



Multi-period project portfolio selection under risk considerations and stochastic income

Ali Asghar Tofighian¹ · Hamid Moezzi² · Morteza Khakzar Barfuei³ · Mahmood Shafiee⁴

Received: 3 May 2017 / Accepted: 4 October 2017 / Published online: 2 February 2018
© The Author(s) 2018. This article is an open access publication

Abstract

This paper deals with multi-period project portfolio selection problem. In this problem, the available budget is invested on the best portfolio of projects in each period such that the net profit is maximized. We also consider more realistic assumptions to cover wider range of applications than those reported in previous studies. A novel mathematical model is presented to solve the problem, considering risks, stochastic incomes, and possibility of investing extra budget in each time period. Due to the complexity of the problem, an effective meta-heuristic method hybridized with a local search procedure is presented to solve the problem. The algorithm is based on genetic algorithm (GA), which is a prominent method to solve this type of problems. The GA is enhanced by a new solution representation and well selected operators. It also is hybridized with a local search mechanism to gain better solution in shorter time. The performance of the proposed algorithm is then compared with well-known algorithms, like basic genetic algorithm (GA), particle swarm optimization (PSO), and electromagnetism-like algorithm (EM-like) by means of some prominent indicators. The computation results show the superiority of the proposed algorithm in terms of accuracy, robustness and computation time. At last, the proposed algorithm is wisely combined with PSO to improve the computing time considerably.

Keywords Portfolio selection · Risk analysis · Investment · Genetic algorithm · Particle swarm optimization · Project interdependency

Introduction

Project portfolio selection (PPS) is one of the most important decision-making problems for most organisations in project management and engineering management (Lean et al. 2012). This problem includes selection of optimum portfolio of projects among a range of available projects which are subject to a number of enterprises' intrinsic constraints such as budget, available resources, as

well as some extrinsic and technical limitations of the real world (Tofighian and Naderi 2015). As defined by the Project Management Institute (PMI) (2008), a portfolio is a collection of projects or programs grouped together to facilitate effective management of work to meet strategic business objectives. The projects or programs of the portfolio may not necessarily be interdependent or directly related.

Recently, PPS has become one of the most active research topics in the fields of economic analysis (e.g., see Tofighian and Naderi 2015; Lee et al. 2006; Wu and Chen 2015), R&D projects (e.g., see Fang et al. 2008; Bhattacharyya et al. 2011; Hassanzadeh et al. 2014), supplier selection (e.g., see Hosseinasab and Ahmadi 2015; Vazhayil and Balasubramanian 2014; Lorca and Prina 2014), etc. Any model to solve this problem should consider relations between projects, uncertainties associated with incomes and risk issues so that the obtained results to be more valid. A retrospect of the literature reveals that numerous researches have taken into account the

✉ Mahmood Shafiee
m.shafiee@cranfield.ac.uk

¹ Industrial Engineering Department, Faculty of Engineering, Kharazmi University, Tehran, Iran

² Industrial Engineering Department, Faculty of Engineering, University of Science and Culture, Tehran, Iran

³ Industrial Engineering Research Group, Technology Development Institute, ACECR, Tehran, Iran

⁴ Cranfield University, College Road, Cranfield, Bedfordshire MK43 0AL, UK

uncertainty and risk issues by stochastic programming and fuzzy programming. However, the cost dependency between projects is often disregarded.

In this paper, a novel formulation of the PPS problem is developed taking into account cost dependencies as presented in [Golmohammadi and Pajoutan \(2011\)](#). In addition to the existing features, some others such as multi-periods and possible investment of extra budget for each period in financial institutions are also included. The main objective of the proposed model is to maximize the net profit earned from investing available budget. Due to the complexity of the problem, no analytical method could be established to approximate a global optimal solution in a reasonable length of time. To overcome this challenge, a new genetic-based algorithm with new solution representation and operators is developed. The performance of the solution approach is then compared with a well-known algorithm, called particle swarm optimization (PSO) by means of four prominent indicators, namely: the mean gained fitness, standard deviation, relative percentage deviation and run time. The computations performed show that the proposed solution approach can substantially increase the accuracy and robustness of the results.

The rest of the paper is organized as follows: Sect. 2 presents the background and a brief overview on project portfolio selection problem. In Sect. 3, a mathematical model is presented to solve the problem. Section 4 describes the proposed algorithm and solution approaches developed based on three well-known meta-heuristics, namely: genetic algorithm (GA), particle swarm optimization (PSO) and EM-like algorithms. The calibration of these methods, computational results and the comparison between them are presented in Sect. 5. Finally the paper is concluded in Sect. 6.

Literature review

The widespread use of portfolio selection models in real-world situations has led to extensive studies in this area in recent years. For an accurate modelling of the problem, all real-world conditions and features should be considered. Uncertainty and risk are two substantial components of real-world conditions, and hence, several number of research papers consider these two elements, irrespective of the application of the study. For example, [Kocadağlı and Keskin \(2015\)](#) asserted that the risk-return trade-off is the main concern of financial theory and proposed a new portfolio selection model based on fuzzy goal programming techniques that incorporated the risk issues of investor and market trend. [Huang \(2008\)](#) provided a new definition of risk in the field of portfolio selection and developed a new model. In this study, a hybrid intelligent

algorithm based on GA was also proposed and the effectiveness of the algorithm for solving the model was evaluated. [Hosseininasab and Ahmadi \(2015\)](#) explored one of the newest fields of PPS and proposed a new two phase supplier selection model that considers risks. [Li et al. \(2015\)](#) developed a fuzzy portfolio selection model considering background risks which may affect investors' decisions.

In addition to risk and uncertainty, considering dependencies between projects has an important role to model real-world cases. [Rebiasz \(2013\)](#) presented a new method for selecting investment projects in a fuzzy environment. It was concluded that economic dependencies between projects significantly affect their effectiveness and risk. [Mathuria et al. \(2015\)](#) proposed a new framework aiming to improve the profit-risk trade-off portfolio optimization of a power generation company. In another study, [Lorca and Prina \(2014\)](#) used PPS approach to select the best power producer in a competitive electricity market while considering locational electricity prices and risk management such that the expected profit of the company was maximised. [Vazhayil and Balasubramanian \(2014\)](#) developed a multi-objective model to select the optimal electricity generation portfolio for India's 12th 5-year plan with taking into account risks and barriers. Then, they used an Intelligent Pareto search Genetic Algorithm (IPGA) to solve the model.

Despite all the above-mentioned studies on the risk and uncertainty issues, only a few research contemplate cost dependencies between projects. [Golmohammadi and Pajoutan \(2011\)](#) developed a PPS model with taking into account dependency between projects, stochastic revenue and risk. Two meta-heuristic algorithms based on GA and electromagnetism-like (EM-like) were proposed to solve the problem. The performance of the two algorithms was compared together and it was found out that the GA performed better than the EM-like algorithm. As noticed from the literature reviewed above, the main focus of the solution method has been so far on the meta-heuristic algorithms because of the NP-hard nature of the problem. GA and PSO are widely used throughout the literature for solving NP-hard complex problems. In addition, there are some good examples of using GA and PSO for PPS problem-solving. [Guang-Feng et al. \(2012\)](#) presented a PSO algorithm for solving the cardinality constrained Markowitz portfolio optimization problem. They compared this meta-heuristic method with GA, SA (simulated annealing) and TS (Tabu search) and showed that PSO performed better in most cases. [Zhu et al. \(2011\)](#) proposed a PSO-based algorithm to solve non-linear constrained portfolio optimization problem with multi-objective functions. The results from PSO were compared with those obtained from GA and VBA (Virtual Bee Algorithm) and it



was shown that PSO results were comparable with or superior than GA and VBA. According to the previous studies, application of GA and PSO in portfolio selection problem and further examining their performance can be promising.

Problem definition

Project portfolio selection is one of the most recent fields of study and research in project management. Project portfolio management is widely applied to utilise resources for maximum profit. Until recently, several researches have been conducted to shape these models into reality, but only a few of them have considered the cost dependency between projects. As mentioned earlier, the formulations by [Golmohammadi and Pajoutan \(2011\)](#) will be further extended in this study to achieve the maximum benefit at the end of the last period of the project. In each period, there are a number of available projects and a possibility of bank investments which can be chosen by decision-makers (henceforth referred to as DM). In many cases, the balance of the budget (after selecting projects) is less than the minimum financial needs of the unselected projects. Thus, the surplus of the budget can be invested in banks for more benefits. The net benefit of the first period will be considered as the budget of the next period. This sequence will continue to the end of last period (T). Even though the costs and expected incomes of the projects at the beginning of the first period are already known to the organisation, the value of money changes over time. Hence, the time value of money is also considered in this study. In several investments, particularly in R&D project portfolio selections, finding the deterministic value for incomes is very difficult and tends to be inaccurate. So, this study considers incomes as independent normally distributed stochastic variables. However, the costs are considered deterministic because the resources needed and subsequent costs for each project are specified. In real-world cases projects that have relations like time relations or monetary/financial relations among them are common. In majority of the cases, these relations are synergic and may reduce costs. For instance, assume that the projects i and m are very similar in terms of the resources, techniques and a some other aspects. If the project i is selected in the period j , the associated costs of the project m in later periods will be much less, because of increase in the level of knowledge and experiences, etc. Also, the project m will be more straightforward to perform compared to the project i . For this reason, our paper also considers dependencies and relations between projects that affect costs. Because of the nature of stochastic incomes, risk must be primary in the chance-constrained approach as what was proposed in [Charnes and Cooper \(1962\)](#).

Characteristics

Some specific assumptions of this study are described as below:

If a project starts, it must be continued and finished without any pause or break.

1. The whole financing process is self-financing, which means all finances should be inserted in the selected project/projects at once, and no extra money will be financed thereafter.
2. The revenues of the projects are independent normally distributed stochastic variables.
3. The investment in banks has a deterministic revenue.
4. The risk is considered in each period.
5. A deterministic budget is considered for all projects.
6. Costs of each project are deterministic and dependent on next and previous periods.

Regarding these characteristics, in Sects. 3.2 and 3.3, the parameters, indexes and decision variables are introduced and in Sect. 3.4 our mathematical model is presented.

Parameters and indexes

n	Total number of available competitive projects
N	Set of all available projects $N = \{1, \dots, n\}$
T	Total number of periods
j	index for time periods $j \in \{1, \dots, T\}$
N_j	Set of existing projects in j -th period $N_j \subseteq N$
i, m	Indexes for the projects $i, m \in N$
z_{ji}	Cost of i -th project in j -th period without taking account the time value of money
c_{ji}	Cost of i -th project in j -th period with respect to the time value of money
r_{ji}	Revenue of project i in j -th period with respect to the time value of money
v_{ji}	Revenue of project i in j -th period without taking account the time value of money
σ_{ji}	Standard deviation of the revenue of project i in j -th period with respect to the time value of money
σ_{ji}	Standard deviation of the revenue of project i in j -th period without taking account the time value of money
s_j	Income from investments in banks in j -th period
t_j	Amount of investments in banks in j -th period
τ_{jim}	Cost coefficient of dependent project i in j -th period and project m in period $j + 1$
RC_j	Minimum acceptable revenue of projects in j -th period which is accepted by the decision maker
α	Maximum acceptable risk for earning at least RC_j
β	Total available budget
$Rate$	Annual interest rate
b	Interest rate of investing in bank



Decision variables

x_{ji} 1 if i -th project selected in j -th periods; otherwise 0
 y_j 1 if investment in bank made in j -th period; otherwise 0

Formulation

$$\max z = \sum_{j=1}^T \left(s_j y_j - t_j y_j + \sum_{i \in N_j} (E(r_{ji} x_{ji}) - c_{ji} x_{ji}) \right) + \sum_{j=1}^{T-1} \sum_{i \in N_j} \left(\sum_{m \in N_{j+1}} \tau_{jim} x_{ji} c_{j+1m} x_{j+1m} \right), \quad (1)$$

subject to:

$$P \left(\sum_{i \in N_j} r_{ji} x_{ji} \leq RC_j \right) \leq \alpha \quad \forall j, \quad (2)$$

$$\beta = t_1 y_1 + \sum_{i \in N_1} c_{1i} x_{1i} \quad (3)$$

$$s_j = t_j (1 + b) \quad \forall j, \quad (4)$$

$$c_{ji} = z_{ji} \times (1 + \text{Rate})^{j-1} \quad \forall i, j, \quad (5)$$

$$E(r_{ji}) = E(z_{ji} \times (1 + \text{Rate})^{j-1}) \quad \forall i, j, \quad (6)$$

$$\sigma_{ji} = \sigma'_{ji} \times (1 + \text{Rate})^{j-1} \quad \forall i, j, \quad (7)$$

$$t_{j+1} y_{j+1} + \sum_{m \in N_{j+1}} \left(c_{j+1m} x_{j+1m} - \sum_{i \in N_j} \tau_{jim} x_{ji} c_{j+1m} x_{j+1m} \right) = s_j y_j + \sum_{i \in N_j} r_{ji} x_{ji} \quad (8)$$

$$\forall j \in \{T\},$$

$$\sum_{i \in N \setminus N_j} x_{ji} = 0 \quad \forall j, \quad (9)$$

$$x_{ji} \in \{0, 1\} \quad \forall j, i, \quad (10)$$

$$y_j \in \{0, 1\} \quad \forall j. \quad (11)$$

Equation (1) represents an objective function consisting of two parts. The first part $\sum_{j=1}^T (s_j y_j - t_j y_j + \sum_{i \in N_j} (E(r_{ji} x_{ji}) - c_{ji} x_{ji}))$ maximises the net profit by selecting the projects as well as the investments in banks. The second part $\sum_{j=1}^{T-1} \sum_{i \in N_j} (\sum_{m \in N_{j+1}} \tau_{jim} x_{ji} c_{j+1m} x_{j+1m})$ includes all costs reduced due to project relations and dependencies. Constraint (2) represents risk, which assures that the probability of obtained revenue in each period is less than minimum acceptable revenue of the projects in that period. It should also be lower than α . An attempt to

rewrite this constraint with respect to the ration of the project costs in j -th period is shown below:

$$P \left(\sum_{i \in N_j} r_{ji} x_{ji} \leq (1 + q) \times \sum_{i \in N_j} \left(c_{ij} x_{ij} - \left(\sum_{m \in N_{j+1}} \tau_{jim} x_{ji} c_{j+1m} x_{j+1m} \right) \right) \right) \leq 1 - \alpha; \quad (12)$$

where q represents the minimum rate of return (ROR) accepted by the DM. It means DM will not accept the risk of selecting projects if the gained revenue is lower than the cost of selected projects considering minimum acceptable ROR. Constraint (3) reflects the limits on the budget for the first period. Furthermore, all available budgets will be invested since the remainder of the budget after project selection can be entirely invested in banks. Hence, Constraint (3) is formulated as Equality. In this constraint, t_1 is the difference between the initial budget and the summation of costs for selected projects in the first period. Constraint (4) represents the total income generated from investments in banks for period j . Constraints (5), (6) and (7) calculate future values of costs, expected revenues and standard deviation of revenues for period j . Constraint (8) assures that costs in period $j + 1$ are less than all incomes in period j . It also assigns these incomes as the budget of next period. Constraint (9) ensures that in each period unavailable project could not be selected. Constraints (10) and (11) define the decision variables.

Illustrative example

In what follows, a numerical example is provided to demonstrate different aspects of the problem and the model presented. We assume that five projects are available in each period and there are three periods in planning horizon (see Fig. 1. Total available budget (β), minimum accepted rate of return (q), interest rate of investing in bank (b) and annual interest rate (Rate) are 919.5, 10, 5 and 6%, respectively.

The cost coefficients of dependent projects (τ_{jim}) are presented in Table 1.

As shown in Table 1, if project 2 is selected in period 1 the cost of project 4 in period 2 will decrease by 5.92%. The part A in Table 1 shows that available projects in each period may be dependent on the projects in the consecutive period and part B indicates that there is no dependency between the projects in each period.

This illustrative example was solved with Lingo 10 and the results are shown in Fig. 2. As can be seen, projects 4 and 5 are selected in the first and the second periods, whereas, in the third period, the projects 2, 4 and 5 are selected. Net profit value for this solution is estimated to be 965.89.

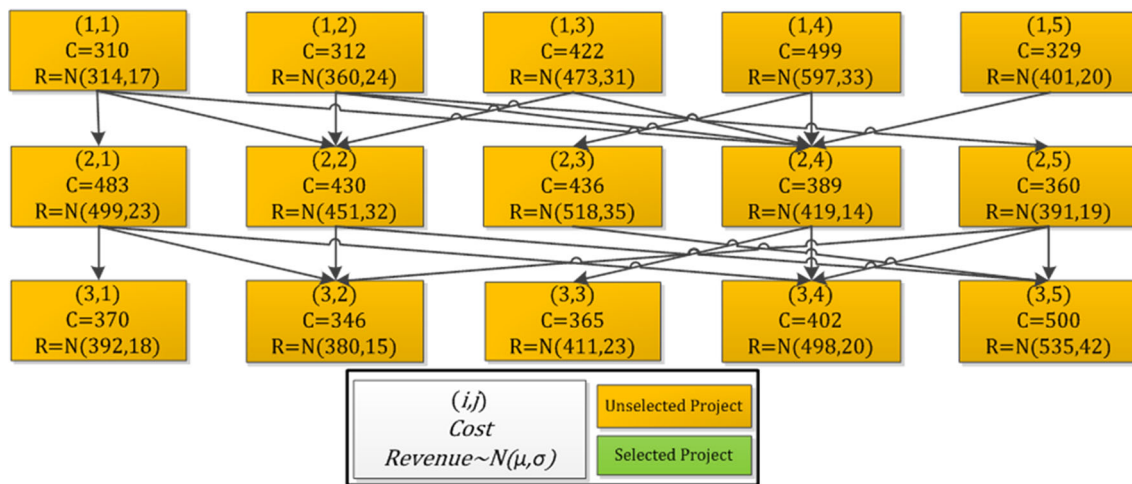


Fig. 1 Available projects in each period

Table 1 Cost coefficients of dependent projects (τ_{jim})

	2,1	2,2	2,3	2,4	2,5	3,1	3,2	3,3	3,4	3,5
1,1	0.3071	0.2259	0	0.3219	0	A				
1,2	0	0.0262	0	0.0592*	0.3104					
1,3	0	0.3164	0	0.3090	0					
1,4	0	0	0.0103	0.0157	0					
1,5	0	0	0	0.3004	0					
2,1	B					0.0655	0.3048	0	0.1252	0
2,2						0	0.0890	0	0	0.3197
2,3						0	0	0	0	0.1877
2,4						0	0	0.0980	0.1494	0
2,5						0	0.1398	0	0.2509	0.2493

Following these results, an examination of the solution was carried out. If projects 4 and 5 are selected in the first period, the summation of costs will be $449 + 329 = 778$. With respect to the initial budget 919.5, it is evident that the budget is not violated and the remainder i.e., $t_1 = 919.5 - 778 = 91.5$ will be invested in a bank. Risk constraint is also satisfied because $P\left(z < \frac{(1+0.1) \times (499+329) - 597 - 401}{\sqrt{33^2 + 20^2}}\right) = 0.012$ and it is far below 5%. Net profit of the first period is 1094.075, which is obtained from investing in bank ($91.5 \times (1 + 0.05) = 96.075$) and expected revenue of projects 4 and 5 ($597 + 401 = 998$). Additionally, net profit of the first period is considered as the budget of the second period. In context with the project dependencies and time value of money, the summation of costs of selected projects will be $[389(1.06) + 360(1.06)] -$

$[0.015 \times 412.34 + 0.3 \times 412.34] = 663.6$ and the invested budget in bank is $t_2 = 430.4$. There is no violation of the risk constraint, because $P\left(z < \frac{743.6 - (419+391)(1.06)}{\sqrt{(14 \times 1.06)^2 + (19 \times 1.06)^2}}\right) < 0.05$ and net profit for second period is $(419 + 391)(1.06) + 430.47(1.05) = 1310.59$. Calculations in the third period are same as that of the second period.

Proposed meta-heuristic algorithm

The project portfolio selection problem is a non-deterministic polynomial-time hard (NP-hard) problem (Doerner et al. 2006) which could be solved by meta-heuristic

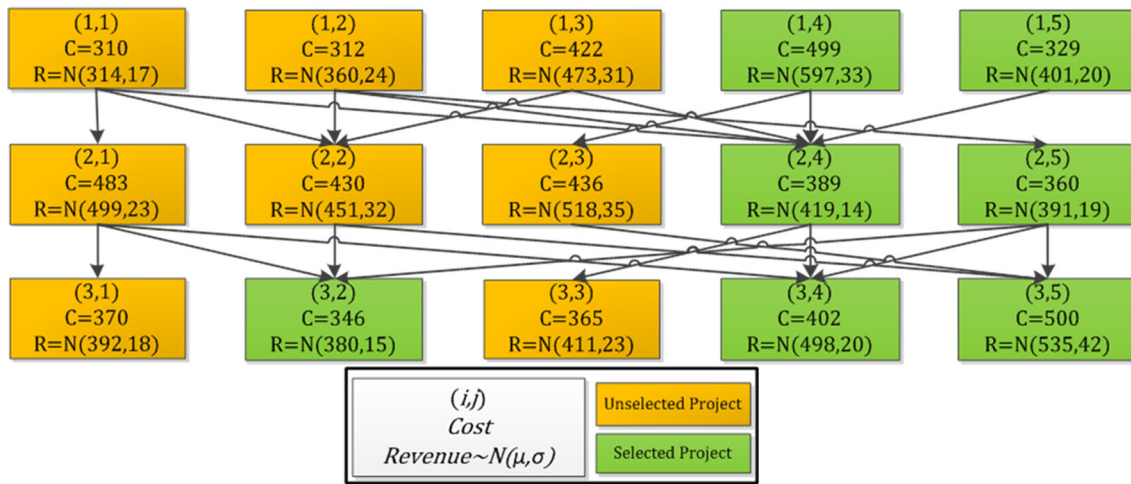


Fig. 2 Lingo solution for the numerical example

algorithms. As mentioned earlier, three prominent meta-heuristic algorithms in the literature are: genetic algorithm (GA), particle swarm optimization (PSO) and EM-like algorithm. So after describing the proposed algorithm, we calibrate all these methods and solve the problem with them.

Genetic algorithm

Genetic algorithms (GAs) are population-based meta-heuristics, which are widely used to solve combinatorial problems. In the last decade, GAs have been used to solve countless problems and found to be an effective and robust search method. In this paper, a GA-based solution approach to solve the PSS model is presented, which is shown in Fig. 3.

The first population is generated using generation and evaluation mechanisms which will be discussed later. Each population includes agents called individuals that represent chromosomes in GAs. Encoding of these chromosomes is also discussed in Sect. 4.1.1. After evaluating individuals, they are sorted and the fittest is selected as the global best solution. The main loop then begins at this point. Three types of offspring will be generated; two of them would be generated from current population or parents by the means of a classical GA operator called Crossover and mutation. For the sake of diversity, a portion of brand new offspring are also generated and to make better solutions of current ones, also a local search mechanism is used which is discussed in Sect. 4.1.4. All offspring and parents are evaluated and sorted, and subsequently deleting the worst extra individuals. This pruned generation is considered as current population. The best solution for this generation is compared with the global best solution, and the best amongst

them is further considered as the new global best solution. This process will continue until the termination criterion is met.

Encoding scheme

The key issue in using GAs is encoding a solution into a chromosome. This encoding is used to formulate recognisable solutions for computers. A binary matrix $T \times \max(n_j)$ to encode the solutions is proposed. Each row and column represents one period and one project, respectively. If the element of row i and column j is one, it means that project i in period j is selected. For instance, solution shown in Fig. 2 is decoded as below:

$$\begin{bmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \end{bmatrix}_{3 \times 5}$$

Population generation and evaluation mechanisms

The characteristic of the problem is that the generation mechanism is a step-by-step mechanism. This means that the solutions for the first period are generated first and then a solution of the second period is generated based on the first period’s solutions and so on. In each step, the feasibility of the solution is checked against the budget and risk constraints (Fig. 4). This mechanism is shown in Fig. 5.

As budget of each period is calculated simultaneously with the population generation mechanism, evaluating mechanism becomes straight forward. This mechanism is shown in Fig. 6.

Fig. 3 The proposed genetic algorithm

```

Procedure: The proposed genetic algorithm
iteration ← 1
Population generation mechanism
Evaluation mechanism
Sort Population
Find Global BestSolution
Repeat until termination criterion is not met
{
OffSpring1 ← Cross over mechanism
OffSpring2 ← Mutation mechanism
OffSpring3 ← Population generation mechanism
cumulative Population ← {Population ∪ OffSpring1 ∪ OffSpring2 ∪ OffSpring3}
Evaluate cumulative Population
Sort cumulative Population
Omit extra individuals in cumulative Population
Population ← cumulative Population
Find BestSolution
If BestSolution is better than GlobalBestSolution then
{
Global BestSolution ← BestSolution
}
iteration ← iteration + 1
}

```

Fig. 4 The population generation mechanism

```

Procedure: Population generation mechanism
For all individuals
{
Assign initial budget as the budget of the first period
Randomly generate a binary matrix (1 × max(nj)) as Solution1
period ← 1
For all periods
{
Repeat until cost of current period doesn't violate budget of current period AND
risk constraint satisfied
{
Randomly generate a binary matrix (1 × max(nj)) as the solution matrix of current
period
}
Calculate t and s of current period and budget of next period
period ← period + 1
}
individual(i) ← Solution
}

```

GA's operators and offspring

As mentioned earlier, in order to generate new population, the parents and all three types of offspring are used. Type 1 of these offspring are those which are generated by the crossover operator. Two parents are selected by means of tournament selection (see Sect. 5.2). Later, a random binary matrix with the same size as that of the parents is generated as a mask matrix. Elements with value 0 of the mask matrix are selected from first parent and elements with value 1 are selected from the second. If the solution is not feasible, the mask matrix will keep on changing until a feasible solution is reached. Crossover mechanism is shown in Fig. 7.

Type 2 of offspring are those generated by the mutation operator. First, a row and a column of parent matrix are selected. Then, if the element is 1, it will change to 0 and vice versa. The feasibility of this process is checked and if

the child is not feasible, another row and column will be selected. Figure 8 shows this mechanism.

The last types of offspring are completely new. This type of offspring is generated for the sake of diversity and to avoid sticking to the local optimum. This is generated by the means of population generation mechanism, which was explained earlier. The numbers for each one of these types are predetermined and will be discussed in Sect. 5.2.

Local search mechanism

To enhance the accuracy of the optimal solution found, we apply a local search mechanism. Figure 8 shows pseudo code of this mechanism. We select one of current solutions by the means of tournament selection. It helps us to select better solutions with higher chance. This mechanism starts from very first period and try to add as many projects as it



Fig. 5 The population evaluation mechanism

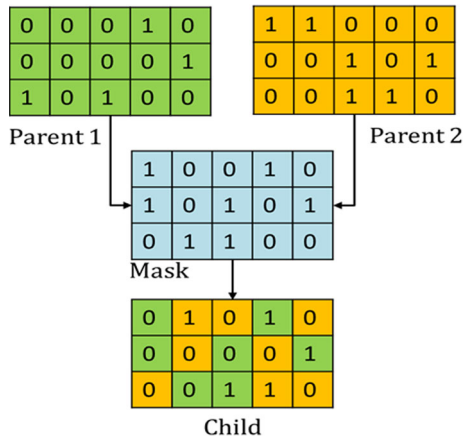


Fig. 6 The crossover mechanism

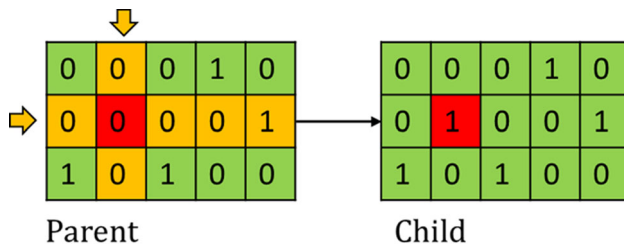


Fig. 7 The mutation mechanism

can to the selected projects. First it makes a blacklist filled with selected projects. Next it checks if any projects can be added to this period or not (if all available projects are selected then blacklist is full and no other projects can be added). Then it tries to add project to each period. If more than 50% of projects are not selected it is a good sign that shows we can enhance current period so number of tries is calculated as random number between 50% of the projects to maximum number of tries, otherwise it can be anything from one to maximum number of tries (see Sect. 5.2). Afterwards, in the “while-loop” an available project is randomly selected and added to the list of projects. Then, mechanism checks the feasibility of new solution, if it is feasible, mechanism updates solution, budget and adds selected project to the blacklist; otherwise it adds selected project to blacklist and increments try variable by one.

Computational evaluation

This section gives a description of the computational evaluations. It also compares proposed algorithm with GA, PSO and EM-like on different test data sets. These

Procedure: Population evaluation mechanism

$Fitness\ of\ each\ individual \leftarrow (budget\ of\ period\ T + 1) - (initial\ budget)$

algorithms are coded in MATLAB 7.12 and executed on a laptop computer with Core i7, and Windows 7 using 4 GB of RAM.

Test data generation

Although, in reality, a problem with more than 15 available projects is rarely faced, in this study to gain a more accurate assessment of the proposed algorithms large-scale problems are considered. Table 2 shows problem parameters and the strategy for generating their characteristics.

Furthermore, Table 2 shows that the number of available projects is a uniform random number between 5 and 70. Due to fluctuations in *Rate*, the number of periods is considered to vary between 3 and 12. Since in real-world situations, revenue of fulfilling each project is usually bigger than its cost, r_{ji} is considered to be up to 1.5 times more than cost. Other parameters are calculated based on the data derived from Table 2.

Parameter calibration

Appropriate design of the parameters and operators significantly improves the effectiveness of a meta-heuristic algorithm. In this section, we study the behaviour of the proposed algorithms regarding different operators and parameters. Among different DOE methods, Taguchi approach is one of the most prominent and suitable methods as it does not need full factorial trials. In this approach, orthogonal arrays are used to study numerous decision variables with a limited number of trails. The responses of these trials are converted to the signal-to-noise (S/N) ratio. The following definition for *S/N* is used for a maximisation problem:

$$\frac{S}{N} = -10 \log \frac{1}{n} \sum_i \frac{1}{F_i^2} \quad (13)$$

where F_i is the mean value of fitness function and n is the number of trials. In this paper, eight control factors are included: the number of population (PN), mutation rate (MR), crossover rate (CR), crossover method (CT), brand new population (NP), Local search rate (LS), tries rate (TR), parent selection method (PS). Table 3 depicts the levels of these factors. The orthogonal array L_9 is chosen because it meets all minimum requirements (Fig. 9). This array is presented in Table 4 Ten different problems with different sizes are generated and each experiment is performed three times. With respect to the orthogonal array

Fig. 8 Local search mechanism

```

Procedure: Local search mechanism
Sol ← Select from current population based on roulette wheel selection
Current budget ← β
For each period {
  Blacklist ← {selected project of current period}
  maxTry ← number of available projects in this period – size of black list
  If (maxTry > 0) {
    tempSol ← Sol
    If (maxTry > n/2) { terminationTry ← random number between n/2 and maxTry}
    else { terminationTry ← random number between 1 and maxTry}
    While (try < terminationTry) {
      randomly select a non-selected project in this period and add to tempSol
      If (feasible) {update Sol, Current budget of next period}
      else {try ← try +1}
      Add the project to blacklist
    } //end while
  } // end if
} //end for each
    
```

Table 2 Problem generating strategy

Parameter	Small size	Medium size	Large size
n	$U(5, 25)$	$U(26, 50)$	$U(51, 70)$
T	$U(3, 7)$	$U(8, 12)$	$U(13, 20)$
α	5%	5%	5%
b	{5%, 10%, 15%, 20%}	{5%, 10%, 15%, 20%}	{5%, 10%, 15%, 20%}
z_{ji}	$U(300, 500)$	$U(300, 500)$	$U(300, 500)$
r_{ji}	$U(1, 1.5) \times z_{ji}$	$U(1, 1.5) \times z_{ji}$	$U(1, 1.5) \times z_{ji}$
τ_{jim}	$U(0, 1)$	$U(0, 1)$	$U(0, 1)$
Rate	$b \pm U(0.01, 0.02)$	$b \pm U(0.01, 0.02)$	$b \pm U(0.01, 0.02)$
β	$U(0.4, 0.65) \times \sum_i z_{1i}$	$U(0.4, 0.65) \times \sum_i z_{1i}$	$U(0.4, 0.65) \times \sum_i z_{1i}$
σ_{ji}	$U(0.03, 0.08)$	$U(0.03, 0.08)$	$U(0.03, 0.08)$
q	$0.1 + 0.5 \times U(0, 1) \times \text{Rate}$	$0.1 + 0.5 \times U(0, 1) \times \text{Rate}$	$0.1 + 0.5 \times U(0, 1) \times \text{Rate}$

L_9 , the total performing number is $10 \times 3 \times 9 = 270$. After performing the experiments, fitness values are individually transformed into S/N ration. Figure 9 shows the average S/N ratio that is obtained at each level.

Based on Fig. 10 and Table 5 the best level for each parameter is set as follows: PN = 30, MU = 0.15, CR = 70, NP = 0.2, LS = 0.6, TR = 03, CT = mask matrix and PS = tournament selection. To assess the impact of each factor on the performance of proposed algorithm, the delta test is used.

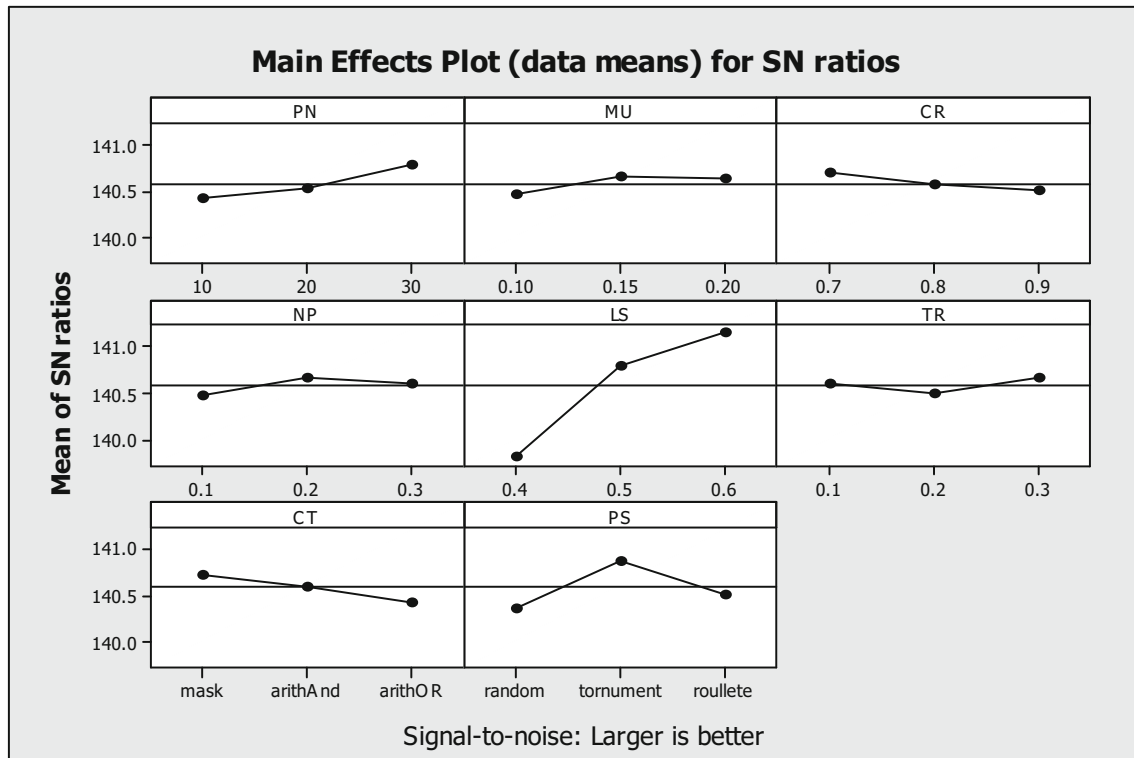
Table 5 shows levels and delta values obtained by each factor. The most effective factor is the local search usage rate, while number of tries has the least impact on the solutions. The calculation of GA, EM-like and PSO’s parameter calibration are not mentioned, but by utilising same methodology and state of art, the best values for each algorithm and associated factors are shown in Table 6.

Results

This section compares the results obtained from the proposed algorithm with the results obtained from three other algorithms. According to prior knowledge, 150 instances are generated (50 instances for each size of problems) and each one is solved 10 times. For the sake of brevity, 82 instances (25, 25 and 32 for small, medium and large sizes) are selected. The results are examined based on four criteria: mean, standard deviation, mean RPD (relative percentage deviation) and run time. Mean value indicates that how much an algorithm is better than the others based on the quality of solutions. Variance is used to find the algorithm having higher precision. Mean RPD is used to find out which algorithm produces results, nearest to best found solution among all algorithms. The RPD is calculated as follows:

Table 3 Proposed algorithm's factors and levels

Level	PN	MU	CR	CT	NP	LS	TR	PS
1	10	0.1	0.7	Mask matrix	0.1	0.4	0.1	Random
2	20	0.15	0.8	Arithmetic AND	0.2	0.5	0.2	Tournament
3	30	0.2	0.9	Arithmetic OR	0.3	0.6	0.3	Roulette wheel

**Fig. 9** *S/N* ratio for the proposed algorithm's factors

$$RPD_{\text{experiment}} = \frac{z_{\text{best across all algorithms}} - z_{\text{experiment}}}{z_{\text{best across all algorithms}}} \times 100. \quad (14)$$

In our case, the lower mean RPD implies the better solution. The terminating criterion should be varying based on the size of problem, so each algorithm runs up until 50, 40, 30 iterations in a row with no improvements for small, medium and large size instances, respectively. The run time is then calculated without including these last iterations. Note that for small, medium and large sizes each algorithm executed 10, 10 and 5 times, respectively.

Table 7 summarizes computational results for each algorithm and for each size. It shows superiority of the proposed algorithm in almost all indicators and all instances (except 1/25, 4/25 and 2/32 instances in small, medium and large size problems, respectively), which means that the proposed algorithm produces better and more robust results when compared with the others. Note that for large size instances, PSO achieves a better average CPU time,

and we will discuss and use it as a leverage to improve our proposed algorithm for large sizes in the next subsection. To examine the significance of this superiority, we should perform ANOVA test.

Table 8 shows results of ANOVA test for RPD (as the most important indicator) at 95% level. *P* values reveal that algorithms are significantly different. Since the proposed algorithm produces better RPD, it can be considered as superior to the others at 95% confidence level. Because of close results of PSO to the proposed algorithm, another ANOVA test is performed at 95% confidence level and the results are shown in Table 9.

Further discussions and improvements

We found out that the proposed algorithm has superiority in terms of all indicators and it performed better than other algorithms, and the only significant merit of PSO in comparison with the proposed algorithm is the computational time when problems have large size. Table 10 shows



Table 4 Orthogonal array L_{27}

# Trial	PN	MU	CR	CT	NP	LS	TR	PS
1	10	0.1	0.7	Mask	0.1	0.4	0.1	Random
2	10	0.1	0.7	Arithmetic AND	0.1	0.5	0.2	Tournament
3	10	0.1	0.7	Arithmetic OR	0.1	0.6	0.3	Roulette wheel
4	10	0.15	0.8	Mask	0.2	0.4	0.1	Tournament
5	10	0.15	0.8	Arithmetic AND	0.2	0.5	0.2	Roulette wheel
6	10	0.15	0.8	Arithmetic OR	0.2	0.6	0.3	Random
7	10	0.2	0.9	Mask	0.3	0.4	0.1	Roulette wheel
8	10	0.2	0.9	Arithmetic AND	0.3	0.5	0.2	Random
9	10	0.2	0.9	Arithmetic OR	0.3	0.6	0.3	Tournament
10	20	0.1	0.8	Arithmetic OR	0.3	0.4	0.2	Random
11	20	0.1	0.8	Mask	0.3	0.5	0.3	Tournament
12	20	0.1	0.8	Arithmetic AND	0.3	0.6	0.1	Roulette wheel
13	20	0.15	0.9	Arithmetic OR	0.1	0.4	0.2	Tournament
14	20	0.15	0.9	Mask	0.1	0.5	0.3	Roulette wheel
15	20	0.15	0.9	Arithmetic AND	0.1	0.6	0.1	Random
16	20	0.2	0.7	Arithmetic OR	0.2	0.4	0.2	Roulette wheel
17	20	0.2	0.7	Mask	0.2	0.5	0.3	Random
18	20	0.2	0.7	Arithmetic AND	0.2	0.6	0.1	Tournament
19	30	0.1	0.9	Arithmetic AND	0.2	0.4	0.3	Random
20	30	0.1	0.9	Arithmetic OR	0.2	0.5	0.1	Tournament
21	30	0.1	0.9	Mask	0.2	0.6	0.2	Roulette wheel
22	30	0.15	0.7	Arithmetic AND	0.3	0.4	0.3	Tournament
23	30	0.15	0.7	Arithmetic OR	0.3	0.5	0.1	Roulette wheel
24	30	0.15	0.7	Mask	0.3	0.6	0.2	Random
25	30	0.2	0.8	Arithmetic AND	0.1	0.4	0.3	Roulette wheel
26	30	0.2	0.8	Arithmetic OR	0.1	0.5	0.1	Random
27	30	0.2	0.8	Mask	0.1	0.6	0.2	Tournament

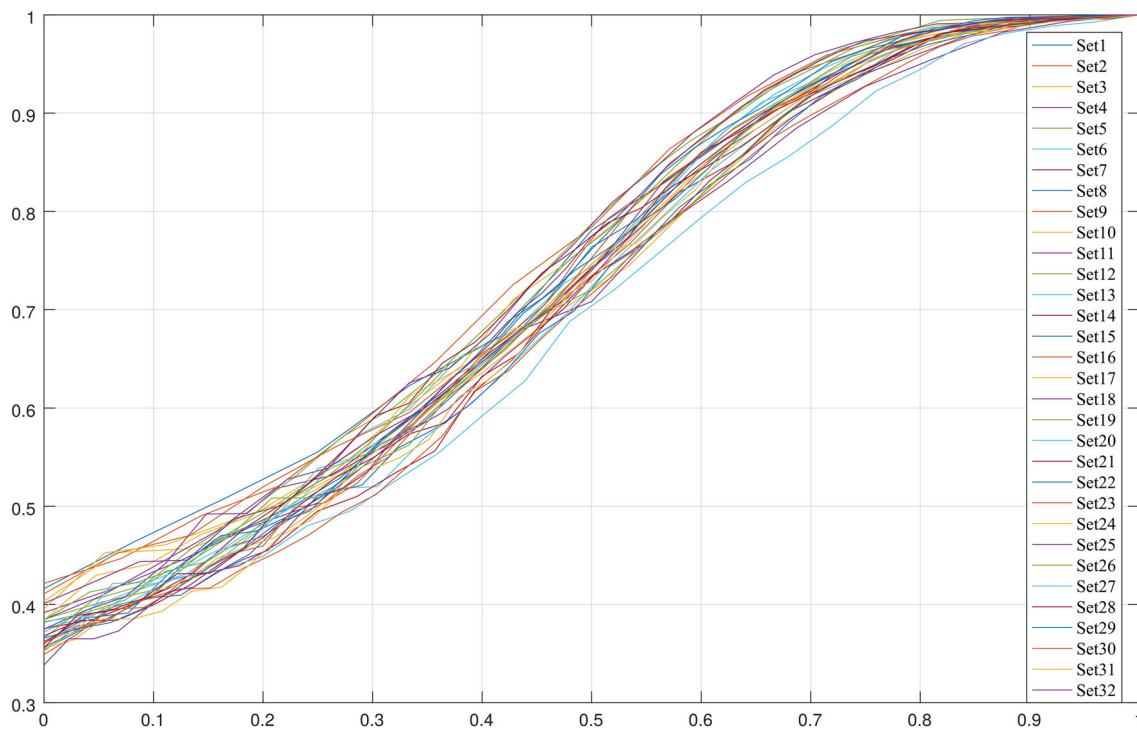


Fig. 10 PSO behaviour versus time based on mean fitness for large size instances

Table 5 Proposed algorithm's factors and values

Level	PN	MU	CR	NP	LS	TR	CT	PS
1	140.4	140.5	140.7	140.5	139.8	140.6	140.7	140.4
2	140.5	140.7	140.6	140.7	140.8	140.5	140.6	140.9
3	140.8	140.6	140.5	140.6	141.2	140.7	140.4	140.5
Delta	0.4	0.2	0.2	0.2	1.3	0.2	0.3	0.5
Rank	3	5	7	6	1	8	4	2

computational time for each algorithm and for each size based on size of problem (N and T).

The results show if the initial solution is good enough, the proposed algorithm can reach its best solution faster and more efficient. So we used PSO algorithm to generate initial solution. But to reach the best results we should find an optimal time for generating initial solution using PSO.

Based on the behaviour of PSO versus time (as represented in Fig. 10) we examined results for 60, 70, 80, 90 and 100 percent of needed time of PSO by running hydride algorithm for large size problems.

Table 11 shows the results, and it reveals that the best time for generating and combining PSO results with the proposed algorithm is 90%. It means if we generate initial solution of the large instances with PSO at 90% of needed time, we can improve the computation time of the proposed algorithm by almost 40% which is really significant and helpful at large size instances.

Conclusion and further research

The current paper studied the multi-period project portfolio selection problem. In order to solve real-world cases, we developed a novel mathematical model that considers

Table 6 GA, EM and PSO factors and values

Algorithm	Factors	Value
GA	Initial population size	100
	Mutation rate	0.2
	Crossover rate	0.9
	New population rate	0.1
EM-like	Initial population size	30
	Number of local search iterations	4
	Mutation rate	0.1
	Crossover rate	0.09
PSO	Initial population size	30
	Inertia weight	1
	Acceleration constant (cognitive component)	2
	Acceleration constant (social component)	2
	Acceleration constant (fully random component)	1

Factors and values of EM-like algorithm are extracted from [Golmohammadi and Pajoutan \(2011\)](#) and GA and PSO factors are calculated based on Taguchi method

Table 7 Computational results summary

Algorithm	Size	Mean	SD	Mean time	RPD
Proposed	Small	97,315.81	482.2145	2.398585	0.000244
	Medium	2,984,851	4267.029	96.14295	0.000233
	Large	42,183,207	15,833.35	1359.906	0.00002475
GA	Small	95,592.56	508.0994	8.967175	0.027924
	Medium	2,924,360	17,529.18	233.2555	0.024827
	Large	41,014,050	63,395.4	1944.904	0.03413694
PSO	Small	96,189.83	514.9191	5.367311	0.020501
	Medium	2,980,726	4005.634	100.6131	0.00722
	Large	42,036,953	44,297.95	510.1284	0.01042391
EM-like	Small	81,369.56	576.1008	134.9864	0.170351
	Medium	2,548,406	8379.092	494.43	0.16043
	Large	34,453,016	50,203.17	2849.459	0.18758738



Table 8 ANOVA test results for algorithms at 95%

	DF	SS	MS	F	P
Small	3	0.45568	0.15189	56.41	0.000
Medium	3	0.42805	0.14268	102.01	0.000
Large	3	0.73558	0.24519	117.00	0.000

Table 9 ANOVA test results for proposed and PSO algorithms at 95% level

	DF	SS	MS	F	P	
PSO	Small	1	0.005129	0.005129	19.76	0.000
	Medium	1	0.000610	0.000610	5.12	0.028
	Large	1	0.001730	0.001730	10.97	0.002

dependencies between projects cost, risk issues, stochastic revenue, and the possibility of investing in financial institutions such as banks. Due to NP-hard nature of this type of problems, a new meta-heuristic algorithm was proposed to solve the proposed model. We enhance genetic algorithm (GA) by using a new solution representation, examining and selecting best possible operators, and developing a local search mechanism. After some modification and parameter tuning, its performance was compared with GA, particle swarm optimization (PSO) and electromagnetism-

like (EM-like), which have been vastly used in the literature. Three sizes of problems were generated and the performance of these algorithms was compared with regard to four criteria, namely: the mean gained fitness, standard deviation, relative percentage deviation and run time. In all sizes, the proposed algorithms produced more robust results in about 90% of instances. We also investigated the performance of the algorithm versus time and provided a time-based terminating criterion for each size based on n and T . Results show that the proposed algorithm needs lower run time in small and medium size instances. To enhance run time for in large instances, we applied PSO algorithm to generate some initial solutions and found the best time to combine these two algorithms, so the run time was reduced by more than 40%.

There is substantial scope for future research in the area of project portfolio selection. Some possible extensions are as follows:

- (a) The proposed model will be extended in the nearest future by including resource constraints, assuming that the capital can be invested in each period and considering the possibility of suspension and pause in projects fulfillment.
- (b) To overcome the uncertainty associated with parameters, the presented model in this study will be extended by fuzzy logic.
- (c) Interdependencies between projects as well as cost relations will also be modelled and taken into account in the future research.

Table 10 Needed time for each algorithm and each size of problem

Size	Algorithm	Needed time
Small	Proposed	$6.95 \times 10^{-3}N^2 + 0.12NT + 0.225142T^2 - 0.54N - 2.73T + 8.03$
	GA	$2.65 \times 10^{-2}N^2 + 0.35NT - 3.64 \times 10^{-2}T^2 - 1.57N - 1.11T + 6.24$
	PSO	$0.0058N^2 + 0.08NT + 3.15 \times 10^{-1}T^2 - 0.13N - 2.6T + 5.25$
	EM-like	$3.81 \times 10^{-1}N^2 + 5.19NT - 8.55 \times 10^{-1}T^2 - 23.37N - 13.85T + 84.05$
Medium	Proposed	$0.118N^2 + 1.57NT + 3.28T^2 - 19.64N - 99.76T + 727$
	GA	$0.404N^2 + 2.67NT - 2.48T^2 - 44.69N - 10.55T + 649$
	PSO	$-0.061N^2 + 0.493NT + 1.01T^2 + 5.73N - 25.56T - 62$
	EM-like	$0.856N^2 + 5.65NT - 5.33T^2 - 94.87N - 20.45T + 1367$
Large	Proposed	$1.71N^2 + 11.34NT + 12.26T^2 - 338.94N - 869.53T + 15093$
	GA	$4.84N^2 + 11.15NT + 5.70T^2 - 703.45N - 634.43T + 24284$
	PSO	$-7.32 \times 10^{-1}N^2 + 1.01NT + 1.35T^2 + 92.74N + 62.54T + 2784$
	EM-like	$-0.002N^2 - 1.65 \times 10^{-3}NT + 1.85 \times 10^{-3}T^2 + 3.14 \times 10^{-1}N + 4.50 \times 10^{-2}T + 10.15$

Table 11 Time effect of hybridization with PSO

Hybridization point	60%	70%	80%	90%	100%
Time	1363.919	1256.267	869.299	803.5462	823.6545

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Bhattacharyya R, Kumar P, Kar S (2011) Fuzzy R&D portfolio selection of interdependent projects. *Comput Math Appl* 62(10):3857–3870. <https://doi.org/10.1016/j.camwa.2011.09.036>
- Charnes A, Cooper WW (1962) Chance constraints and normal deviates. *J Am Stat Assoc* 57(297):73–79. <https://doi.org/10.1080/01621459.1962.10482155>
- Doerner K, Gutjahr W, Hartl R, Strauss C, Stummer C (2006) Pareto ant colony optimization with ILP preprocessing in multiobjective project portfolio selection. *Eur J Op Res* 171:830–841. <https://doi.org/10.1016/j.ejor.2004.09.009>
- Fang Y, Chen L, Fukushima M (2008) A mixed R&D securities portfolio selection model. *Eur J Op Res* 185:700–715. <https://doi.org/10.1016/j.ejor.2007.01.002>
- Golmohammadi A, Pajoutan M (2011a) Meta heuristics for dependent portfolio selection problem considering risk. *Expert Syst Appl* 38:5642–5649. <https://doi.org/10.1016/j.eswa.2010.10.071>
- Golmohammadi A, Pajoutan M (2011b) Meta heuristics for dependent portfolio selection problem considering risk. *Expert Syst Appl* 38(5):5642–5649. <https://doi.org/10.1016/j.eswa.2010.10.071>
- Guang-Feng D, Woo-Tsong L, Chih-Chung L (2012) Markowitz-based portfolio selection with cardinality constraints using improved particle swarm optimization. *Expert Systems Application*. <https://doi.org/10.1016/j.eswa.2011.09.129>
- Hassanzadeh F, Nemati H, Sun M (2014) Robust optimization for interactive multiobjective programming with imprecise information applied to R&D project portfolio selection. *Eur J Oper Res* 238(1):41–53. <https://doi.org/10.1016/j.ejor.2014.03.023>
- Hosseinasab A, Ahmadi A (2015) Selecting a supplier portfolio with value, development, and risk consideration. *Eur J Oper Res* 245(1):146–156. <https://doi.org/10.1016/j.ejor.2015.02.041>
- Huang X (2008) Portfolio selection with a new definition of risk. *Eur J Oper Res* 186(1):351–357. <https://doi.org/10.1016/j.ejor.2007.01.045>
- Kocadağlı O, Keskin R (2015) A novel portfolio selection model based on fuzzy goal programming with different importance and priorities. *Expert Syst Appl* 42(20):6898–6912
- Lean Y, Shouyang W, Fenghua W (2012) Genetic algorithm-based multi-criteria project portfolio selection. *Ann Oper Res* 197(1):71–86. <https://doi.org/10.1007/s10479-010-0819-6>
- Lee G, Morrison A, O’Leary J (2006) The economic value portfolio matrix: a target market selection tool for destination marketing organizations. *Tour Manage* 27(4):576–588. <https://doi.org/10.1016/j.tourman.2005.02.002>
- Li T, Zhang W, Xu W (2015) A fuzzy portfolio selection model with background risk. *Appl Math Comput* 256:505–513
- Lorca Á, Prina J (2014) Power portfolio optimization considering locational electricity prices and risk management. *Electr Power Syst Res* 109:80–89. <https://doi.org/10.1016/j.epsr.2013.12.004>
- Mathuria P, Bhakar R, Li F (2015) GenCo’s optimal power portfolio selection under emission price risk. *Electr Power Syst Res* 121:279–286. <https://doi.org/10.1016/j.epsr.2014.11.006>
- PMI (2008) The standard for portfolio management, 2nd edn. Project Management Institute, Newtown Square, Pennsylvania
- Rebiasz B (2013) Selection of efficient portfolios—probabilistic and fuzzy approach, comparative study. *Comput Ind Eng* 64(4):1019–1032. <https://doi.org/10.1016/j.cie.2013.01.011>
- Tofighian AA, Naderi B (2015) Modeling and solving the project selection and scheduling. *Comput Ind Eng* 83:30–38. <https://doi.org/10.1016/j.cie.2015.01.012>
- Vazhayil J, Balasubramanian R (2014) Optimization of India’s electricity generation portfolio using intelligent Pareto-search genetic algorithm. *Int J Electr Power Energy Syst* 55:13–20. <https://doi.org/10.1016/j.ijepes.2013.08.024>
- Wu H, Chen H (2015) Nash equilibrium strategy for a multi-period mean–variance portfolio selection problem with regime switching. *Econ Model* 46:79–90
- Zhu H, Yi W, Kesheng W, Yun C (2011) Particle Swarm Optimization (PSO) for the constrained portfolio optimization problem. *Expert Syst Appl* 38(8):10161–10169. <https://doi.org/10.1016/j.eswa.2011.02.075>

