

A Genetic Algorithm Approach for Multi-criteria Project Selection for Analogy-Based Software Cost Estimation

Sweta Kumari and Shashank Pushkar

Abstract This paper presents genetic algorithms as multi-criteria project selection for improving the Analogy Based Estimation (ABE) process, which is suitable to reuse past project experience to create estimation of the new projects. An attempt has also been made to create a multi-criteria project selection problem with and without allowing for interactive effects between projects based on criteria which are determined by the decision makers. Two categories of projects are also presented for comparison purposes with other traditional optimization methods and the experimented results show the capability of the proposed Genetic Algorithm based method in multi-criteria project selection problem and it can be used as an efficient solution to the problem that will enhance the ABE process. Here, Mean Absolute Relative Error (MARE) is used to evaluate the performance of ABE process and it has been found that interactive effects between projects may change the results.

Keywords Software cost estimation • Analogy based estimation • Genetic algorithm • Multi-criteria decision making • Nonlinear integer programming

1 Introduction

Success of software organizations rely on proper management activities such as planning, budgeting, scheduling, resource allocation and effort requirements for software projects. Software effort estimation is the process of making an approximate judgment of the costs of software. Inaccurate estimation of software cost lowers the proficiency of the project, wastes the company's budget and can result in failure of the

S. Kumari (✉) · S. Pushkar
Department of CSE, BIT, Mesra, Ranchi, India
e-mail: swetak44@gmail.com

S. Pushkar
e-mail: shashank_pushkar@yahoo.com

entire project. Broadly, there are two types of cost estimation methods: Algorithmic and Non-algorithmic. Algorithmic method calculates cost of the software by using formula or some experimental equations whereas Non-algorithmic methods use a historical data that is related to the previously completed similar projects for calculating the cost of the software [1]. Analogy based estimation is a Non-algorithmic method in that a critical role is played by the similarity measures between a pair of projects. Here, a distance is calculated between the software project being estimated and each of the historical software projects. It then finds the most similar project that is used to estimate the cost. Estimation by analogy is essentially a form of Case Based Reasoning [2]. However, as it is argued in [3] there are certain advantages in respect with rule based systems, such as the fact that users are keen to accept solutions from analogy based techniques, rather than solutions derived from uncomfortable chains of rules or neural nets. Naturally, there are some difficulties with analogy-based estimation such as lack of appropriate analogues and issues with selecting and using them. Choosing an appropriate set of projects participating in cost estimation process are very important for any organizations to achieve their goals. In this process, several reasons involve such as the number of investment projects, presence of multiple decision criteria such as takings maximization or risk minimization, business and operational rules such as budget limits and time windows for starting dates. Project selection is a difficult task, if the project interactions in terms of multiple selection criteria and information of preferences of decision-makers are taken into account, mainly in the presence of a huge number of projects.

2 Related Works

Various methods and mathematical models have been developed to deal with the problem of selecting and scheduling projects. Since the pioneering ranking method from [4] other methods have been proposed: Scoring [5], Analytical Hierarchy Process [6, 7] and Goal Programming [8] among others. However, these methods think that project interdependencies do not exist [9]. Other authors have proposed project selection models that deal with the existence of interdependencies. According to [8], these models, classified by the fundamental solution method are: Dynamic Programming [10] models reflect interdependency in special cases; Quadratic/Linear 0–1 programming models have a quadratic objective function and linear constraints, they limit interdependency only in the objective and between two projects. Quadratic/Quadratic 0–1 programming [11] with interdependency between two projects in the objective function and in the resource constraints; and Nonlinear 0–1 programming, with interdependency reflected in the objective function and the constraints among as many projects as necessary [9, 12] presents a linear 0–1 programming model for the selection and scheduling of projects, that includes technical interdependency. Medaglia et al. [13] also present some features of

technical interdependence in linear project selection and scheduling models. According to the literature review, it has been found that estimation by analogies requires a significant amount of computations as well as lack of appropriate analogous and issues with selecting and using them. And it has been also observed that interactive effects between projects are significant for selecting a project because it may give unwanted result [14]. In this research, we propose Genetic Algorithms (GA) as multi-criteria project selection for improving the ABE system's performance and try to make a multi-criteria project selection problem allowing for project interactions that are based on multiple criteria and Decision makers preference information based on some significance. Here, the performance of ABE system is analyzed and compared in terms of MARE. Rest of the paper is divided in different sections as follows: a detail formulation of a multi-criteria project selection problem is described in Sect. 2. In Sect. 3, GA is used as the optimization technique for Project Selection problem. In Sect. 4, Numerical examples are also given for illustration purpose. Experiments and comparison results are described in Sect. 5 and in the end, the concluding remark is discussed.

3 Formulation of the Project Selection Problem

In project selection, a decision-maker is deals with the problem of selecting an appropriate subset of projects from an inappropriate set of projects based on a set of selection criteria. This process is known as the multi-criteria decision making (MCDM) process [15–17]. In general, there are two types of MCDM problems: multi-attribute decision making (MADM) problem and multi-objective decision making (MODM) problem. Based on this categorization, multi-criteria project selection problem is seen as a distinctive MADM problem in terms of the characteristics of project selection. In this paper N projects are considered for selection as well as evaluation, and decision variable x_n denotes whether the project is selected or not. If there are no interactive effects between projects, the project selection problem can be formulated in the following form:

$$\begin{aligned}
 \text{Max } E &= \sum_{n=1}^N \left(\sum_{j=1}^J p_j c_{nj} \right) x_n \\
 \text{s.t. } &\sum_{n=1}^N x_n = R \\
 &x_n = \{0, 1\}, x_n = 1, 2, \dots, 7
 \end{aligned} \tag{1}$$

Therefore, the multi-criteria project selection problem with interactive effects between projects based on criteria can be formulated in the following form:

$$\begin{aligned}
Max E = & \sum_{n=1}^N \left(\sum_{j=1}^J p_j c_{nj} \right) x_n + \sum_{j=1}^J \sum_{m=1}^M \left(p_j (b_j(c_m)) \left(\sum_{n=1}^Q c_{nj} \right) \right) \prod_{n=1}^Q x_n \\
s.t. & \sum_{n=1}^N x_n = R \\
& x_n = \{0, 1\}
\end{aligned} \tag{2}$$

Here, E is the total effects of selected projects, R is the number of the selected projects based on criteria, N is the number of projects to be evaluated and selected, P_j is preference degree of decision makers on criterion j , $j = 1, 2 \dots J$, C_{nj} is the value of projects n in criteria j , $b_j(c_m)$ is the value of interactive effects in a combination of m projects in j , $j = 1, 2 \dots J$, c_m is the Combination of m projects, $m = 1, 2 \dots M$ and Q is the number of variables with interactive effects. It has been seen that, Eq. (2) becomes a 0–1 nonlinear integer programming problem. This kind of problem may be solved by branch and bound algorithm and dynamic programming method [18]. To solve this kind of problem, we incorporated the Genetic Algorithms for project selection.

4 GA-Based Optimization Approach

GA is optimization algorithms in evolutionary computing techniques, proposed in 1975 by a scientist Holland and extensively studied by De Jong, Goldberg [19, 20] and others [21]. It is inspired by natural biological evolution. GAs operates on a population of potential solutions applying the principle of survival of the fittest to produce better and better approximations to a solution. At each generation, a new set of approximations is created by the process of selecting individuals according to their level of fitness in the problem domain and breeding them together using operators borrowed from natural genetics. This process leads to the evolution of populations of individuals that are better suited to their environment than the individuals that they were created from, just as in natural adaptation. A detailed process of the Project Selection for ABE using GA is as follows:

1. **Chromosome:** Each individual chromosome consists of a number of digits. The value of each bit is set to be either 0 or 1: 0 means the related project in the historical data set is not selected and 1 means it is selected.
2. **Population generation:** After the encoding of the individual chromosome, the system then generates a population of chromosomes. It often represented in an array of chromosomes, while a chromosome is composed of many genes. In the mathematical optimization problem, a gene corresponds to a variable x_n , and a chromosome corresponds to a solution represented in a set of genes $x = (x_1, x_2 \dots x_R)$ if R variables exist.

3. **Fitness function:** Each individual is evaluated by using the fitness function in GA. The GA is designed to maximize the fitness value. Generally the fitness function defines a score which gives each chromosome the probability to be chosen for reproduction or to survive.
4. **Selection:** Selection operator is used to create the population with higher fitness. Here, roulette wheel approach is used to select chromosomes from the current population with higher fitness.
5. **Crossover:** The main goal of crossover operator is to generate different offspring chromosomes to obtain a more optimal solution than their parents. Apply crossover techniques such as one-point, two-point and multi-point to initial chromosomes to produce new offspring chromosomes.
6. **Mutation:** Each bit of the chromosomes in the new population is chosen to change its value with a probability of 0.1, in such a way that a bit “1” is changed to “0” and a bit “0” is changed to “1”. The main goal of the mutation operation is to prevent the GA from converging too quickly in a small area of the search space.
7. **Stopping criteria:** The new chromosome will be evaluated to verify whether it is a best solution or not. If it is satisfied, then the optimized results are obtained. If not satisfied, then it is repeated from Step 3 to Step 7 until a certain number of generations, a best fitness value or a convergence criterion of the population is reached.

5 Numerical Examples

The COCOMO data set [22] is chosen for the experiments and comparisons. This dataset consists of two independent variables-Size and EAF (Effort Adjustment Factor) and one dependent variable-Effort. Size is in KLOC (thousands of lines of codes) and effort is given in man-months. In this research our main task to reduce the whole project into an appropriate set of projects participating in cost estimation process that could save computing time and produce accurate results since it eliminates unwanted projects and contains only related projects. Here, we also compare the efficiency of GA-based optimization approach to the traditional approach. ILOG CPLEX barrier is a typical nonlinear optimizer which is used for computation and comparison purpose. This dataset is divided into two categories of projects. The first category of projects contains only seven variables and our objective is to select two best projects from the given projects and the second category of projects contains twenty-one variables in the project selection. Here, we want to select eleven best projects based on criteria which are decided by decision makers. Let the two criteria are $j = \{1: \text{Magnitude of Relative Error (MRE)},$

2: Absolute Relative Error (ARE)}. The MRE and ARE can be calculated by the following equation:

$$MRE = \frac{|Actual - Estimated|}{Actual} \quad (3)$$

$$ARE = |Actual - Estimated| \quad (4)$$

Let P_j represents the preference degree which is determined by decision makers in terms of criteria j . In general, preference can be calculated by the following equation:

$$p'_j = \frac{p_j}{\sum_{j=1}^J p_j} \quad (5)$$

Table 1 shows the MRE and ARE value of seven different projects and Table 2 contains data which was obtained by using the Eq. (5).

Here, the interactive effects between these projects are not considered. But, it can be calculated by some statistical methods such as analysis of variance (ANOVA) [23]. For comparison purpose, interactive effects between the projects are also given in Table 3. Here, interactive effects between projects are assumed.

When the interactive effects between projects are not considered, the optimization problem in terms of Eq. (1) for project selection can be formulated as

$$\begin{aligned} \text{Max } E &= 20.793x_1 + 3.897x_2 + 8.378x_3 + 1.214x_4 \\ &+ 344.213x_5 + 99.798x_6 + 107.477x_7 \\ \text{s.t. } &x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 = 2 \\ &x_n = \{0, 1\}, \quad n = 1, 2, \dots, 7 \end{aligned} \quad (6)$$

Table 1 Seven different projects of COCOMO dataset

Criteria (j)	Preference (P_j)	Projects (n)						
		$e1$	$e2$	$e3$	$e4$	$e5$	$e6$	$e7$
ARE	2	31.44	5.82	12.55	1.73	519.11	151.03	162.59
MRE	1	0.13	0.17	0.29	0.22	4.85	0.36	0.51

Table 2 Data which was obtained by using the Eq. (5) for seven different projects of COCOMO dataset

Criteria (j)	Preference (P_j)	Projects (n)						
		$e1$	$e2$	$e3$	$e4$	$e5$	$e6$	$e7$
ARE	0.66	31.44	5.82	12.55	1.73	519.11	151.03	162.59
MRE	0.33	0.13	0.17	0.29	0.22	4.85	0.36	0.51

Table 3 Interactive effects between projects based on criteria

Criteria (j)	Projects pairs (C_m)										
	$e1e2$	$e1e3$	$e1e4$	$e1e5$	$e1e6$	$e1e7$	$e2e3$	$e2e4$	$e2e5$	$e2e6$	
ARE	0.55	0.60	0.30	0.20	0.50	0.20	0.45	0.40	0.65	0.45	
MRE	-0.15	0	-0.20	-0.05	-0.25	0	-0.42	-0.15	0	-0.15	
Criteria (j)	Projects pairs (C_m)										
	$e2e7$	$e3e4$	$e3e5$	$e3e6$	$e3e7$	$e4e5$	$e4e6$	$e4e7$	$e5e6$	$e5e7$	$e6e7$
ARE	0.35	0.55	0.40	0.60	0.55	0.40	0.65	0.45	0.55	0.60	0.30
MRE	-0.10	0	0	-0.80	-0.35	-0.15	-0.05	-0.03	0	-0.15	-0.22

Table 4 Comparison of objective value and computational time for GA-based method and CPLEX optimizer for seven variables

Equation no.	ILOG CPLEX barrier optimizer based objective value	GA based objective value	Computational time (s)	
			CPLEX	GA
6	451.691	728.6767	9	11
7	721.378	857.2201	13	15

If interactive effects between projects are considered, then the problem will become complex and from Eq. (2), the optimization problem allowing for project interactions can be formulated as

$$\begin{aligned}
 \text{Max } E = & 20.793x_1 + 3.897x_2 + 8.378x_3 + 1.214x_4 + 344.213x_5 + 99.798x_6 \\
 & + 107.477x_7 + 0.02x_1x_2 + 17.420x_1x_3 + 6.544x_1x_4 + 72.591x_1x_5 \\
 & + 60.174x_1x_6 + 25.612x_1x_7 + 5.392x_2x_3 + 1.974x_2x_4 + 255.195x_2x_5 \\
 & + 46.558x_2x_6 + 38.88x_2x_7 + 5.184x_3x_4 + 140.358x_3x_5 + 64.605x_3x_6 \\
 & + 63.483x_3x_7 + 137.251x_4x_5 + 65.525x_4x_6 + 48.796x_4x_7 \\
 & + 243.261x_5x_6 + 269.688x_5x_7 + 62.034x_6x_7 \\
 \text{s.t. } & x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 = 2 \\
 & x_n = \{0, 1\}, \quad n = 1, 2, \dots, 7
 \end{aligned} \tag{7}$$

The above Eqs. (6) and (7) can be solved by a typical nonlinear optimizer, ILOG CPLEX barrier optimizer as well as GA-based method and the results are shown in Table 4.

Here, it is also noticed that from Eqs. (6) and (7), when interactive effects between projects are not considered, then the optimal solution $x = (0, 0, 0, 0, 0, 1, 1)$ and the optimal solution $x = (0, 0, 0, 0, 1, 0, 1)$ when it is considered. It indicates

Table 5 Comparison of objective value and computational time for GA-based approach and CPLEX optimizer for twenty-one variables

Equation no.	ILOG CPLEX barrier optimizer based objective value	GA based objective value	Computational time (s)	
			CPLEX	GA
8	1139.955	1579.1074	1.07	15
9	6699.812	8812.295	1.16	22

that interactive effects are very important for project selection and the computational time of CPLEX barrier optimizer is better than the GA-based optimization method because of small number of different projects and quadratic terms because GA-based optimization is a heuristic optimization method that may change the finale result, when a small size problem is encountered.

In this section, we want to check the efficiency of the proposed GA-based optimization method to the other nonlinear optimization method. Suppose there are twenty-one projects and more than two evaluations are presented and we want to select eleven best projects from the given selection problem. When the interactive effects between projects are not considered, then the objective function can be computed by using Eq. (1) as

$$\begin{aligned}
 \text{Max } E = & 20.793x_1 + 3.897x_2 + 8.378x_3 + 1.214x_4 + 344.213x_5 + 99.798x_6 \\
 & + 107.477x_7 + 27.166x_8 + 27.888x_9 + 1.647x_{10} + 9.016x_{11} + 8.326x_{12} \\
 & + 2.996x_{13} + 1.512x_{14} + 133.983x_{15} + 85.022x_{16} + 157.615x_{17} \\
 & + 112.854x_{18} + 12.851x_{19} + 23.146x_{20} + 0.128x_{21} \\
 \text{s.t. } & \sum_{i=1}^{21} x_n = 11 \\
 & x_n = \{0, 1\}, \quad n = 1, 2, \dots, 21
 \end{aligned} \tag{8}$$

Interactive effects between projects are considered and similarly, the objective function E can be calculated by using Eq. (2) and the results are shown in Table 5.

$$\begin{aligned}
\text{Max } E = & 4.903x_{1x_2} + 13.065x_{1x_3} + 8.729x_{1x_4} + 236.186x_{1x_5} + 54.194x_{1x_6} + 44.792x_{1x_7} \\
& + 26.316x_{1x_8} + 19.427x_{1x_9} + 14.775x_{1x_{10}} + 16.338x_{1x_{11}} + 8.674x_{1x_{12}} + 5.902x_{1x_{13}} \\
& + 11.092x_{1x_{14}} + 108.212x_{1x_{15}} + 79.24x_{1x_{16}} + 35.618x_{1x_{17}} + 86.748x_{1x_{18}} + 18.428x_{1x_{19}} \\
& + 15.287x_{1x_{20}} + 6.244x_{1x_{21}} + 4.83x_{2x_3} + 3.226x_{2x_4} + 69.208x_{2x_5} + 46.56x_{2x_6} + 66.624x_{2x_7} \\
& + 17.021x_{2x_8} + 7.915x_{2x_9} + 1.901x_{2x_{10}} + 1.903x_{2x_{11}} + 7.258x_{2x_{12}} + 3.388x_{2x_{13}} \\
& + 1.048x_{2x_{14}} + 41.311x_{2x_{15}} + 39.946x_{2x_{16}} + 104.826x_{2x_{17}} + 69.881x_{2x_{18}} + 13.257x_{2x_{19}} \\
& + 14.777x_{2x_{20}} + 1.378x_{2x_{21}} + 3.769x_{3x_4} + 122.593x_{3x_5} + 86.349x_{3x_6} + 75.135x_{3x_7} \\
& + 15.925x_{3x_8} + 10.831x_{3x_9} + 2.474x_{3x_{10}} + 3.433x_{3x_{11}} + 10.751x_{3x_{12}} + 6.744x_{3x_{13}} \\
& + 4.372x_{3x_{14}} + 35.503x_{3x_{15}} + 18.634x_{3x_{16}} + 107.735x_{3x_{17}} + 42.267x_{3x_{18}} + 6.281x_{3x_{19}} \\
& + 7.827x_{3x_{20}} + 3.781x_{3x_{21}} + 85.721x_{4x_5} + 10.076x_{4x_6} + 32.535x_{4x_7} + 22.492x_{4x_8} \\
& + 20.273x_{4x_9} + 0.548x_{4x_{10}} + 3.495x_{4x_{11}} + 7.988x_{4x_{12}} + 1.844x_{4x_{13}} + 0.887x_{4x_{14}} \\
& + 26.992x_{4x_{15}} + 43.002x_{4x_{16}} + 87.233x_{4x_{17}} + 45.475x_{4x_{18}} + 8.301x_{4x_{19}} + 13.255x_{4x_{20}} \\
& + 0.483x_{4x_{21}} + 154.803x_{5x_6} + 359.76x_{5x_7} + 258.72x_{5x_8} + 314.114x_{5x_9} + 68.849x_{5x_{10}} \\
& + 105.471x_{5x_{11}} + 227.731x_{5x_{12}} + 189.702x_{5x_{13}} + 68.809x_{5x_{14}} + 95.295x_{5x_{15}} \\
& + 192.041x_{5x_{16}} + 199.68x_{5x_{17}} + 295.958x_{5x_{18}} + 159.925x_{5x_{19}} + 127.888x_{5x_{20}} \\
& + 188.341x_{5x_{21}} + 82.724x_{6x_7} + 76.018x_{6x_8} + 70.124x_{6x_9} + 45.593x_{6x_{10}} + 32.582x_{6x_{11}} \\
& + 37.715x_{6x_{12}} + 46.183x_{6x_{13}} + 60.667x_{6x_{14}} + 93.416x_{6x_{15}} + 101.514x_{6x_{16}} + 128.518x_{6x_{17}} \\
& + 180.45x_{6x_{18}} + 84.343x_{6x_{19}} + 24.542x_{6x_{20}} + 34.91x_{6x_{21}} + 53.721x_{7x_8} + 81.018x_{7x_9} \\
& + 21.762x_{7x_{10}} + 52.319x_{7x_{11}} + 46.167x_{7x_{12}} + 71.673x_{7x_{13}} + 48.934x_{7x_{14}} + 84.368x_{7x_{15}} \\
& + 105.718x_{7x_{16}} + 105.908x_{7x_{17}} + 132.011x_{7x_{18}} + 66.048x_{7x_{19}} + 39.029x_{7x_{20}} + 26.83x_{7x_{21}} \\
& + 27.463x_{8x_9} + 20.114x_{8x_{10}} + 27.031x_{8x_{11}} + 7.056x_{8x_{12}} + 19.541x_{8x_{13}} + 15.681x_{8x_{14}} \\
& + 56.299x_{8x_{15}} + 33.561x_{8x_{16}} + 83.056x_{8x_{17}} + 48.924x_{8x_{18}} + 29.913x_{8x_{19}} + 30.081x_{8x_{20}} \\
& + 10.881x_{8x_{21}} + 19.138x_{9x_{10}} + 7.349x_{9x_{11}} + 16.215x_{9x_{12}} + 18.449x_{9x_{13}} + 16.088x_{9x_{14}} \\
& + 40.42x_{9x_{15}} + 39.421x_{9x_{16}} + 55.554x_{9x_{17}} + 84.316x_{9x_{18}} + 20.282x_{9x_{19}} + 10.16x_{9x_{20}} \\
& + 8.382x_{9x_{21}} + 4.767x_{10x_{11}} + 6.423x_{10x_{12}} + 2.747x_{10x_{13}} + 2.396x_{10x_{14}} + 74.524x_{10x_{15}} \\
& + 30.285x_{10x_{16}} + 63.639x_{10x_{17}} + 40x_{10x_{18}} + 11.524x_{10x_{19}} + 16.032x_{10x_{20}} + 0.614x_{10x_{21}} \\
& + 3.424x_{11x_{12}} + 5.361x_{11x_{13}} + 4.12x_{11x_{14}} + 92.832x_{11x_{15}} + 42.248x_{11x_{16}} + 58.217x_{11x_{17}} \\
& + 66.904x_{11x_{18}} + 8.694x_{11x_{19}} + 19.191x_{11x_{20}} + 1.811x_{11x_{21}} + 5.044x_{12x_{13}} + 5.754x_{12x_{14}} \\
& + 56.819x_{12x_{15}} + 51.236x_{12x_{16}} + 82.815x_{12x_{17}} + 102.756x_{12x_{18}} + 15.775x_{12x_{19}} \\
& + 6.257x_{12x_{20}} + 5.006x_{12x_{21}} + 2.396x_{13x_{14}} + 82.091x_{13x_{15}} + 26.34x_{13x_{16}} + 32.084x_{13x_{17}} \\
& + 57.824x_{13x_{18}} + 3.147x_{13x_{19}} + 6.491x_{13x_{20}} + 1.538x_{13x_{21}} + 47.353x_{14x_{15}} + 38.82x_{14x_{16}} \\
& + 47.668x_{14x_{17}} + 51.363x_{14x_{18}} + 7.777x_{14x_{19}} + 4.873x_{14x_{20}} + 1.008x_{14x_{21}} + 164.093x_{15x_{16}} \\
& + 72.823x_{15x_{17}} + 110.862x_{15x_{18}} + 58.655x_{15x_{19}} + 70.592x_{15x_{20}} + 87.087x_{15x_{21}} \\
& + 48.454x_{16x_{17}} + 118.582x_{16x_{18}} + 34.189x_{16x_{19}} + 32.387x_{16x_{20}} + 42.488x_{16x_{21}} \\
& + 67.528x_{17x_{18}} + 127.681x_{17x_{19}} + 31.519x_{17x_{20}} + 94.544x_{17x_{21}} + 62.721x_{18x_{19}} \\
& + 88.175x_{18x_{20}} + 84.621x_{18x_{21}} + 12.494x_{19x_{20}} + 10.962x_{19x_{21}} + 8.101x_{20x_{21}} \\
\text{s.t. } \sum_{i=1}^{21} x_n = 11 \\
x_n = \{0, 1\}, \quad n = 1, 2, \dots, 21
\end{aligned} \tag{9}$$

Similarly, from Eqs. (8) and (9), it has been found that when interactive effects between projects are not considered, then the optimal solution $x = (1, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 1, 0)$ and the optimal solution $x = (1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0)$ when it is considered. Here, it has been found that

Table 6 Results and comparisons on COCOMO dataset

Project category	MARE of before project selection	MARE of after project selection with without consideration of project interactions	MARE of after project selection with consideration of project interactions
Seven variable	126.33	44.81	97.38
Twenty-one variable	85.62	82.05	81.31

the computational times are increased for both GA-based optimization method and CPLEX barrier optimizer when the number of deferent projects and quadratic terms are increased but the results shows that GA is better than CPLEX for solving this type of problem.

6 Experimental Results and Comparisons

This approach has been experimented on a COCOMO dataset. To evaluate the performance of proposed GA as multi-criteria project selection for improving the ABE, the Mean Absolute Relative Error (MARE) is used here. The MARE is defined in Eq. (10) as follows:

$$MARE = \frac{1}{n} \sum_{i=1}^n |f_i - y_i| \quad (10)$$

where f_i is the Estimated and y_i is the Actual value respectively, n is the number of projects. Table 6 shows the comparison of results among various projects. It shows that MARE value of after project selection gives better results in comparison with MARE value of whole projects and also it has been found that the MARE value is changed with and without allowing for interactive effects between projects. These results suggest that the selection of an appropriate projects can not only reduce the information required to approximate software effort, but it can also result in obtaining better approximations and it also specify that GA-based optimization method can be used as an effective solution to other optimization method (i.e., CPLEX).

7 Concluding Remarks

In this study, a novel approach based on Genetic Algorithm has been proposed to solve the problem of multi-criteria project selection that will improve the performance of analogy based software cost estimation. It formulates a multi-criteria

project selection problem with and without interactive effects between projects based on criteria which are determined by the decision makers. Detailed examples have also been illustrated that describe the effectiveness of the GA-based optimization method to enhance the process of Software Cost Estimation. The proposed approach has also been compared with other optimization methods used. The well-known COCOMO datasets are chosen for the experiments and comparisons. The results show that the GA based project selection approach has lowest MARE value, so that it will be able to provide good estimation capabilities for ABE system. It has also been observed that the proposed approach has a limitation. In some cases this GA method is not able to find the exact global optimum because there is no absolute assurance to find the best solution. However, this limitation is for any such meta-heuristic technique used for optimization. The future direction can be experimenting with some more methods for project selection that can help to overcome the above limitation and can further improve the process of software cost estimation.

References

1. Kumari, S., Pushkar, S.: Performance analysis of the software cost estimation methods: a review. *Int. J. Adv. Res. Comput. Sci. Softw. Eng.* **3**, 229–238 (2013)
2. Mukhopadhyay, T., Vicinanza, S.S., Prietula, M.J.: Examining the feasibility of a case-based reasoning model for software effort estimation. *MIS Q.* **16**, 155–171 (1992)
3. Shepperd, M.J., Schofield, C.: Estimating software project effort using analogies. *IEEE Trans. Softw. Eng.* **23**(12), 736–743 (1997)
4. Lorie, J.H., Savage, L.J.: Three problems in rationing capital. *J. Bus.* **28**(4), 229–239 (1955)
5. Rengarajan, S., Jagannathan, P.: Projects selection by scoring for a large R&D organization in a developing country. *R&D Manage.* **27**, 155–164 (1997)
6. Lockett, G., Hetherington, B., Yallup, P.: Modelling a research portfolio using AHP: a group decision process. *R&D Manage.* **16**(2), 151–160 (1986)
7. Murahaldar, K., Santhanam, R., Wilson, R.L.: Using the analytical hierarchy process for information system project selection. *Inf. Manage.* **17**(1), 87–95 (1990)
8. Lee, J.W., Kim, S.H.: Using analytic network and goal programming for interdependent information systems project selection. *Comput. Oper. Res.* **19**, 367–382 (2000)
9. Santhanam, R., Kyparisis, G.J.: A decision model for interdependent information system project selection. *Eur. J. Oper. Res.* **89**, 380–399 (1996)
10. Nemhauser, G.L., Ullman, Z.: Discrete dynamic programming and capital allocation. *Manage. Sci.* **15**(9), 494–505 (1969)
11. Aaker, D.A., Tyebjee, T.T.: Model for the selection of interdependent R&D projects. *IEEE Trans. Eng. Manage.* **25**(2), 30–36 (1978)
12. Ghasemzadeh, F., Archer, N., Iyogun, P.: Zero-one model for project portfolio selection and scheduling. *J. Oper. Res. Soc.* **50**(7), 755 (1999)
13. Medaglia, A.L., Hueth, D., Mendieta, J.C., Sefair, J.A.: Multiobjective model for the selection and timing of public enterprise projects. *Soc. Econ. Plann. Sci.* (in press, 2007). <http://dx.doi.org/10.1016/j.seps.2006.06.009>
14. Carlsson, C., Fuller, R.: Multiple criteria decision making: the case for interdependence. *Comput. Oper. Res.* **22**, 251–260 (1995)
15. Korhonen, P., Moskowitz, H., Wallenius, J.: Multiple criteria decision support—a review. *Eur. J. Oper. Res.* **63**, 361–375 (1992)

16. Stewart, T.J.: A critical survey on the status of multiple criteria decision making theory and practice. *Omega* **20**, 569–586 (1992)
17. Ostermark, R.: Temporal interdependence in fuzzy MCDM problems. *Fuzzy Sets Syst.* **88**, 69–79 (1997)
18. Hammer, P., Rudeanu, S.: *Boolean Methods in Operations Research and Related Areas*. Springer, Berlin (1968)
19. De Jong, K.A.: Analysis of behavior of a class of genetic adaptive systems. Ph.D. thesis. University of Michigan, Ann Arbor, MI (1975)
20. Goldberg, D.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, New York (1989)
21. Back, T., Schwefel, H.P.: An overview of evolutionary algorithms for parameter optimization. *Evol. Comput.* **17**(1), 87–95 (1993)
22. Boehm, B.W.: *Software Engineering Economics*. Prentice-Hall, Englewood Cliffs (1981)
23. Cox, D.R.: Interaction. *Int. Stat. Rev.* **52**, 1–25 (1984)

Computational Intelligence in Data Mining - Volume 3
Proceedings of the International Conference on CIDM,
20-21 December 2014

Jain, L.C.; Behera, H.S.; Mandal, J.K.; Mohapatra, D.P.
(Eds.)

2015, XXVIII, 717 p. 254 illus., Hardcover

ISBN: 978-81-322-2201-9