiveness and efficiency on algorithm performance are competitive to other optimization algorithms.[19–21]

To solve the uncertain, NP-hard and multiobjective characteristic of UMRAP, in this paper, a modified multiobjective ABC (MOABC) algorithm is designed for obtaining Pareto optimal set in UMRAP, which inserts the fast non-dominated sort procedure from the well-known fast non-dominated sorting genetic algorithm (NSGA-II)[22] into basic ABC algorithm. To the best of our knowledge, this paper is the first application of ABC algorithm in reliability design with multiple objectives. In order to test the performance of the proposed MOABC algorithm in multiobjective optimization problem, three well-known test problems are presented to compare its performance with NSGA-II, the result of which shows that the MOABC outperforms NSGA-II greatly. Moreover, a multiobjective RAP from Ref. 23 is presented and solved by MOABC, and it shows that the obtained result in Ref. 23 is not on the Pareto front found by MOABC. This indicates that the application of MOABC in UMRAP is more meaningful and practical.

The paper is organized in the following manner. In Section 2, some useful definitions and properties about uncertainty theory with application to UMRAP are introduced. In Section 3, the UMRAP in repairable series-parallel systems is described and its corresponding mathematical model is presented; the approach generating Pareto optimal set is proposed and its validity is proved. In Section 4, the classic ABC algorithm is introduced briefly, and a modified ABC algorithm for solving multiobjective optimization is proposed. In Section 5, an application case study is provided to illustrate the solution of UMRAP and demonstrate the efficacy and efficiency of MOABC for complicated reliability optimization problems. Finally, the major results of the research are presented.

2. Preliminaries

In this section, some foundational concepts and properties of uncertainty theory are introduced, which will be used throughout this paper.

Let Γ be a nonempty set, and $\mathcal{L}$ a σ -algebra over Γ. Each element Λ in $\mathcal{L}$ is called an event. A set function $\mathcal{M}$ from $\mathcal{L}$ to [0, 1] is called an uncertain measure if it satisfies the following axioms.[24]

Axiom 1 (Normality axiom). $\mathcal{M}\{\Gamma\} = 1$ for the universal set Γ.

Axiom 2 (Duality axiom). $\mathcal{M}\{\Lambda\} + \mathcal{M}\{\Lambda^c\} = 1$ for any event Λ.

Axiom 3 (Subadditivity axiom). For every countable sequence of events Λ_1, Λ_2, $\cdots$, we have

$$\mathcal{M}\bigcup_{i=1}^{\infty}\Lambda_i \leqslant \sum_{i=1}^{\infty}\mathcal{M}\{\Lambda_i\} \tag{1}$$

In 2009, Liu proposed the fourth axiom of uncertainty theory called product measure axiom.[25]

Axiom 4 (Product axiom). Let $(\Gamma_k, \mathcal{L}_k, \mathcal{M}_k)$ be uncertainty spaces for $k = 1, 2, \cdots$. The product uncertain measure $\mathcal{M}$ is an uncertain measure satisfying:

$$\mathcal{M}\left\{\prod_{k=1}^{\infty}\Lambda_k\right\} = \bigwedge_{k=1}^{\infty}\mathcal{M}_k\{\Lambda_k\} \tag{2}$$

where Λ_k are arbitrarily chosen events from $\mathcal{L}_{\|}$ for $k = 1, 2, ...$, respectively. The triplet $(\Gamma, \mathcal{L}, \mathcal{M})$ is referred to as an uncertainty space,[24] in which an uncertain variable is defined as follows.

Definition 1 (Ref. 24). An uncertain variable is a measurable function ξ from an uncertainty space $(\Gamma, \mathcal{L}, \mathcal{M})$ to the set of real numbers, i.e., for any Borel set B of real numbers, the set

$$\{\xi \in B\} = \{\gamma \in \Gamma | \xi(\gamma) \in B\} \tag{3}$$

is an event.

Definition 2 (Ref. 25). The uncertain variables ξ_1, ξ_1, $\cdots$, ξ_n are said to be independent if

$$\mathcal{M}\{\bigcap_{i=1}^{n}(\xi_i \in B_i)\} = \bigwedge_{i=1}^{n}\mathcal{M}\{\xi_i \in B_i\} \tag{4}$$

for any Borel sets B_1, B_2, $\cdots$, B_n of real numbers.

Definition 3 (Ref. 14). The uncertainty distribution Φ of an uncertain variable ξ is defined by

$$\Phi(x) = \mathcal{M}\{\xi \leqslant x\} \tag{5}$$

for any real number x.

Definition 4 (Ref. 26). Let ξ be an uncertain variable with regular uncertainty distribution Φ. Then the inverse function Φ^{-1} is called the inverse uncertainty distribution of ξ.

Definition 5 (Ref. 24). Let ξ be an uncertain variable. Then the expected value of ξ is defined by

$$E[\xi] = \int_0^{\infty}\mathcal{M}\{\xi \geqslant x\}\mathrm{d}r - \int_{-\infty}^{0}\mathcal{M}\{\xi \leqslant x\}\mathrm{d}r \tag{6}$$

provided that at least one of the two integrals is finite.

Definition 6 (Ref. 24). Let ξ be an uncertain variable with finite expected value e. Then the variance of ξ is:

$$V(\xi) = E[(\xi - e)^2] \tag{7}$$

For calculation convenience, we stipulate that the variance of ξ is

$$V(\xi) = 2\int_0^{+\infty} x(1 - \Phi(e + x) + \Phi(e - x))\mathrm{d}x \tag{8}$$

Definition 7 (Ref. 12). An uncertain variable ξ is called linear if it has a linear uncertainty distribution:

$$\Phi(x) = \begin{cases} 0 & \text{if } x \leqslant a \\ (x-a)/(b-a) & \text{if } a \leqslant x \leqslant b \\ 1 & \text{if } x \geqslant b \end{cases} \tag{9}$$

denoted by $L(a, b)$ where a and b are real numbers with $a < b$.

Definition 8 (Ref. 12). An uncertain variable ξ is called zigzag if it has a zigzag uncertainty distribution:

$$\Phi(x) = \begin{cases} 0 & \text{if } x \leqslant a \\ \dfrac{x-a}{2(b-a)} & \text{if } a \leqslant x \leqslant b \\ \dfrac{x+c-2b}{2(c-b)} & \text{if } b \leqslant x \leqslant c \\ 1 & \text{if } x \geqslant c \end{cases} \tag{10}$$

denoted by $Z(a, b, c)$ where a, b, c are real numbers with $a < b < c$.

Definition 9 (Ref. 12). A lognormal uncertain variable has an uncertainty distribution:

$$\Phi(x) = \left[1 + \exp\left(\frac{\pi(e - \ln x)}{\sqrt{3}\sigma}\right)\right]^{-1}, \ x \geqslant 0 \tag{11}$$

denoted by LOGN(e, σ), where e and σ are real numbers with $\sigma > 0$.

Definition 10 (Ref. 27). For a multiobjective programming problem $\min(f_1(x), f_2(x), \cdots, f_m(x))$, s.t., $x \in X$, a solution $x \in X$ is called efficient with respect to objectives $f_1, f_2, \cdots, f_m$ if it is not dominated by any feasible $y \in X, y \neq x$, that is to say if there is no $y \in X$ such that: (A) $f_j(y) \leqslant f_j(x)$ for all $j = 1, 2, \cdots, m$, and (B) $f_j(y) < f_j(x)$ for at least one j. The set of efficient solutions is called the efficient set. The image point $(f_1(x), f_2(x), \cdots, f_m(x))$ of an efficient solution x is called Pareto-optimal, and the set of Pareto-optimal points, i.e., the image of the efficient set in the objective space R^m under $f = (f_1, f_2, \cdots, f_m)$, is called the Pareto frontier.

Theorem 1 (Ref. 14). *Let $\xi_1, \xi_2, \cdots, \xi_n$ be uncertain variables, and f a real-valued measurable function. Then $f(\xi_1, \xi_2, \cdots, \xi_n)$ is an uncertain variable.*

Theorem 2 (Ref. 26). *Let ξ be an uncertain variable with regular uncertainty distribution Φ. If the expected value exists, then*

$$E(\xi) = \int_0^1 \Phi^{-1}(\alpha)\mathrm{d}\alpha \tag{12}$$

Theorem 3 (Ref. 26). *Let $\xi_1, \xi_2, \cdots, \xi_n$ be independent uncertain variables with regular uncertainty distributions $\Phi_1, \Phi_2, \cdots, \Phi_n$, respectively. If the function $f(x_1, x_2, \cdots, x_n)$ is strictly increasing with respect to $x_1, x_2, \cdots, x_m$ and strictly decreasing with respect to $x_{m+1}, x_{m+2}, \cdots, x_n$, then $\xi = f(\xi_1, \xi_2, \cdots, \xi_n)$ is an uncertain variable with inverse uncertainty distribution:*

$$\begin{aligned}\Psi^{-1}(\alpha) = f(&\Phi_1^{-1}(\alpha), \Phi_2^{-1}(\alpha), \cdots,\\ &\Phi_m^{-1}(\alpha), \Phi_{m+1}^{-1}(1-\alpha), \Phi_{m+2}^{-1}(1-\alpha), \cdots, \Phi_n^{-1}(1-\alpha))\end{aligned} \tag{13}$$

3. UMRAP in repairable systems

3.1. Problem description

A commonly used definition of a repairable system indicates that a system can be repaired to operate normally in the event of any failure, and system availability is the main measure of interest which needs to be maximized to achieve the optimal performance. For repairable system, availability is a very meaningful measure, and achieving a high or required level of availability is an essential requisite. The most common approach to guarantee the availability level is to employ redundancy, which can be referred to as a series-parallel system. A series-parallel system is a system made of parallel subsystems put in serial, which can be shown in Fig. 1. A parallel subsystem works when at least one of its components works while a series system fails when at least one of its components fails. The problem of redundancy allocation in repairable system design can be formulated as a multiobjective optimization problem, i.e., minimize the cost while maximize the availability of the global system simultaneously. Uncertainty has always been an important consideration when designing and analyzing repairable systems. It assumes that the failure rate and repair rate of components are uncertain variables in uncertain environment, and its uncertainty distribution can be available through expert knowledge before system design or with incomplete data. In this paper, we suppose that the subsystems are parallel, and that in each subsystem all components are identical. Here, the term 'identical' means that the failure rate and repair rate of components have the same uncertainty distribution in uncertain environment. Then the decision variables of this problem are the number of components k_i in each subsystem ($i = 1, 2, \ldots, s$), and the UMRAP is to find the optimal values of k_i to minimize the cost and maximize the system availability in uncertain environment.

3.2. Mathematical formulation

The formulation of the mathematical model taking into account all the influential factors is very important for the success of optimization. Let k_i, A_s, C_s be the number of components in each subsystem, the availability of the system and the system cost, respectively. Then the UMRAP can be presented as follows:

$$\begin{cases}\max\limits_k A_s(k) & = \prod\limits_{i=1}^{s}\left[1-\left(\dfrac{\lambda_i}{\lambda_i+\mu_i}\right)^{k_i}\right]\\ \min\limits_k C_s(k) & = \sum\limits_{i=1}^{s} k_i(a_i\lambda_i^{p_i}+b_i\mu_i^{q_i})\\ \text{s.t.} & l_{ki} \leqslant k_i \leqslant u_{ki}, k_i \in N_+\end{cases} \tag{14}$$

The variables in the UMRAP model above are illustrated in Table 1.

Since maximizing the availability A_s is equivalent to minimizing the unavailability $(1 - A_s)$, the mathematical formulation can be transformed as follows:

$$\begin{cases}\min\limits_k U_s(k) & = 1 - A_s(\lambda, \mu, k)\\ \min\limits_k C_s(k) & \\ \text{s.t.} & l_{ki} \leqslant k_i \leqslant u_{ki}, k_i \in N_+,\ i = 1, 2, \ldots, s\end{cases} \tag{15}$$

Since the unavailability $U_s(k)$ and system cost $C_s(k)$ are uncertain variables, which cannot be compared and optimized directly, the equivalent deterministic models should be proposed to remove the uncertain ambiguity. In this paper, the expected value and variance of uncertain variables are adopted to convert the original UMRAP into a deterministic multiobjective RAP as follows:

$$\begin{cases}\min\limits_k (E[U_s(k)], & E[C_s(k)], V[U_s(k)], V[C_s(k)])\\ \text{s.t.} & l_{ki} \leqslant k_i \leqslant u_{ki}, k_i \in N_+,\ i = 1, 2, \ldots, s\end{cases} \tag{16}$$

It can be obtained that $U_s(k)$ is strictly increasing with respect to λ_i and strictly decreasing with respect to μ_i; $C_s(k)$ is strictly increasing with respect to μ_i, a_i, b_i and strictly

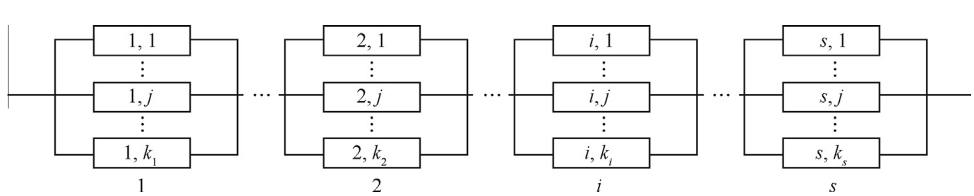

Fig. 1 General structure of a parallel-series system.

Table 1 Nomenclature used in Eq. (14).

Variable	Signification
s	Number of subsystems
i	Index of a subsystem
λ_i	Positive uncertain variable of the failure rate of components in subsystem i defined on the uncertainty space $(\Gamma, \mathcal{L}, \mathcal{M})$, with uncertainty distribution $\Phi_i(x)$
λ	$\lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_s\}$
μ_i	Positive uncertain variable of the repair rate of components in subsystem i defined on the uncertainty space $(\Gamma, \mathcal{L}, \mathcal{M})$, with uncertainty distribution $\Psi_i(x)$
a	$a = \{a_1, a_2, \ldots, a_s\}$
b	$b = \{b_1, b_2, \ldots, b_s\}$
μ	$\mu = \{\mu_1, \mu_2, \ldots, \mu_s\}$
k_i	Number of components in subsystem i
k	$k = \{k_1, k_2, \ldots, k_s\}$
l_k, u_k	Vector of minimum and maximum number of components allowable in each subsystem
a_i	Positive uncertain variable defined on the uncertainty space $(\Gamma, \mathcal{L}, \mathcal{M})$, with uncertainty distribution $\Upsilon_{ai}(x)$
b_i	Positive uncertain variable defined on the uncertainty space $(\Gamma, \mathcal{L}, \mathcal{M})$, with uncertainty distribution $\Upsilon_{bi}(x)$
p_i, q_i	Constants of the cost function where $p_i < 0$, $q_i > 0$

decreasing with respect to λ_i. Then according to Definitions 5, 6 and Theorems 2, 3, the closed forms of $E[U_s(k)]$, $E[C_s(k)]$, $V[U_s(k)]$, $V[C_s(k)]$ are presented as follows:

$$E[U_s(k)] = \int_0^1 1 - \prod_{i=1}^{s}\left[1 - \left(\frac{\Phi_i^{-1}(\alpha)}{\Phi_i^{-1}(\alpha) + \Psi_i^{-1}(1-\alpha)}\right)^{k_i}\right]\mathrm{d}\alpha \tag{17}$$

$$E[C_s(k)] = \int_0^1 \sum_{i=1}^{s} k_i(\Upsilon_{ai}^{-1}(\alpha)\Phi_i^{-1}(1-\alpha)^{p_i} + \Upsilon_{bi}^{-1}(\alpha)\Psi_i^{-1}(\alpha)^{q_i})\mathrm{d}\alpha \tag{18}$$

Let e_u, e_c, Φ_u and Φ_c be the $E[U_s(k)]$, the $E[C_s(k)]$, the uncertainty distribution of uncertain objective U_s and the uncertainty distribution of uncertain objective $C_s(k)$ respectively, then

$$V[U_s(k)] = 2\int_0^{+\infty} x(1 - \Phi_u(e_u + x) + \Phi_u(e_u - x))\mathrm{d}x \tag{19}$$

$$V[C_s(k)] = 2\int_0^{+\infty} x(1 - \Phi_c(e_c + x) + \Phi_c(e_c - x))\mathrm{d}x \tag{20}$$

In this paper, the calculation of expected value and variance is on the basis of MATLAB Uncertainty Toolbox[28] directly.

3.3. Solution approach

There are two main ways to solve the multiobjective programming problem. The first way is to aggregate the multiple objectives into a single objective, which is then solved for an optimal solution, while the second way is to obtain a list of Pareto optimal solutions (i.e., efficient set) first, and then apply the decision maker preferences to find solution(s) of his/her choice. The second way can be preferable if the decision maker wants to consider different preferences, and compare solutions giving different priorities to the objectives in order to observe how the solutions change. However, in the solution of quadruple-objective programming problem Eq. (16), its Pareto efficient set consists of thousands of solutions, in which many are undesirable. For instance, one solution with small variance but huge expected value of unavailability and cost is a Pareto efficient solution in Eq. (16), but it maybe undesirable in practical application. To pick a single solution that best reflects one's preferences in such an efficient set is a daunting task. To alleviate this problem, a new solution approach is proposed to obtain a pruned efficient set, which involves breaking the quadruple-objective programming Eq. (16) into two biobjective programming problems as follows. The pruned efficient set in Eq. (16) is the intersection of efficient sets in Eqs. (21) and (22).

$$\begin{cases} \max\limits_{k}(E[U_s(k)], \quad E[C_s(k)]) \\ \text{s.t.} \qquad l_{ki} \leqslant k_i \leqslant u_{ki},\ k_i \in N_+,\ i = 1, 2, \ \ldots,\ s \end{cases} \tag{21}$$

$$\begin{cases} \max\limits_{k}(V[U_s(k)], \quad V[C_s(k)]) \\ \text{s.t.} \qquad l_{ki} \leqslant k_i \leqslant u_{ki}, k_i \in N_+,\ i = 1, 2, \ \ldots,\ s \end{cases} \tag{22}$$

Proposition 1. *Let A be the Pareto efficient set of quadruple-objective programming problem $P : \min_{x\in D}(f_1(x), f_2(x), f_3(x), f_4(x))$, A_1 and A_2 be the Pareto efficient set of biobjective programming problem $P_1 : \min_{x\in D}(f_1(x), f_2(x))$ and $P_2 \min_{x\in D}(f_3(x), f_4(x))$ respectively, then $A_1 \cap A_2 \subset A$.*

Proof. If x^* belongs to A_1, by the Definition 10, there does not exist $x \in D$ such that $f_i(x) \leqslant f_i(x^*)$, at lest one i_0 such that $f_{i0}(x) < f_{i0}(x^*), i = 1, 2$.

Similarly, if x^* belongs to A_2, we can obtain that there does not exist $x \in D$ such that $f_j(x) \leqslant f_j(x^*)$, at least one j_0 such that $f_{j0}(x) < f_{j0}(x^*), j = 1, 2$.

Since $x^* \in A_1 \cap A_2$, it is evident that there does not exist $x \in D$ such that $f_k(x) \leqslant f_k(x^*)$, at least one k_0 such that $f_{k0}(x) < f_{k0}(x^*), k = 1, 2, 3, 4$, which means that x^* is Pareto efficient solution to problem P, that is, $A_1 \cap A_2 \subset A$. The theorem is proved.

According to Proposition 1, the solution of Eq. (16) is converted into solution of its two Eqs. (21) and (22). In order to solve UMRAP efficiently, a new multiobjective ABC algorithm is proposed in the next section.

4. A modified ABC algorithm

Considering the uncertain, multiobjective and combinatorial nature in UMRAP, in order to improve the quality and spread of the solutions, a variant of basic ABC algorithm combined with NSGA-II is proposed for multiobjective optimization problem in this section.

4.1. Basic ABC algorithm

In the basic ABC algorithm, there are three essential components, that is, food source positions, nectar-amount, and three kinds of foraging bees (employed bees, onlookers, and scouts). Each food source position represents a feasible solution to the optimization problem considered and the nectar-amount of a food source corresponds to the quality (fitness) of the solution represented by that food source. Each kind of foraging bee performs one particular operation for generating new candidate food source positions. Employed bees are those bees which are searching the food around the food source in their memory currently; they are responsible for sharing the information about food sources with onlooker bees. Onlooker bees are those bees which are waiting in the hive for the information from the employed bees; they tend to choose good food source with more nectar-amount shared by the employed bees, and then further tap the foods around the selected food source. Scout bees are those bees which are carrying out random searches for discovering new food sources if the employed bees and onlookers cannot find a better neighboring food source. Thus, the ABC algorithm visualizes the employed and onlooker bees as performing the job of local search (exploitation), whereas the onlookers and scouts bees as performing the job of global search (exploration).

4.2. Framework of MOABC

(1) Population initialization

In our proposed algorithm, the solutions (food positions) of UMRAP $k = \{k_1, \cdots, k_i, \cdots, k_s\}$ is represented as the variable $x = \{x_1, \cdots, x_i, \cdots, x_s\}$, and the integer decision variables k_i are treated as real variables. While in the evaluation of objective functions, the real values x_i are transformed to the nearest integer values. Based on the food position representation, we can initialize the population with S real variables in the bound of decision variables randomly, where S is the number of food sources, that is:

$$x_i^j = l_{ki} + \mathrm{r\ and}\ \ (0,1)(u_{ki} - l_{ki}) \tag{23}$$

where $i \in \{1, 2, \cdots, s\}$, and $j \in \{1, 2, \cdots, S\}$.

(2) Exploitation search

Usually employed bees and onlooker bees use the same search operator to perform exploitation search. Since the food positions are represented as real variable vectors, the exploitation search in basic ABC algorithm is applied here. The exploitation search for food position x_i^j is as follows:

$$v_i^j = x_i^j + r_i^j(x_i^j - x_i^k) \tag{24}$$

where $k \in \{1, 2, \cdots, s\}$ is generated randomly, and $k \neq j$, r_i^j is random number in $[-1, 1]$.

(3) Greedy selection strategy

In the employed and onlooker bee phase, the greedy selection strategy should be applied to selecting the better foods after exploitation search. In the basic ABC algorithm, the greedy selection is to select a better solution based on the fitness of food source. However, in the MOABC, the goal is to obtain an efficient set, rather than a single efficient solution. Thus, the rank and crowding distance of solution in NSGA-II are adopted here to select better solutions and maintain the Pareto optimality and diversity in the efficient set.

Rank is used to divide the solutions into several levels according to the dominance degree. The first front contains all the nondominated solutions in current population, and the second front contains all the solutions that are dominated by the individuals in the first front only, and so on. Individuals in the first front are given a rank value of 1, and individuals in second front are assigned a rank value of 2.

For the individuals in the same rank, large crowding distance will result in better diversity in the population. After the solutions are divided in respective fronts, the members of the same rank sequence is in an ascent order according to each objective, and then the crowding of each solution is defined as the sum of the normalized distance between its right and left neighbors in the sequence. For the first and last solutions in every front, their crowding distances are defined as infinity. In this paper, the crowding distance d_i by considering two objectives is defined as follows:

$$d_i = \begin{cases} \infty & \text{if } i = 1 \text{ or } i = \text{Last} \\ \sum_{j=1}^{2} \dfrac{\mathrm{obj}_j(s_{i+1}) - \mathrm{obj}_j(s_{i-1})}{\mathrm{obj}_j^{\max} - \mathrm{obj}_j^{\min}} & \text{otherwise} \end{cases} \tag{25}$$

where $\mathrm{obj}_j(s_i)$ is the value of the jth objective of s_i, and $\mathrm{obj}_j^{\max}$ and $\mathrm{obj}_j^{\min}$ are the maximal and minimal values of the jth objective known so far. Based on the rank value and crowding distance, solution a is better than b if $r_a < r_b$ or ($r_a = r_b$ and $d_a > d_b$).

(4) Employed bee phase

In the employed bee phase, the exploitation search procedure is applied to generate new neighboring food sources, and the new greedy selection strategy is adopted to compare the new generated solution with the original solution. When the generated solution is better than original solution, it will replace the original solution and update the population.

(5) Onlooker bee phase

In the onlooker bee phase, since it is difficult to determine the fitness value in multiobjective optimization problem, it is hard to choose food source according to the probability value which is proportional to its fitness value. Therefore, the tournament selection with size 4 is applied in this paper. Specifically, randomly select four employed bee solutions from the population, and then determine the best employed bee solution as the food source of the onlooker bee according to its distance. Then use the exploitation search procedure to generate new neighboring solution and greedy selection strategy to update the population.

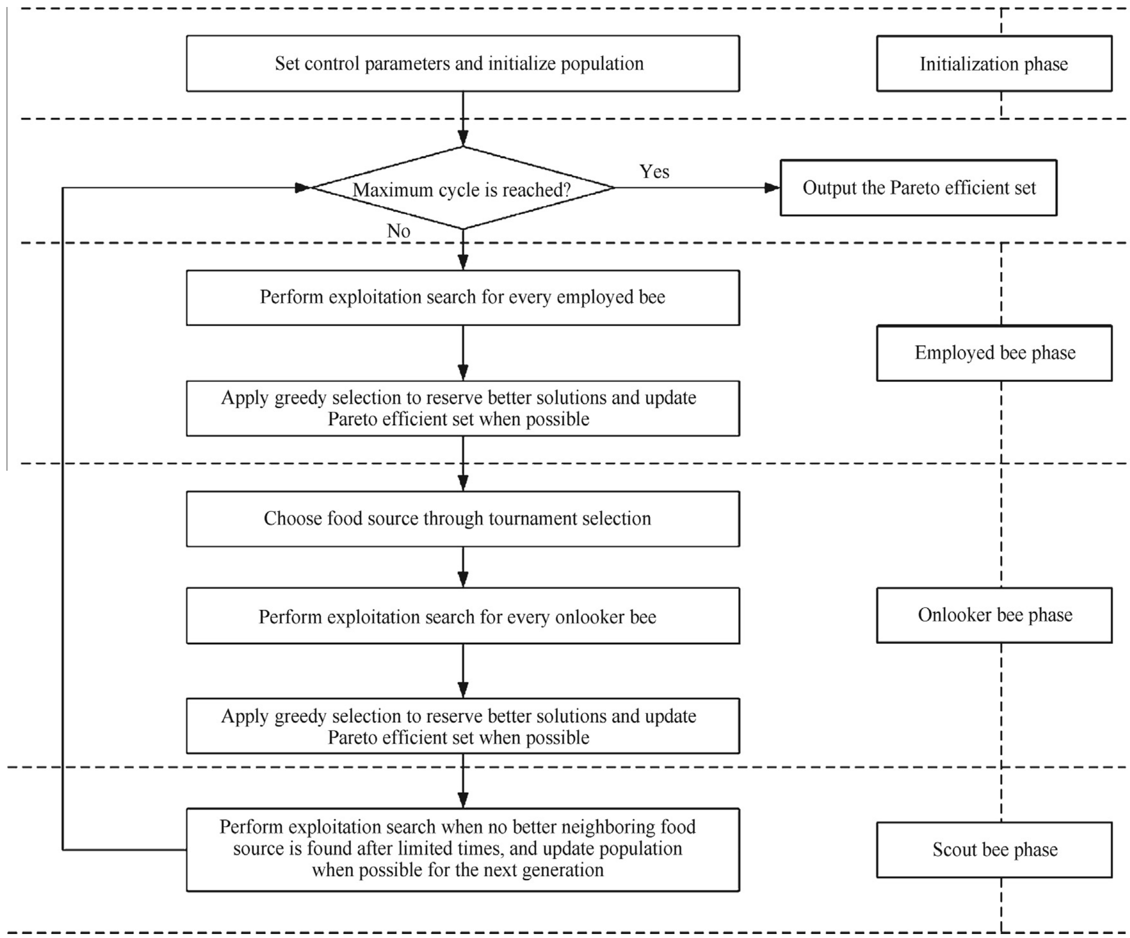

Fig. 2 Framework of MOABC for multiobjective optimization problem.

(6) Scout bee phase

In the scout bee phase, the solution is generated randomly to replace the worst solution in the population according to rank and crowding distance. Since the solution is generated randomly, global exploration is stressed in scout bee phase and it also may help enhance population diversity to some extent.

Straightforwardly, the framework of the proposed MOABC for multiobjective optimization problem is illustrated in Fig. 2. It can be seen that the proposed ABC algorithm not only applies rank value to generate new neighboring food sources at different levels, but also applies crowding distance to maintain the solution diversity. It stresses the balance of the global exploration and local exploitation; at the same time, it also stresses the diversity of population during the searching process, which will assure that a desirable Pareto efficient set is available.

4.3. Performance test

(1) Test on real-value functions

There are many test functions for multiobjective optimization. To test the performance of MOABC proposed in this paper, three biobjective test problems with convex, non-convex and discontinuous Pareto fronts respectively are selected as follows.[29]

Test problem ZDT1 (convex Pareto-optimal front).

$$\begin{cases} f_1(x) = x_1 \\ f_2(x) = g(1 - \sqrt{f_1/g}) \\ g = 1 + (9\sum_{i=2}^{d} x_i)/(d-1) \\ x_i \in [0,1], \quad i = 1, 2, \cdots, 30 \end{cases} \tag{26}$$

where d is the number of dimensions. The Pareto-optimality is reached when $g = 1$.

Test problem ZDT2 (non-convex Pareto-optimal front).

$$\begin{cases} f_1(x) = x_1 \\ f_2(x) = g(1 - (f_1/g)^2) \end{cases} \tag{27}$$

Test problem ZDT3 (discontinuous Pareto-optimal front).

$$\begin{cases} f_1(x) = x_1 \\ f_2(x) = g[1 - \sqrt{f_1/g} - \dfrac{f_1}{g}\sin(10\pi f_1)] \end{cases} \tag{28}$$

where g in functions ZDT2 and ZDT3 is the same as in function ZDT1. In the ZDT3 function, f_1 varies from 0 to 0.852 and f_2 from −0.773 to 1.

After generating 100 Pareto points by MOABC, the Pareto front generated by MOABC is compared with the true front of three test problems. In all the rest of the figures, the vertical axis is for f_2 while the horizontal axis is for f_1. The performance measure is defined as the mean square error (MSE)

Table 2 Control parameters adopted in the ABC algorithm.

Parameter	Value
Colony size	200
Limit	100
Number of onlookers	Half of the colony size
Number of employed bees	Half of the colony size
Number of scouts	1

between the estimate Pareto front PF^e to its correspond true front PF^t as

$$M_f = \frac{1}{N}\sum_{j=1}^{N}(PF_j^e - PF_j^t)^2 \tag{29}$$

where N is the number of points.

To compare the performance of MOABC, the NSGA-II is chosen as the comparison algorithm. In the MOABC algorithm, the maximum number of cycles was taken as 500. The percentage of onlooker bees and employed bees were the half of the colony and the number of scout bees was selected to be one. Parameters set for the ABC algorithm are given in

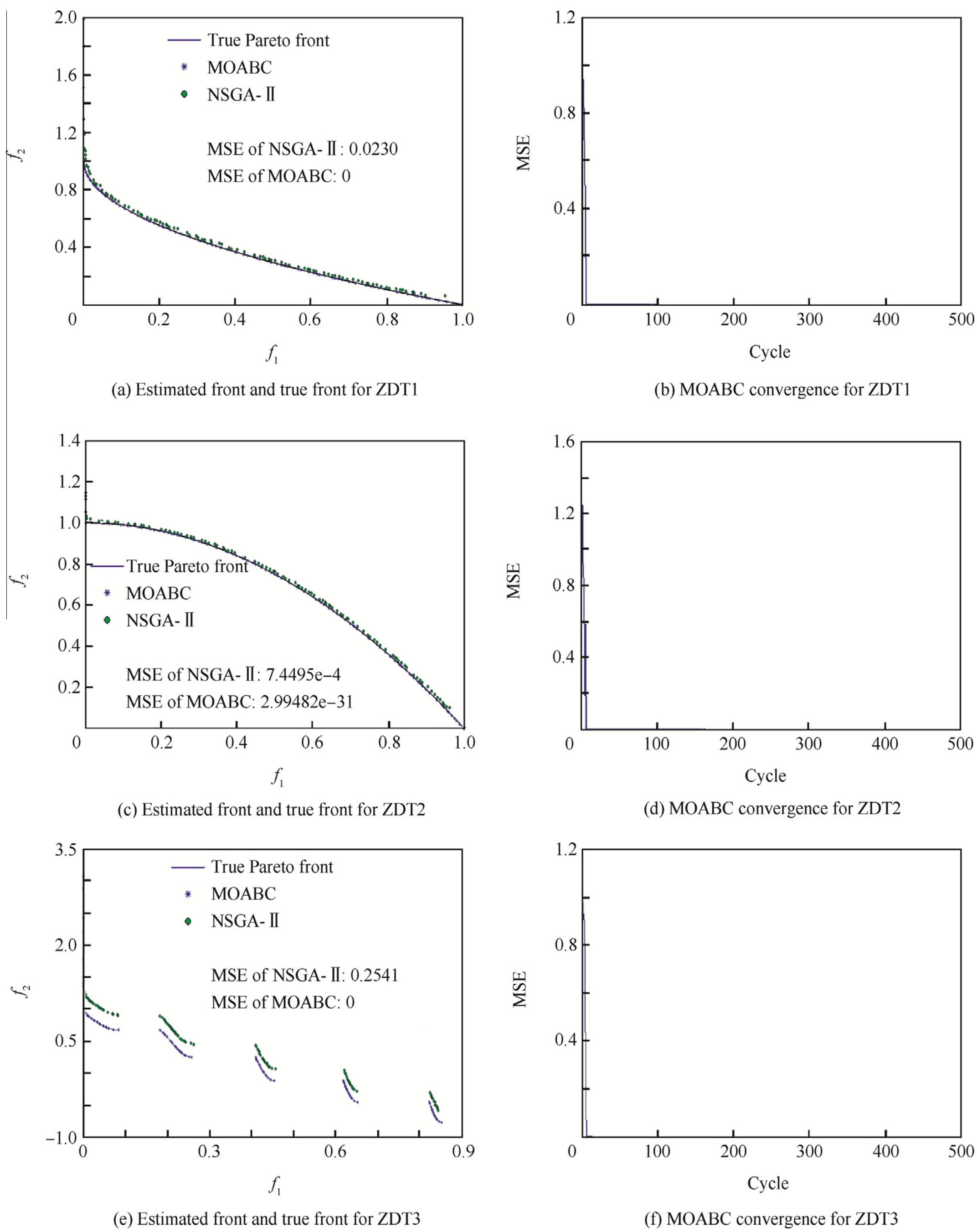

Fig. 3 Test performance of MOABC and NSGA-II on ZDT1, ZDT2, and ZDT3.

Table 2. In the NSGA-II, the population size is set as 100 and the maximum generation is set as 500.

Fig. 3 show the test performance of MOABC and NSGA-II on ZDT1, ZDT2, and ZDT3 respectively. It is clear that the estimated front obtained by MOABC is better distributed on the true front than that obtained by NSGA-II. Especially, the MOABC can obtain the two ends of Pareto front, that is to say, it will guarantee that the decision solutions are never neglected or outside the Pareto efficient set. In addition, the estimated front obtained by MOABC is much closer to the true front than that obtained by NSGA-II. The MSE of MOABC achieves zero within 200 cycles, while the MSE of NSGA-II is still much higher after 500 cycles. Undoubtedly, the proposed MOABC outperforms NSGA-II greatly, thus, it is reasonable to expect good performance of MOABC in the solution of UMRAP.

(2) Test on engineering problem

In order to further verify the applicability of MOABC in the solution of UMRAP, a multiobjective RAP problem from Ref.23 is presented here and solved by MOABC. In Ref. 23, the optimal result is $U_s = 0.0007$ ($A_s = 0.9993$), $C_s = 546.43$. The obtained result in Ref.23 and that using MOABC are shown in Fig. 4.

It is clear that the optimization result in Ref. 23 is not on the Pareto front, that is, the corresponding solution is not Pareto efficient. While using MOABC, all possible trade-offs can be presented and considered. Table 3 shows one solution from the Pareto front. It can be seen that the andof this solution are both smaller than the results in Ref. 23.

5. Case study

In this section, a case study is presented to illustrate the solution of UMRAP based on MOABC. The parameters set for MOABC are the same as shown in Table 2. The system considered is assumed to consist of six parallel subsystems in serial, that is to say, $s = 6$. The input data assumed are shown in Table 4.

Firstly, the Eq. (21) is solved by MOABC and NSGA-II adopted in Section 4.3, and the estimated fronts obtained with 100 efficient solutions are shown in Fig. 5. It is shown that the estimated front obtained using NSGA-II does not include the solution with the highest availability, and several solutions in the front are not Pareto efficient. By comparison, the solutions obtained using MOABC is better and of more diversity.

Furthermore, as can be seen in the estimated front obtained using MOABC, the expected value of cost increases exponentially when the expected value of unavailability decreases, which is basically consistent with real circumstance.

Secondly, the variance Eq. (22) is solved using MOABC, and the estimated front with 100 efficient solutions is shown in Fig. 6.

Finally, we take the intersection of the efficient sets obtained by MOABC in Eqs. (21) and (22). The result is shown in Table 5.

Table 4 Input data of UMRAP.

Input data	Detailed information
λ_i	$L(10^{-4}(2+0.8(i-1), 3.5+0.8(i-1))), i = 1, 2, \ldots, 6$
μ_i	$Z(0.1+0.25(i-1), 0.2+0.25(i-1),$ $0.25+0.25(i-1)), i = 1, 2, \ldots, 6$
a_i	$\mathrm{LOGN}(0.01(i-4)^2+0.02, 0.03), i = 1, 2, \ldots, 6$
b_i	$\mathrm{LOGN}(0.02(i-3)^2+0.03, 0.02), i = 1, 2, \ldots, 6$
l_k	$1 \times \{1, 1, 1, 1, 1, 1\}$
u_k	$6 \times \{1, 1, 1, 1, 1, 1\}$
p	$-0.8 \times \{0.4, 0.2, 0.8, 1, 1.2, 0.8\}$
q	$0.85 \times \{0.4, 0.2, 0.8, 1, 1.2, 0.8\}$

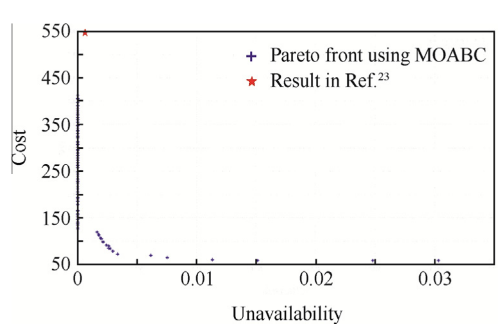

Fig. 4 Result comparison between Ref. 23 and MOABC.

Table 3 One solution from the Pareto front ($U_s = 4.1642 \times 10^{-11}$; $C_s = 289.40$).

k_i	$\lambda_i(10^{-3}\mathrm{h}^{-1})$	μ_i (h^{-1})
9	0.00198	0.2021
7	0.00200	0.3604
4	0.00193	0.7650
5	0.00198	0.4588
5	0.00200	0.5947

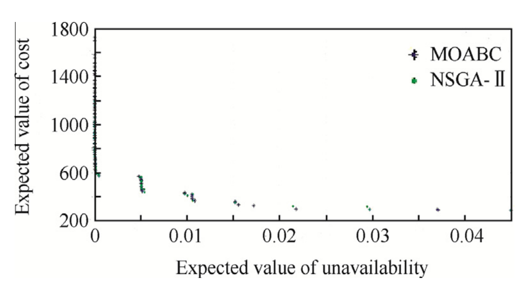

Fig. 5 Estimated front of Eq. (21) using MOABC and NSGA-II.

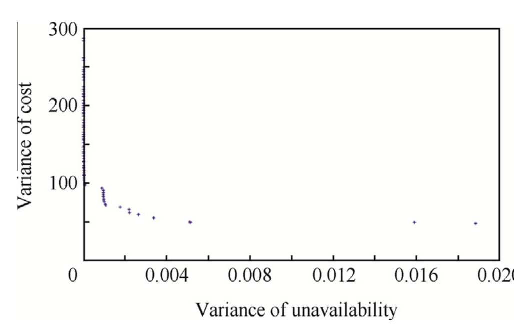

Fig. 6 Estimated front of Eq. (22) using MOABC.

Table 5 Obtained results for UMRAP.

No.	Solution	$E[U_s]$	$E[C_s]$ (unit)	$V[U_s]$	$V[C_s]$ (unit)
1	6 6 5 4 3 6	1.31×10^{-7}	1120	1.47×10^{-7}	181
2	6 6 4 3 3 4	3.07×10^{-7}	960	3.82×10^{-7}	156
3	6 5 4 4 3 5	1.33×10^{-7}	1060	1.51×10^{-7}	172
4	6 5 5 5 5 5	2.03×10^{-10}	1450	1.35×10^{-9}	240
5	6 5 5 4 3 5	1.31×10^{-7}	1090	1.47×10^{-7}	177
6	6 6 3 3 2 3	2.59×10^{-5}	754	1.08×10^{-5}	122
7	3 6 3 1 1 2	1.05×10^{-2}	4182	2.20×10^{-3}	66
8	6 5 5 5 4 5	8.73×10^{-10}	1300	2.34×10^{-9}	214
9	6 6 4 3 2 5	2.55×10^{-5}	844	1.05×10^{-5}	135
10	6 5 3 2 2 2	7.85×10^{-5}	653	3.11×10^{-5}	106
11	4 2 2 2 1 2	5.17×10^{-3}	448	1.04×10^{-3}	72
12	4 3 3 2 2 2	7.93×10^{-5}	629	3.22×10^{-5}	103
13	6 6 3 2 2 3	5.64×10^{-5}	684	2.26×10^{-5}	110
14	6 6 4 4 4 3	1.09×10^{-7}	1150	1.09×10^{-7}	190
15	2 2 1 1 1 1	2.17×10^{-2}	299	5.15×10^{-3}	49
16	6 6 6 6 5 6	1.22×10^{-10}	1580	1.07×10^{-9}	261
17	**6 6 6 6 6 6**	1.19×10^{-10}	1730	1.07×10^{-9}	286
18	6 6 5 5 5 5	1.44×10^{-10}	1450	1.12×10^{-9}	240
19	6 6 5 4 3 5	1.31×10^{-7}	1090	1.47×10^{-7}	178
20	6 6 6 5 5 6	1.28×10^{-10}	1510	1.08×10^{-9}	249
21	**1 1 1 1 1 1**	4.49×10^{-2}	287	1.88×10^{-2}	47
22	6 6 6 6 6 5	1.21×10^{-10}	1700	1.07×10^{-9}	283
23	6 6 5 4 3 4	1.31×10^{-7}	1070	1.48×10^{-7}	174
24	6 6 5 5 4 5	8.13×10^{-10}	1310	2.11×10^{-9}	215
25	6 6 6 5 5 5	1.31×10^{-10}	1490	1.09×10^{-9}	246

The bold values denote the both ends of Pareto front, which can be considered as the best solution of two conflicting objectives separately.

From Table 5 we can see that the results obtained include the solution for the highest availability {6, 6, 6, 6, 6, 6}, and the solution for the lowest cost {1, 1, 1, 1, 1, 1}, and the solutions obtained are all nondominated for both expected value and variance of objectives. The decision maker can select a design solution from the results according to his/her preference and the design requirements of system. For instance, if the system availability is required greater than 0.993, considering the trade-off between availability and cost, then the solution 11 may be selected; while if the system availability is required greater than 0.9993, the solutions 6, 9,10,12, 13 may be selected. In addition, if the design cost of system is under constraint, for example, if the maximum design cost is set as 1500 units, then the solutions 16, 17, 20, 22 should be eliminated.

6. Conclusions

In this paper, the UMRAP is studied based on uncertainty theory originally, and a solution approach is provided to obtain a desirable Pareto efficient set. Furthermore, an efficient multiobjective optimization algorithm based on combining some aspects of NSGA-II and ABC algorithm is proposed to solve UMRAP, called MOABC. The proposed MOABC has been tested on three well-known test problems and a multiobjective RAP in literature, the results of which show that MOABC outperforms NSGA-II greatly. Its convergence rate is fast and the estimated front it obtained can be very close to the true front. Finally, an application study is presented and solved by MOABC. Results suggest that UMRAP is much closer to real life, and the MOABC is an efficient multiobjective optimizer.

Acknowledgements

This work was supported by National Natural Science Foundation of China (No. 71171199) and Natural Science Foundation of Shaanxi Province of China (No. 2013JM1003).

References

1. Coit DW, Smith AE. Reliability optimization of series-parallel systems using a genetic algorithm. *IEEE Trans Reliab* 1996;**45**(2):254–60.
2. Billionnet A. Redundancy allocation for series-parallel systems using integer linear programming. *IEEE Trans Reliab* 2008;**57**(3):507–16.
3. Huang HZ, Qu J, Zuo MJ. Genetic-algorithm-based optimal apportionment of reliability and redundancy under multiple objectives. *IIE Trans* 2009;**41**(4):287–98.
4. Liang YC, Lo MH. Multi-objective redundancy allocation optimization using a variable neighborhood search algorithm. *J Heuristics* 2010;**16**(3):511–35.
5. Kaveh KD, Abtahi AR, Tavana M. A new multi-objective particle swarm optimization method for solving reliability redundancy allocation problems. *Reliab Eng Syst Safe* 2013;**111**:58–75.
6. Zadeh LA. Fuzzy sets. *Inform Control* 1965;**8**(3):338–53.
7. Verma AK, Srividya A, Gaonkar RSP. Fuzzy-reliability engineering: concepts and applications. New Delhi: Narosa; 2007. p. 13–6.
8. Utkin LV, Gurov SV. *New reliability models based on imprecise probabilities. Advanced signal processing technology by soft computing*; 2001. p. 110–39.
9. Garg H, Sharma SP. Multi-objective reliability-redundancy allocation problem using particle swarm optimization. *Comput Ind Eng* 2013;**64**(1):247–55.

10. Zadeh LA. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Set Syst* 1978;**1**(1):3–38.
11. Liu BD, Liu YK. Expected value of fuzzy variable and fuzzy expected value models. *IEEE Trans Fuzzy Syst* 2002;**10**(4):445–50.
12. Liu BD. *Uncertainty theory*. 4th ed. [Internet]. 2012 [cited 2013 Nov 18]. Available from: http://orsc.edu.cn/liu/ut.pdf.
13. Liu BD. Why is there a need for uncertainty theory? *J Uncertain Syst* 2012;**6**(1):3–10.
14. Liu BD. *Theory and practice of uncertain programming*. 2nd ed. Berlin: Springer-Verlag; 2009. p. 12–28.
15. Karaboga D. An idea based on honey bee swarm for numerical optimization. Turkey: Erciyes University; 2005. Report No.: Technical Report-TR06.
16. Rodriguez FJ, Carlos GM, Blum C, Lozano M. *An artificial bee colony algorithm for the unrelated parallel machines scheduling problem. Parallel Problem Solving from Nature-PPSN XII*. Berlin Heidelberg: Springer; 2012. p. 143–52.
17. Wang L, Zhou G, Xu Y, Wang SY, Liu M. An effective artificial bee colony algorithm for the flexible job-shop scheduling problem. *Int J Adv Manuf Technol* 2012;**60**(1–4):303–15.
18. Wang ZT, Guo JS, Zheng MF, Wang Y. A new hybrid intelligent algorithm for fuzzy multiobjective programming problem based on credibility theory. *Math Pro Eng* 2014;**2014**, 11 pages, Article ID: 909203, http://dx.doi.org/10.1155/2014/909203.
19. Karaboga D, Akay B. A comparative study of artificial bee colony algorithm. *Appl Math Comput* 2009;**214**(1):108–32.
20. Karaboga D, Basturk B. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm. *J Global Optim* 2007;**39**(3):459–71.
21. Karaboga D, Basturk B. On the performance of artificial bee colony (abc) algorithm. *Appl Soft Comput* 2008;**8**(1):687–97.
22. Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: Nsga-II. *IEEE Trans Evol Comput* 2002;**6**(2):182–97.
23. Elegbede C, Adjallah K. Availability allocation to repairable systems with genetic algorithms: a multi-objective formulation. *Reliab Eng Syst Safe* 2003;**82**(3):319–30.
24. Liu BD. *Uncertainty theory*. 2nd ed. Berlin: Springer-Verlag; 2007. p. 22–8.
25. Liu BD. Some research problems in uncertainty theory. *J Uncertain Syst* 2009;**3**(1):3–10.
26. Liu BD. *Uncertainty theory: a branch of mathematics for modeling human uncertainty*. 2nd ed. Berlin: Springer-Verlag; 2011. p. 18–26.
27. Chinchuluun A, Pardalos PM. A survey of recent developments in multiobjective optimization. *Ann Oper Res* 2007;**154**(1):29–50.
28. Liu BD. Matlab uncertainty toolbox. [Internet]. 2010 Feb 24 [cited 2013 Nov 18]. Available from: http://orsc.edu.cn/liu/resources.htm.
29. Zitzler E, Deb K, Thiele L. Comparison of multiobjective evolutionary algorithms: empirical results. *Evol Comput* 2000;**8**(2):173–95.

Guo Jiansheng is a professor at Air Force Engineering University. He received his B.S. degree from Air Force Engineering University in 1983. His area of research includes aviation engineering and information management system.

Wang Zutong is a doctor candidate at Air Force Engineering University. Her current research interests are combinatorial optimization.

Zheng Mingfa received the B.S. and M.S. degrees in mathematics from Hebei University in 2005 and 2008 respectively, and now he is a doctor candidate at Xi'an Jiaotong University. His main research interests are operations research in engineering.

Wang Ying is a professor at Air Force Engineering University. Her area of research includes aviation engineering and information management system.