Neurocomputing 142 (2014) 4–15

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Multi-objective new product development by complete Pareto front and ripple-spreading algorithm☆

Xiao-Bing Hu [a,c,*], Ming Wang [a], Qian Ye [a], Zhangang Han [b], Mark S. Leeson [c]

[a] State Key Laboratory of Earth Surface Processes and Resource Ecology, Beijing Normal University, China
[b] School of Systems Science, Beijing Normal University, China
[c] School of Engineering, University of Warwick, Coventry, UK

ARTICLE INFO

Article history:
Received 15 November 2013
Received in revised form
16 January 2014
Accepted 18 February 2014
Available online 9 May 2014

Keywords:
Multi-objective optimization
New product development
Pareto front
Ripple-spreading algorithm

ABSTRACT

Given several different new product development projects and limited resources, this paper is concerned with the optimal allocation of resources among the projects. This is clearly a multi-objective optimization problem (MOOP), because each new product development project has both a profit expectation and a loss expectation, and such expectations vary according to allocated resources. In such a case, the goal of multi-objective new product development (MONPD) is to maximize the profit expectation while minimizing the loss expectation. As is well known, Pareto optimality and the Pareto front are extremely important to resolve MOOPs. Unlike many other MOOP methods which provide only a single Pareto optimal solution or an approximation of the Pareto front, this paper reports a novel method to calculate the complete Pareto front for the MONPD. Some theoretical conditions and a ripple-spreading algorithm together play a crucial role in finding the complete Pareto front for the MONPD. Simulation results illustrate that the reported method, by calculating the complete Pareto front, can provide the best support to decision makers in the MONPD.

1. Introduction

New product development plays an extremely crucial role in company survival and success in the modern increasingly competitive global market; every year, billions of dollars are invested in various new product development projects (NPDPs) worldwide [1–5]. Obviously, not all NPDPs are successful, and there never lack examples where a big-brand company collapses after an NPDP because it misjudges market trends and/or consumes considerable of capital. To avoid such a tragedy, an effective practice is “not to put all eggs in one basket”. Therefore, a company may often have several NPDPs proceeding at one time. Each NPDP has both a profit expectation and a loss expectation, and such expectations vary according to the resources allocated to the NPDP. Basically, the greater the allocated resources the higher the profit expectation is. Increased allocated resources may reduce the failure possibility during the development stage of an NPDP, but cannot necessarily provide a better guarantee of market success. If anything goes wrong during the marketing stage due to many external, uncertain and uncontrollable factors, the larger resource allocation only means a bigger loss. Common sense in the financial sector predicts that a high profit expectation usually comes with a big loss expectation [6]. Therefore, decision makers often have to make a choice between high-profit-big-risk options and low-profit-small-risk options, based on their risk taking willingness and understanding of a market environment. Since available resources are always limited, decision makers usually need to optimize their investment portfolio, in order to maximize the profit expectation while minimizing the loss expectation – two conflicting objectives. In this paper, we are particularly concerned with the problem of allocating limited resources among several NPDPs, so that the overall profit expectation can be maximized while the overall loss expectation can be minimized. This clearly fits in the scope of a multi-objective optimization problem (MOOP), and hereafter we call the concerned problem *multi-objective new product development* (MONPD).

To resolve the MONPD, we need to make use of the Pareto front. As the most important concept in MOOPs, the Pareto front originates from the concept of Pareto efficiency proposed to study economic efficiency and income distribution [7]. In general MOOPs, a solution is called Pareto optimal if there exists no other solution that is better in terms of at least one objective and is not worse in terms of all other objectives [8,9]. The projection of a Pareto optimal solution in the objective space is called a Pareto point. All Pareto points, i.e., the projections of all Pareto optimal solutions, compose the complete Pareto front of an MOOP.

☆This work was supported in part by the National Basic Research Program of China under grant 2012CB955404, the project grant 2012-TDZY-21 from Beijing Normal University, China, and the Seventh Framework Programme (FP7) of the European Union under Grant PIOF-GA-2011-299725.

* Corresponding author.
E-mail address: drxiaobinghu@hotmail.co.uk (X.-B. Hu).

http://dx.doi.org/10.1016/j.neucom.2014.02.058
0925-2312/

The history of such problems is long resulting in the development of many methods for resolving various MOOPs. Basically, most methods can be classified into three categories: aggregate objective function (AOF) based methods [10–14], Pareto-compliant ranking (PCR) based methods [15–25], and constrained objective function (COF) based methods [26–30]. An AOF method combines all of the objectives of an MOOP to construct a single aggregate objective function, and then resolve the single-objective problem to get a Pareto optimal solution. However, it involves subjectiveness in constructing an AOF, and it often fails to find some Pareto optimal solutions if the Pareto front is not convex. A PCR method may overcome such drawbacks of AOF methods by operating on a pool of candidate solutions and favoring non-dominated solutions. Population-based evolutionary approaches (such as genetic algorithms, particle swarm optimization and ant colony optimization) often play a key role in PCR methods to identify multiple Pareto optimal candidate solutions. It should be noted that, due to the stochastic nature of PCR methods, their outputs are Pareto optimal candidate solutions, not necessarily real Pareto optimal solutions. Theoretically, COF methods, by optimizing only one single objective while treating all other objectives as extra constraints, may avoid both the subjectiveness of AOF methods and the loss of Pareto optimality in PCR methods.

Calculating complete Pareto front is a relatively less discussed topic in the study of MOOPs. Theoretically, some nonlinear AOF based methods can prove that for any Pareto point on the Pareto front a set of AOF coefficients definitely exists which can lead to that Pareto point. However, the difficulty is that there lacks a practicable method to find those sets of coefficients that will help to identify the complete Pareto front [28]. For PCR methods, guaranteeing the complete Pareto front is theoretically a mission impossible, largely because of the stochastic nature of employed population-based approaches [15]. COF methods, given well posed objective function constraints, may theoretically guarantee the finding of the complete Pareto front but like AOF methods, the practicality of finding proper constraints is a big issue [30]. Therefore, most existing methods can only produce an incomplete or approximate Pareto front [10,15,26–30]. In particular, as pointed out in [26], very few results are available on the quality of the approximation of the Pareto front for discrete MOOPs.

We have recently proposed a deterministic method which can, theoretically and practically, guarantee the finding of complete Pareto front for discrete MOOPs [31]. Some theoretical conditions and a general methodology were reported in [31], and a case study on a multi-objective route optimization problem (ROP) was used to prove the correctness and practicability. In this paper, we will particularly apply the method of [31] to the MONPD. Actually, there is a substantial body of literature on optimizing investment portfolios [6,32–38] similar to MONPD, but little work has been reported to calculate complete Pareto front of such investment portfolio optimization problems. To calculate the complete Pareto front for MONPD, firstly, we will improve the theoretical conditions and the methodology reported in [31]. The most challenging part in the method of [31] is to design an algorithm that is capable of finding the global kth best solution for any given k in terms of a given single objective. Designing such an algorithm is largely problem-dependent, and is often difficult because most optimization algorithms only calculate the global 1st best solution. MONPD is quite different from the ROP in [31]. For example, in the ROP, every objective needs to be minimized; however, in MONPD, the profit expectation needs to be maximized although the loss expectation is to be minimized. Therefore, MONPD demands a new algorithm to calculate the general kth *best* (rather than only the kth *smallest*) single-objective solution. By successfully developing a new ripple-spreading algorithm for MONPD, this paper will further prove the practicability and the potential of the methodology of resolving discrete MOOPs by calculating complete Pareto front.

The remainder of this paper is organized as following. Section 2 gives some theoretical results for calculating complete Pareto front for discrete MOOPs. Section 3 describes mathematically the details of MONPD. Section 4 reports a ripple-spreading algorithm for MONPD. Simulation results are given in Section 5, and the paper ends with some conclusions and discussions on future work in Section 6.

2. Theoretical results for calculating the complete Pareto front

We have recently reported some theoretical results and a general methodology to guarantee, theoretically and practicably, the finding of the complete Pareto front for discrete MOOPs [31]. The work in [31] is the theoretical foundation of this application paper. In this section, we will introduce some improvements to the work of [31], in order to better apply to MONPD later.

First of all, we need a general mathematical formulation of discrete MOOPs as following:

$$\min_x [g_1(x), g_2(x), \ldots, g_{N_{Obj}}(x)]^T, \tag{1}$$

subject to

$$h_I(x) \le 0, \tag{2}$$

$$h_E(x) = 0, \tag{3}$$

$$x \in \Omega_X, \tag{4}$$

where g_i is the ith objective function of the total N_{Obj} objective functions, h_I and h_E are the inequality and equality constraints, respectively, x is the vector of optimization or decision variables belonging to the set of Ω_X, and x is of discrete value. A Pareto-optimal solution x^* to the above problem is so that there exists no x that makes

$$g_i(x) \le g_i(x^*), \text{ for all } i = 1, \ldots, N_{Obj}, \tag{5}$$

$$g_j(x) < g_j(x^*), \text{ for at least one } j \in [1, \ldots, N_{\mathrm{Obj}}]. \tag{6}$$

The projection of such an x^* in the objective space is called a Pareto point. The above problem usually has a set of Pareto optimal solutions, whose projections compose the complete Pareto front.

2.1. Theoretical conditions

According to the theoretical results in [31], we have the following statements for discrete MOOPs.

Lemma 1. *Suppose we sort all discrete $x \in \Omega_X$ according to a certain objective function $g_j(x)$, and $x_{j,i}$ has the ith smallest g_j. For a given constant c, if there exists an index k that satisfies*

$$g_j(x_{j,k}) \le c < g_j(x_{j,k+1}), \tag{7}$$

then the number of Pareto points whose $g_j \le c$ is no more than k, and all the associated x values are included in the set $[x_{j,1},\ldots,x_{j,k}]$.

Lemma 2. *Suppose we have a constant vector $[c_1, \ldots, c_{N_{Obj}}]$, the element c_j is for objective function g_j, and after sorting all discrete $x \in \Omega_X$ according to each objective function g_j, we have k_j satisfying Condition (7). If for any $j = 1,\ldots,N_{Obj}$,*

$$g_i(x_{j,k_j}) \le g_i(x_{i,k_i}), \text{ for all } i \ne j, \tag{8}$$

then the total number of Pareto points is no more than

$$N_{PP} \le \sum_{j=1}^{N_{Obj}} k_j, \tag{9}$$

and all associated x values are included in the union set

$$\Omega_{U1} = \cup_{j=1}^{N_{Obj}} [x_{j,1}, \ldots, x_{j,k_j}]. \tag{10}$$