



Contents lists available at ScienceDirect

Applied Soft Computing

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



Software project portfolio optimization with advanced multiobjective evolutionary algorithms

Thomas Kremmel^{a,*}, Jiří Kubalík^b, Stefan Biffl^c

^a *aeon-group. s3p Unternehmensberatungs GmbH, Linke Wienzeile 14/26, 1060 Vienna, Austria*

^b *Department of Cybernetics, CTU Prague, Technická 2, 166 27 Prague 6, Czech Republic*

^c *Christian Doppler Laboratory, Software Engineering Integration for Flexible Automation Systems, Favoritenstr. 9/188, A-1040 Vienna, Austria*

ARTICLE INFO

Article history:

Received 18 February 2009

Received in revised form 23 February 2010

Accepted 27 April 2010

Available online xxx

Keywords:

Evolutionary computing

Real-world complexities

Decision support

Multiobjective optimization

Software project portfolio management

ABSTRACT

Large software companies have to plan their project portfolio to maximize potential portfolio return and strategic alignment, while balancing various preferences, and considering limited resources. Project portfolio managers need methods and tools to find a good solution for complex project portfolios and multiobjective target criteria efficiently. However, software project portfolios are challenging to describe for optimization in a practical way that allows efficient optimization. In this paper we propose an approach to describe software project portfolios with a set of multiobjective criteria for portfolio managers using the COCOMO II model and introduce a multiobjective evolutionary approach, mPOEMS, to find the Pareto-optimal front efficiently. We evaluate the new approach with portfolios choosing from a set of 50 projects that follow the validated COCOMO II model criteria and compare the performance of the mPOEMS approach with state-of-the-art multiobjective optimization evolutionary approaches. Major results are as follows: the portfolio management approach was found usable and useful; the mPOEMS approach outperformed the other approaches.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction

Project portfolio management (PPM) is a set of processes used to support a business in conducting the mix of projects, which best fit the organisation's various needs. This set of processes includes the selection of projects an organisation conducts, maintaining the selected projects in portfolios, and periodically reviewing the mix of projects, to check whether the selection still supports the main business goals. Various portfolio processes have already been presented. Most of them do cover the whole life cycle of portfolios, from selecting projects and optimizing portfolios to phase gate review steps [1,16].

Portfolio optimization is a process in PPM that create the best mix of projects, out of all potential candidates. Selection of projects, and optimization of projects can be conducted either manually or automatically. Manual approaches to select projects are for example the Analytic Hierarchy Process (AHP), Q-Sort, scoring models, and portfolio matrices [1]. Commonly used manual approaches are based on some sort of direct comparison and ranking of the alternatives based on project data and individual preferences. Manually

conducting the selection of projects is restricted in the number of projects it can deal with, constraints and objective preferences which can be taken care of, as well as the number of objectives the decision makers are able to optimize. This is especially true, since the complexity grows exponentially with the number of available project alternatives [5]. The project selection problem is NP-hard problem [5] thus there is no exact algorithm that solves larger instances of this problem to proven optimality. Hence, heuristic algorithms are an option for finding at least approximate solutions to the optimal ones.

The complexity of the project selection problem is based on the often high number of projects from which a subset has to be chosen, various existent restrictions and a multitude of objective preferences which the optimal portfolio should adhere to. Common goals of portfolio optimization are maximization of potential revenue and strategic alignment, as well as minimization of negative synergy effects in-between projects selected for a portfolio. Furthermore, portfolios often have to be balanced regarding characteristics of the portfolios. These balancing requirements are found in the number of projects assigned to a specific category, or in the optimization of a portfolio regarding its overall mean risk value.

Manual approaches to this demanding decision problem are restricted in their usefulness for the problem at hand. Promising alternatives to manually selecting projects are found in the field of mathematically based portfolio optimization, and multiobjective optimization. Automated approaches are superior to manual

* Corresponding author. Tel.: +43 676847974600.

E-mail addresses: thomas.kremmel@aeon-group.at, thomas.kremmel@gmail.com (T. Kremmel), kubalik@labe.felk.cvut.cz (J. Kubalík), stefan.biffl@tuwien.ac.at (S. Biffl).

approaches in a way that they are able to create a set of efficient portfolios, for which it can be assured that there exist no solutions in the search space which promises better values in at least one of the objectives, and offers at least the same in all the others. This set of efficient solutions is commonly referred to as the Pareto-optimal front. Furthermore, the utilized computational power and sophisticated multiobjective meta-heuristics ensure that the solutions found do have at least the same quality as manually created portfolios. With a high probability, solutions on the Pareto-optimal front found by automated approaches are much better than solutions found with manual approaches. Furthermore, manual approaches will never be able to keep up with the search ability of automated approaches, and the number of considered projects, objectives, restrictions and constraints.

This paper presents the first phase of a decision support framework, which focuses on the selection of projects, using an evolutionary optimization algorithm. An implementation of a nature-inspired optimization framework is presented, using the evolutionary algorithm mPOEMS, based on general goals which the portfolio management should support. This optimization framework is able to find efficient portfolios on or close to the Pareto-optimal front. The second phase, not presented in this paper, comprises a group decision process to efficiently select one portfolio out of the Pareto-optimal front.

mPOEMS is an iterative optimization algorithm that seeks in each iteration for such a modification of the current solution, called prototype, that improves its quality the best. Modifications are represented as sequences of elementary actions (simple mutations in standard EAs), defined specifically for the problem at hand. mPOEMS uses an evolutionary algorithm to search for the best action sequence that can be considered an *evolved hypermutation*.

We have chosen mPOEMS since it was recently shown to be suitable for solving discrete multiobjective optimization problems like multiobjective 0/1 knapsack problem [10]. In [10] mPOEMS was shown to produce better or at least competitive results to the state-of-the-art MOEAs such as NSGA-II and SPEA2 on that problem.

The iterative optimization concept based on the evolved hypermutations has proven to be successful approach also for solving hard single-objective combinatorial problems such as the sorting network problem [11], the multiple sequence alignment problem [12] and the shortest common supersequence problem [13].

The presented approach is evaluated on a test set of 50 IT projects. COCOMO II is used to calculate the effort and schedule data of the utilized project pool. The optimization framework claims to consider a very broad range of portfolio optimization characteristics and goals found in the literature. It has been empirically verified that mPOEMS is able to generate high-quality solutions with respect to the given set of optimization objectives. Results of the multiobjective optimization using mPOEMS are compared with results found with the state-of-the-art multiobjective evolutionary NSGA-II and SPEA2.¹ Analyses presented in this paper show, that mPOEMS performs significantly better than NSGA-II and SPEA2 on this problem.

2. Related work

Various approaches have already proven the applicability of mathematically based meta-heuristics for the project selection problem.

In 2000, Ghasemzadeh and Archer [6] presented a decision support system, which follow the process steps of Archer's integrated framework for PPM [1], and combine them with a formal model,

considering various variables and constraints. The possibility to making manual adjustments is given through adding or removing projects. A very interesting contribution is the consideration that projects can start in an arbitrary timeframe, given that the project is completed within the planning horizon.

In 2003, Stummer and Heidenberger presented a three-phase approach [19] for PPM in the field of research and development. With the use of an Integer Linear Programming Model, they provided a process to deal with non-numerically restricted project interdependencies, Logical & Strategic constraints as well as Resource & Benefit constraints. One of the main contributions is the examination of resources and project expenditures in their corresponding timeframes. In comparison to the discounting of project attributes to a certain point in time, timeframes deliver a more accurate image of the business environment [19]. Disadvantages of this approach compromise its inability to deal with incomplete data sets, and the process or time-related restriction on approximately 30 projects [19].

In 2004, Doerner et al. published a so-called *Pareto Ant Colony Optimization Approach* [5]. Artificial ants construct valid project portfolios and take into account complex project interactions. The consideration of project synergy, the good results it has shown in experiments and the possibility of easily adding heuristic information to the algorithm makes it a valuable approach [5].

In 2005, Medaglia et al. presented an evolutionary approach for project selection [15]. They utilized a genetic algorithm, combined with random parameter values to model project selection under uncertainty. To model uncertainty is of specific relevance, because it is often difficult in the project selection stage to define parameter values exactly, due to various internal and external influences (risk, political issues, interest rates, economic changes, and so on). Uncertainty could be modelled as specific random values following triangular, exponential, and Erlang distributions like in [15], or with the use of likely values as proposed in [1].

Closely related to the work presented in this paper is the work by Gueorguiev et al. [7], where a software project planning is formulated as bi-objective optimization problem with robustness and project completion time treated as two competing objectives.

Each of the above-presented approaches add valuable information and knowledge to the topic of creating optimized project portfolios. The approaches have been analysed and inspired the authors of this paper in creating an evolutionary-based approach to the project selection problem. Some ideas of these works are used in the presented approach, and expanded with new methods.

The next section presents the list of goals the presented project selection portfolio optimization approach supports. The list of goals is based on a thorough investigation of the corresponding literature and discussions with experts in the field of portfolio management.

3. Portfolio optimization goals

The following listing describes the goals which an optimization approach for the project selection problem should support:

1. Consider and limit the available resources/budget per timeframe.
2. Support "must-select" restrictions.
3. Take synergy effects into account.
4. Take logical relationships into account.
5. Maximize the overall strategic alignment value.
6. Support balancing of risk, project categories and return time.
7. Maximize potential portfolio return.
8. Provide possibility to define project starting timeframes.

Goal Nr. 1 deals with the limited availability of resources in an organisation. A project needs a certain amount of resources across

¹ The PISA framework [22] was used to adapt NSGA-II and SPEA2 to the presented portfolio optimization approach.

its life cycle. Initial investments and various running costs have to be taken into account in the cost planning process of project management, resulting in a budget plan for each project. This budget plan should be used in a portfolio optimization process to match the selected projects in the portfolio with the overall budget of an organisation, imposed by the strategic planning group. Since the discounting of budget data to a net present value diminishes the richness of the information [18], the budget data should be investigated in the corresponding timeframe.

Goal Nr. 2 deals with the fact that because of legal and economic circumstances, or the executives' will, a project has to be included into any valid portfolio. Therefore, it should be possible to define a "must-select" restriction for projects [14].

Goal Nr. 3 deals with synergy effects that could arise if 2 projects are selected for the same portfolio. These synergy effects could be negative or positive and are considered in the Pareto ant colony optimization approach presented in [5].

Goal Nr. 4 deals with logical relationships between projects. Logical relationships are either the obligatory selection of one or more predecessor projects if a certain project is selected for a portfolio, or a restriction that 2 projects may not be selected for the same portfolio and thus are mutually exclusive. Project interdependencies are considered in [6,19].

Goal Nr. 5 deals with the strategic alignment of projects an organisation conduct. The strategy of an organisation has been defined as a set of approaches to attain certain goals on the business level. According to Reyck et al. [17], organisations at the lowest level of PPM adoption, face problems with commitment of business leaders, poor alignment of projects to strategy, little coordination between projects and conflicting project objectives. The selection of projects under consideration of strategic alignment of the selected projects is of major importance for a business, because the nature and direction of an organisation is defined by the conducted activities [8]. If the projects are not selected in consideration of the strategy, it is probable that the organisation is moving towards an unintended direction, overall goals are not attained and scarce resources are wasted. In order to select projects based on their strategic alignment, a method to quantitatively measure the strategic alignment of projects has to be used. Since there exists a gap of tools to exactly measure in quantitative terms the alignment of projects to the strategy, identified in [8], the strategic alignment values of the project test set is calculated using a scoring model with weighted strategies. This approach creates quantitative strategic alignment values which can be used in the presented optimization approach as another objective to maximize. Since these values are qualitative in nature it is recommended to use sophisticated strategic alignment calculations which quantitatively measure the strategic alignment value of projects.

Goal Nr. 6 deals with the need to create portfolios which are balanced regarding certain characteristics. The strategy plan provides the management of the portfolio not only with goals and approaches to attain this goals, but also with the direction to balance the portfolio. This direction is defined in terms of risk affinity, short-term versus long-term returns, available resources, key criteria, or in risk-to-benefit trade-off preferences [16]. Furthermore categories are used to group projects together which share a common goal or have the same measurement criteria. An optimization approach for project selection should provide a method to create portfolios balanced in the above-mentioned characteristics. Balancing methods are used in [5,6,19].

Goal Nr. 7 deals with the need to maximize potential overall portfolio return. The project portfolio is in fact an investment portfolio. An organisation invests human resources, knowledge and money into a project, with the goal of attaining benefits from this investment. Like the shares for an investment portfolio, the potential projects for the project portfolio have to be evaluated in regard

to their potential financial revenue. As in an investment portfolio, the ultimate goal is to maximize potential overall welfare, while adhering to constraints and risk preferences [14]. The financial metrics ROI and NPV are commonly used to analyse the potential financial performance of projects. The calculation of these metrics is part of the planning process in project management. The use of the NPV metric for portfolio selection is questioned because it treats short-term returns as more favourable than long-term returns. Furthermore, the nature of this metric prefers long-term investments to short-term investments [14]. This could lead to an asset of short-term projects in the project selection process, even if short-term projects are not favoured by the organisation. While the selection of an appropriate financial metric is organisation-dependent, the maximization of the overall portfolio is basically always the same. The project selection optimization process of PPM will always try to maximize the financial metrics which serve as an input to the selection process. Revenue maximization is considered in all investigated mathematical portfolio optimization approaches.

Goal Nr. 8 deals with the need to consider project starting timeframes. In [6] an approach for project selection is presented, where projects are considered as investments that can be started in an arbitrary timeframe, given that the project is finished before the planning horizon. It is of major importance to recognise that most projects cannot start in an arbitrary timeframe, but very often in a few distinct timeframes. It is also possible that a project can only start in one timeframe. For example, to meet a market opportunity. Each of these cases

- Start a project in an arbitrary timeframe,
- Start a project in a few distinct timeframes,
- Start a project in exactly one timeframe,

has to be considered in the process of creating valid portfolio candidates. The restriction that projects have to be finished in the planning horizon will be adhered, because the budget and resource restrictions for timeframes not included in the planning horizon are not defined, and would therefore lead the whole planning concept ad absurdum. In considering starting timeframes of projects and presenting a method to deal with the sequencing of projects, a tool is created to not only to optimize the selection of projects, but as well to optimize the sequencing of projects. The presented portfolio optimization approach does consider all the goals described above.

4. Formal model

The presented project selection problem is similar to the multiobjective knapsack problem as used in [20]. The designated goal of the original multiobjective knapsack problem is to find a subset of items for an arbitrary number of knapsacks, which maximize the profit for each knapsack, while the maximum capacity of all knapsacks under consideration may not be exceeded. A solution is represented by a vector with the length of the maximum available items, where position i is set to 1 if item i is selected, and 0 if not. Here, an optimal set of projects from a pool of all potential projects is sought just like the set of items in the knapsack problem. Additionally, the information assigned to each project not only specifies whether the project is selected ($x_i > 0$) or not ($x_i = 0$), but also specifies when the project starts (value of x_i greater than 0). The planning horizon of a portfolio is divided into timeframes, and timeframes are in turn subdivided into smaller units. These units can be chosen arbitrarily. Months are used here for these units, and one year for the timeframes. The project start is declared by selecting a month for each project from the available planning horizon.

Formally, the task of the project selection problem is defined as follows:

Find a vector $x = (x_1, x_2, \dots, x_p) \in M_1 \times \dots \times M_p$,
 where $M_i \subseteq M$,
 $M = \{0, 1, 2, \dots, T \times 12\}$,
 such that $y = (q_1(x), q_2(x), \dots, q_5(x))$ is maximum,

where x_i is greater than 0 if project i is selected, and 0 if not, p is the number of projects in the candidate pool, M the number of months in the planning horizon, M_i the months in which project i can start, T the number of timeframes in the planning horizon, and $q_n(x)$ are the five measures of portfolio quality (optimization objectives) defined as follows:

1. *Potential revenue* ($q_1(x)$). The project portfolio is in fact an investment portfolio. An organisation invests human resources, knowledge and money into a project, with the goal of attaining benefits from this investment. Like the shares for an investment portfolio, the potential projects for the project portfolio have to be evaluated in regard to their potential financial revenue. Thus, this objective deals with the need to maximize potential overall portfolio return. It is calculated as

$$q_1(x) = \sum_{i=1}^p f_i(x)$$

where $f_i(x) = r_i \cdot w_i$ and r_i is the potential financial revenue of project i , and w_i is 1 if $x_i > 0$, and 0 if $x_i = 0$. The greater the overall potential revenue, the better the solution.

2. *Strategic alignment* ($q_2(x)$). This objective deals with the strategic alignment of projects an organisation conduct. The strategy of an organisation has been defined as a set of approaches to attain certain goals on the business level. According to Reyck et al. [17], organisations at the lowest level of PPM adoption, face problems with commitment of business leaders, poor alignment of projects to strategy, little coordination between projects and conflicting project objectives. The selection of projects under consideration of strategic alignment of the selected projects is of major importance for a business, because the nature and direction of an organisation is defined by the conducted activities [8]. Thus, the strategic alignment on the portfolio level should be maximized. We calculate the overall strategic alignment as

$$q_2(x) = \sum_{i=1}^p s_i(x),$$

where $s_i(x) = a_i \cdot w_i$ and a_i is the strategic alignment value of project i . The greater the overall strategic alignment value, the better the solution.

3. *Resource usage distribution metric* ($q_3(x)$). This objective is introduced in order to create selection pressure towards the selection of portfolios which on the one hand have a high resource usage per timeframe, and on the other hand have the best distribution between the timeframes. Its value is between 0 and 1, where 1 means full resource consumption in each timeframe, and 0 means that at least in one timeframe no resources are consumed. Consequently, the objective function to maximize, may be expressed as,

$$q_3(x) = \prod_{t=1}^T \left(\frac{\sum_{o=1}^l \sum_{i=1}^p r_{o,t,i} \cdot w_i}{\sum_{o=1}^l R_{o,t}} \right)$$

where o is the type of a resource (considering l different resource types), t the timeframe, T the number of timeframes in the planning horizon, $r_{o,t,i}$ the type o resource consumption of project i in timeframe t , and $R_{o,t}$ the type o resource limit in timeframe t . The closer the q_3 is to one, the better the solution. If a portfolio consumes more resources than are available a simple greedy repair algorithm removes projects until all resource constraints are fulfilled. Then, projects with the worst potential revenue/needed resources ratio are removed first.

4. *Risk* ($q_4(x)$). The risk objective expresses a need to minimize a deviation of a median portfolio risk value from a desired risk range. Thus selection pressure towards portfolios with a median risk value in the desired risk range is induced. This objective is calculated as

$$q_4(x) = 1 - d(x)$$

where $d(x)$ is the percentage deviation of the median portfolio risk value from the desired risk range. The closer the value is to 1, the better the solution. Portfolios with a median risk value in the range of the desired risk value will have their risk objective set to 1. Portfolios with a median risk value not in the desired risk range will be punished having the risk objective value set to a value lower than 1.

5. *Synergy* ($q_5(x)$). This quality measure expresses the positive and negative effects among selected projects where the positive synergy is to be maximized while the negative synergy is to be minimized. For this purpose, the sum of negative synergy is subtracted from the sum of positive synergy, consequently referred to as the synergy of a solution.

The positive synergy effect is calculated as

$$y^+(x) = \sum_{i=1}^p y_i^+(x),$$

where

$$y_i^+(x) = \sum_{k=1}^{|K_i|} y_{i,k}^+ \cdot w_i \cdot w_k$$

and y_i^+ is the positive synergy value of project i concerning project k , K_i is a set of projects that have a potential synergy with project i , w_i is 1 if project i is selected, and 0 if not, w_k is 1 if project k is selected, and 0 if not.

Negative synergy effects $y^-(x)$ are calculated the same way and consequently subtracted from the sum of positive synergy creating the combined maximization synergy objective

$$q_5(x) = y^+(x) - y^-(x).$$

The greater the synergy objective, the better the solution.

Additionally objective criteria are considered as goals to optimize combined with restrictions which define the restrictive characteristic of the criterion. This approach ensures the creation of portfolios which adhere to various preferences, for example number of projects assigned to a certain category, and provide the portfolio manager with a tool to restrict and relax the optimization pressure of these criteria.

5. Multiobjective optimizations

The presented project selection problem is a typical multiobjective optimization problem where solutions are sought such that

they are optimal with respect to all, often conflicting objectives. Typically, no single optimal solution can be found that would be superior to all other solutions with respect to all objectives. Instead, a set of optimal solutions, for which it holds there is no other solution in the whole search space that is superior to these solutions with respect to all objectives, is considered the output of the optimization. A good multiobjective optimization technique must be able to search for a set of optimal solutions concurrently in a single run. This is the reason why EAs become very popular in the domain of multiobjective optimization as they are able to evolve a diverse population of high-quality solutions in parallel.

Perhaps the most widespread and successful are multiobjective evolutionary algorithms (MOEAs) that use a concept of dominance for ranking of solutions. By definition [3], a solution x dominates another solution y , if the solution x is no worse than y in all objectives and the solution x is strictly better than y in at least one objective. The concept of dominance can be used to divide any finite set S of solutions into two non-overlapping sets, the non-dominated set $S1$ and its complement set, the dominated set $S2$. The set $S1$ contains all solutions that are not dominated by any other solution of the whole set S . The set $S2$ contains solutions that are dominated by at least one solution of $S1$. If the set S is the whole feasible search space then the set $S1$ is a set of optimal solutions called Pareto-optimal solutions and the curve formed by joining these solutions is called a Pareto-optimal front.

In the absence of any higher-level information, all Pareto-optimal solutions are equally important [3]. That is why the goal in a multiobjective optimization is to find a set of solutions that is (i) as close as possible to the Pareto-optimal front and (ii) as diverse as possible so that the solutions are uniformly distributed along the whole Pareto-optimal front.

Examples of successful MOEAs are the Non-dominated Sorting Genetic Algorithm II (NSGA-II) [4] and the Strength Pareto Evolutionary Algorithm 2 (SPEA2) [21]. They represent state-of-the-art multiobjective optimization evolutionary algorithms, so we used them for comparison with the proposed mPOEMS approach. SPEA2 uses a regular population and an archive (a set of constant size of best solutions found so far). An archive truncation method guarantees that the boundary solutions are preserved. Fitness assignment scheme takes for each individual into account how many individuals it dominates and it is dominated by which is further refined by the incorporation of density information. NSGA-II defines a density estimation metric called *crowding distance* as the largest cuboid enclosing a solution without including any other solution in the population. Then, so-called *crowding comparison operator* guides

the selection process towards solutions of the best non-dominance rank and with largest crowding distance in order to maintain a good spread of solutions.

6. Proposed mPOEMS approach

This section presents the proposed multiobjective optimization approach based on the Prototype Optimization with Evolved Improvement Steps (mPOEMS) [10]. First, a single-objective version of the algorithm is introduced, then its extension for the multiobjective problems and finally the adaptation of mPOEMS to the project selection problem is described.

6.1. POEMS

Prototype Optimization with Evolved Improvement Steps (POEMS) [9] is an iterative optimization approach that employs an EA for finding the best modification of the current solution, called prototype, in each iteration. Modifications are represented as a sequence of fixed length of primitive actions defined specifically for the problem at hand. Besides actions that truly modify the prototype, there is also a special type of action called nop (no operation), which has no effect on the prototype. The evolved action sequences can contain one or more instances of the nop action. Action sequences are assessed based on how well/badly they modify the current prototype, which is passed as an input parameter to the EA. Moreover, sequences that do not change the prototype at all are penalized to avoid a convergence to useless trivial solutions. After the EA finishes, it is checked to determine whether the best evolved sequence improves the current prototype or not. If an improvement is achieved, then the modified prototype is considered a new prototype for the next iteration. Otherwise the current prototype remains unchanged. The iterative process stops after a specified number of iterations. In other words, the POEMS can be considered as an iterative algorithm with *evolved hypermutations*.

6.2. mPOEMS

mPOEMS is a multiobjective extension of the original single-objective POEMS algorithm. Just like NSGA-II and SPEA 2, mPOEMS belongs to the class of multiobjective optimization algorithms that use the concept of dominance. The main differences between mPOEMS and POEMS are

- mPOEMS maintains a set of best solutions found so far, called a solution base, not just a single prototype solution as in POEMS.
- mPOEMS uses a multiobjective EA based on the dominance concept instead of a simple EA.

Algorithm 1: Multiobjective Prototype Optimization with Evolved Improvement Steps

```

1 begin
2    $i \leftarrow 0$ 
3    $SolutionBase^{(i)} \leftarrow InitializeSolutionBase()$ 
4   while not TerminationCondition() do
5      $Prototype^{(i)} \leftarrow ChoosePrototype(SolutionBase^{(i)})$ 
6      $ActionSequences \leftarrow MOEA(Prototype^{(i)}, SolutionBase^{(i)})$ 
7      $NewSolutions \leftarrow ApplyTo(ActionSequences, Prototype^{(i)})$ 
8      $SolutionBase^{(i+1)} \leftarrow Merge(NewSolutions, SolutionBase^{(i)})$ 
9      $i \leftarrow i + 1$ 
10  return  $SolutionBase^{(i)}$ 
11 end
```

The solution base consists of solutions to the problem at hand. mPOEMS improves the solution base in an iterative process, see Algorithm 1. At the beginning of each iteration of mPOEMS one solution from the set of non-dominated solutions of the current solution base is chosen as the current prototype, for which the action sequences will be evolved by the EA. Each MOEA run starts with action sequences generated at random that are evolved in generations using selection, crossover and mutation steps. New solutions generated by applying the action sequences found by the MOEA to the prototype are merged with the current solution base resulting in a new version of the solution base. Only unique solutions are stored in the solution base. Procedures `ChoosePrototype()` and `Merge()` are designed with the aim to keep the set of non-dominated solutions in the solution base as diverse as possible. In `ChoosePrototype()`, the prototype is chosen among non-dominated solutions in the solution base so that the non-dominated front of the evolved solution base is sampled uniformly. Thus, no “inner” part of the non-dominated front is explicitly preferred. An exception from this rule are the extremal solutions that have the best value of some objective observed in the whole solution base. These boundary solutions have in fact the highest probability of being selected, which helps the algorithm to keep the non-dominated front of the evolved solution base wide. `Merge()` procedure merges solutions produced by action sequences evolved by the MOEA with the current solution base. It uses the crowding distance introduced in [4] in order to retain the most unique solutions in the solution base.

At the end, the set of non-dominated solutions of the final solution base is taken as the result of the run. For the details of the mPOEMS algorithm please see [10].

6.3. Adaptation of mPOEMS to the project portfolio optimization problem

In this work, each solution is represented by a vector

$$s = (x_1, x_2, \dots, x_p),$$

where each position x_i describes whether the project i is selected ($x_i > 0$) or not ($x_i = 0$). Value ($x_i > 0$) specifies in which timeframe the project should start.

For the project selection problem the following two actions were introduced:

- `changeMonth projectId, month` – This action changes the starting month of a project randomly to another month, with respect to the valid timeframes for the project under consideration. Initialization and the mutation of this action is done randomly, with the restriction, that the month value has to be selected, out of the set of valid starting months for the project under consideration.
- `switchOnOff projectId` – This action selects a project if it is unselected, and vice versa. It was introduced in order to increase a probability that certain project will not be selected to the portfolio. Note that projects have usually many more than one starting possibility. Thus, the value zero, which means the project is not selected, has a very low probability of being selected for the `changeMonth()` action parameter. Initialization and the mutation of this action is done randomly, with the restriction that this action will not be applied to projects which are mandatory.

Simple crossover and mutation operators are used in the MOEA to generate new population of action sequences from the current one. The crossover operator is a uniform-like operator proposed in [10]. Each gene of both parents have the same probability of being selected for the offspring. Valid offspring constitutes an arbitrary combination of parental actions, whereas each gene can be used

only once. The mutation operator either changes the action type (`nop`, `changeMonth()`, `switchOnOff()`) or the parameters of an action (`projectId` or `month`).

7. Experimental results

This section presents the experiments carried out to evaluate the performance of the proposed approach. First, the test data set based on the Constructive Cost Model (COCOMO II) is described. Then, the configuration of mPOEMS and the compared algorithms used in the experiments is shown and two performance measures that we used for evaluation of the algorithms performance are described. The last part of this section is devoted to the analysis of the achieved results.

7.1. COCOMO II test set

The test set consists of 50 software projects. The project size predominantly determines the cost of a software project. Source lines of code (SLOC) are used to describe the size. The projects are restricted to having at most 37,000 SLOC and at least 1000 SLOC. The maximum duration of a project is restricted to 18 months. The planning horizon is set to 3 years. The planning horizon is divided into timeframes of the length of a year, resulting in three timeframes. In each timeframe, the available resources are restricted to 500 person-months (PM), resulting in a total of 1500 PM for the planning horizon. This value equals 0.6% of the resources needed to conduct all projects in the project pool. Each project has an assigned risk value between 0.20 and 0.80. Since the cost per PM serves as input for the calculation of the potential revenue of a project, this value is set to 5000 Euro per PM. Potential revenue is set to a maximum of 150%, and to the minimum of 85% of the initial costs. Each project is randomly assigned to exactly one category. The strategic alignment value is calculated following a weighting approach where the alignment value to each strategy was set randomly, resulting in an overall strategic alignment value. A maximum number of 30% of all projects is selected to have synergy effects with exactly one project, divided into 15% of the positive synergy and 15% of the negative synergy. Projects which have synergy effects, as well as the projects which have to be selected for the same portfolio in which the synergy effects will take effect, will be selected randomly. Each synergy effect is restricted to be at a maximum of 15% of the total cost of the project, which will trigger the synergy effect. A number of 10% of all projects is selected randomly to be mandatory and 4 projects are manually selected to be mutually exclusive. The well accepted cost estimation framework COCOMO II (Constructive Cost Model) was used to generate the cost, schedule and effort part of the test set. COCOMO II is a framework for software project cost and schedule estimation, and provides the user with tools for estimating the likely cost, effort and time needed to conduct a software project [2]. Framework parameters were set randomly and are normally distributed. The PM effort per phase was mapped to PM effort per month and costs per month have been calculated.

The test data set, including the project interdependencies, can be found at <http://labe.felk.cvut.cz/kubalik/ProjectPool.xml>.

7.2. Experimental setup

In this work we compare the proposed mPOEMS approach with two other state-of-the-art multiobjective optimization evolutionary algorithms – NSGA-II [4] and SPEA2 [21].

NSGA-II and SPEA2 use binary representation of length 50. Each position in the chromosome indicates whether the respective project is or is not selected to the portfolio. Both algorithms use standard 2-point crossover and simple bit-flipping mutation.

NSGA-II and SPEA2 algorithms use the same setting of control parameters:

- population of size 500,
- number of generations 600,
- crossover probability 0.7,
- mutation applied to 1 position per chromosome,
- tournament selection with parameter 2,
- 250 solutions are generated by crossover and mutation in each generation.

The mPOEMS was used with the following setting:

- solution base of size 500,
- number of iterations 150,
- length of evolved action sequences 20,
- population of size 50,
- number of generations 20,
- crossover probability 0.75,
- mutation probability 0.15,
- tournament selection with parameter 4.

All of the compared algorithms were allowed to generate 150,000 candidate solutions during one run.

7.3. Performance measures and statistical tests used for evaluation

For each algorithm k independent runs were carried out and the final non-dominated sets $NDS_{final}^{(a)}(1), \dots, NDS_{final}^{(a)}(k)$ of these individual runs were collected for each algorithm $a \in \{mPOEMS, SPEA2, NSGA-II\}$. They were then merged in the set $B_{final}^{(a)} = NDS_{final}^{(a)}(1) \cup NDS_{final}^{(a)}(2) \cup \dots \cup NDS_{final}^{(a)}(k)$ and a set $NDS_{final}^{(a)}$ of non-dominated solutions of that compound set $B_{final}^{(a)}$ was found.

Besides the $NDS_{final}^{(mPOEMS)}$ also a set of non-dominated solutions denoted as $NDS_{initial}^{(mPOEMS)}$ was derived from a collection of non-dominated solutions of the randomly generated initial solution base gathered over all runs.

A performance of the compared algorithms is assessed based on the quality measures derived from sets $NDS_{final}^{(a)}(i)$ and $NDS^{(a)}$ achieved by the algorithms. Two performance measures were used in this work:

- *Coverage of two sets* $C(X, Y)$, proposed in [20]. The measure is defined in the following way: Given the two sets of non-dominated solutions found by the compared algorithms, the measure $C(X, Y)$ returns a ratio of a number of solutions of Y that are dominated by or equal to any solution of X to the whole set Y . Thus, it returns values from the interval $[0, 1]$. The value $C(X, Y) = 1$ means that all solutions in Y are covered by solutions of the set X . And vice versa, the value $C(X, Y) = 0$ means that none of the solutions in Y are covered by the set X .
- *Hypervolume metric* $H(X)$ also known as *Size of the space covered*, proposed in [20]. A reference volume between the origin and an utopian objective vector (defined by the profit sums of all items in each objective) is taken into account. This measure is defined as a fraction of that volume that is not dominated by the final non-dominated solutions. So, the smaller the value of this measure the better the spread of solutions is, and vice versa.

The $C(X, Y)$ measure was applied to final $NDS_{final}^{(a)}$ sets. The $H(X)$ measure was applied to individual sets $NDS_{final}^{(a)}(i)$. Mean and median values over the set of $H(NDS_{final}^{(a)}(i))$, for $i = 1, \dots, k$, were

calculated for each algorithm. Statistical significance of differences between the mean and median values was checked by means of two-sample t -test and Wilcoxon rank-sum test, respectively. In both cases, the null hypothesis (that the means and medians are equal) is tested at the 1% significance level. $H(NDS_{final}^{(a)}(i))$ values were also visualized using notched boxplots. Boxes whose notches do not overlap indicate that the medians of the two groups differ at the 5% significance level.

7.4. Analysis

We run 60 independent runs with each of the compared algorithms where each run produced a set of non-dominated solutions. In all cases the final population of solutions (a solution base in case of mPOEMS) contained only non-dominated solutions.

Firstly, the ability of the mPOEMS algorithm to generate solutions close to the Pareto-optimal front is illustrated by plots in Fig. 1, where the progress made by the algorithm between the initial and the final solution base is shown. In this figure, a distribution of $NDS_{initial}^{mPOEMS}$ (red dots) and NDS_{final}^{mPOEMS} (blue dots) sets is visualized for all 2-objective projections (For interpretation of the references to color in this sentence, the reader is referred to the web version of the article.). We can see that in all cases solutions from the NDS_{final}^{mPOEMS} set are considerably closer to the upper-right corner² than solutions from the $NDS_{initial}^{mPOEMS}$ set.

Another interesting observation is that objectives q_1 and q_3 are positively correlated, see Fig. 1b. However, this is not a surprising observation. It is quite natural that expected revenue depends on a proper utilization of resources, thus high revenue can be expected only when resources are effectively used (distributed) throughout the whole planning horizon and vice versa. Generally, observations of this kind are very valuable as they can reveal and help to understand intrinsic dependencies and relations of the problem at hand.

When the coverage measure is considered, the situation changes in favour of mPOEMS. First, mPOEMS generates the largest set of non-dominated solutions $NDS_{final}^{(a)}$ out of the three compared algorithms. It has found 2938 unique non-dominated solutions in total while SPEA2 and NSGA-II only 2833 and 2491, respectively.

More importantly, $C(NDS_{final}^{(mPOEMS)}, NDS_{final}^{(Opponent)})$ is significantly bigger than $C(NDS_{final}^{(mPOEMS)}, NDS_{final}^{(Opponent)})$ for both opponent algorithms SPEA2 and NSGA-II, see Table 2. This means that the number of final solutions generated by SPEA2 and NSGA-II, which are dominated by mPOEMS final solutions is bigger than the number of mPOEMS solutions that are dominated by solutions of SPEA2 and NSGA-II, respectively.

So, the SPEA2 can be considered to outperform the other two algorithms with respect to the hypervolume measure while the mPOEMS algorithm is the best algorithm with respect to the coverage measure.

This is a very common situation in multiobjective optimizations since the performance of different algorithms cannot be explicitly compared due to the fact that a quality of the sets of final solutions, not a quality of single solutions, has to be judged [3].

Let us take a deeper look at the obtained results now. The three compared algorithms assume that all of the non-dominated solutions are equally important. In fact, this is not truly realistic assumption in this case. Note, that a user is not interested in just any non-dominated solutions including those that are doing extremely well with respect to some objective(s) while doing very

² Note, since all of the objectives are to be maximized the upper-right corner of each 2D projection of the solution set represents the region of interest.

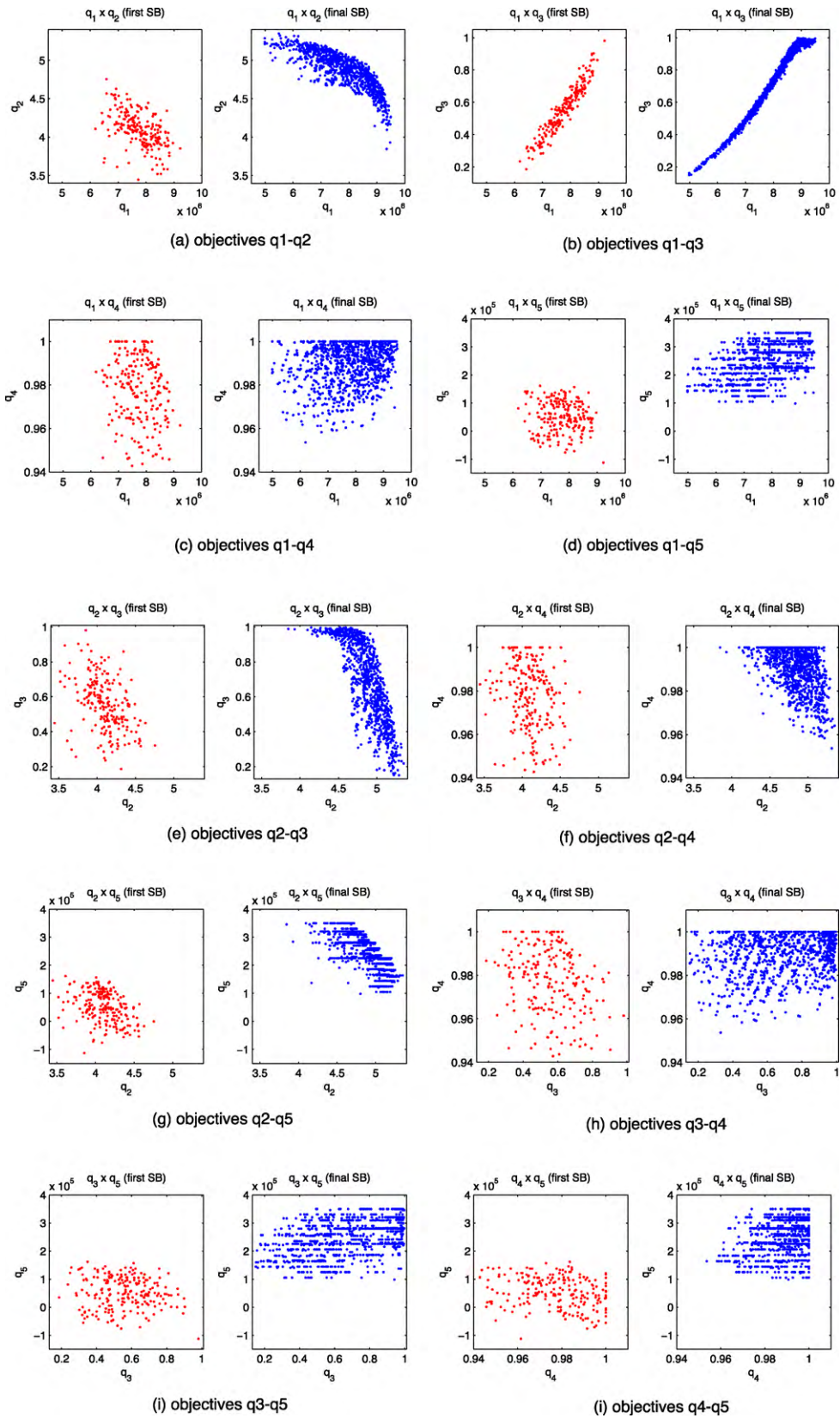


Fig. 1. Illustration of an improvement of the final solution base w.r.t. the initial solution base. For the sake of easy visualization 2D projections of the non-dominated solutions are shown in a couple of separate plots – the left one for the initial solution base and the right one for the final solution base – for all pairs of objectives.

Table 1

Average, median and the standard deviation values calculated from 60 hypervolume values produced by the compared algorithms. First row shows statistics calculated from all non-dominated solutions, second row shows statistics calculated from solutions composed only from the 1st quartile objective values. Bold numbers indicate that the value has been proven to be statistically the best according to the *t*-test and Wilcoxon rank-sum test at the 1 % significance level.

	SPEA2			NSGA-II			mPOEMS		
	Avg	Median	Stdev	Avg	Median	Stdev	Avg	Median	Stdev
All data	0.230	0.227	0.015	0.233	0.233	0.004	0.234	0.234	0.008
1st quartile data	0.405	0.401	0.019	0.461	0.465	0.031	0.373	0.374	0.012

Table 2

Coverage statistics calculated from the final $NDS_{final}^{(a)}$ sets.

	All data			1st quartile data		
	mPOEMS	SPEA2	NSGA-II	mPOEMS	SPEA2	NSGA-II
mPOEMS	–	0.438	0.621	–	0.607	0.869
SPEA2	0.348	–	0.632	0.260	–	0.692
NSGA-II	0.177	0.198	–	0.067	0.156	–

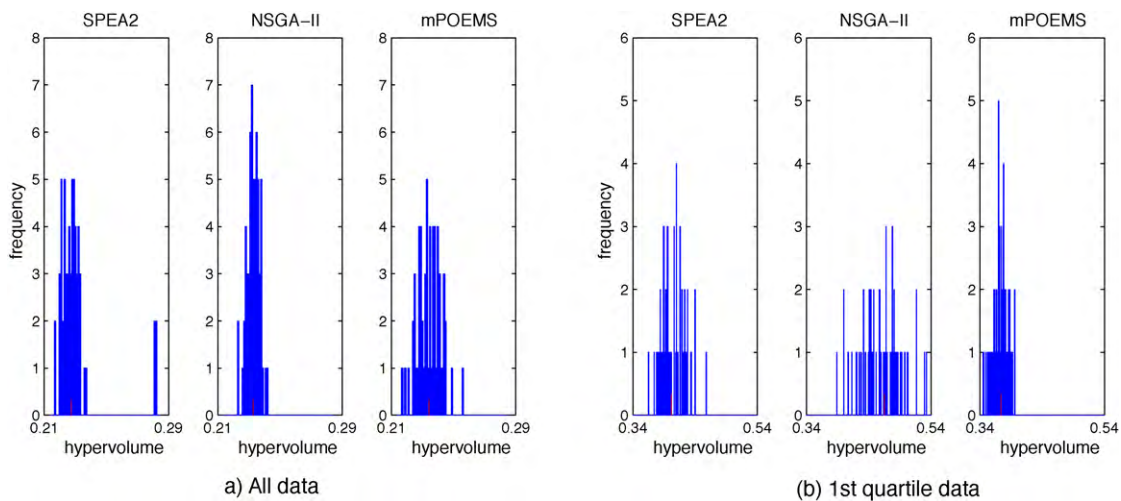


Fig. 2. Distributions of hypervolume values obtained on 60 runs with each of the compared algorithms. (a) is a histogram calculated from all non-dominated solutions, (b) is the histogram calculated from solutions composed of 1st quartile objective. With reference to the color information in the artwork, a parenthetical sentence, “For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.” has been incorporated in the caption to Fig. 2. Please check values. Mean values are indicated by red tick marks. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.)

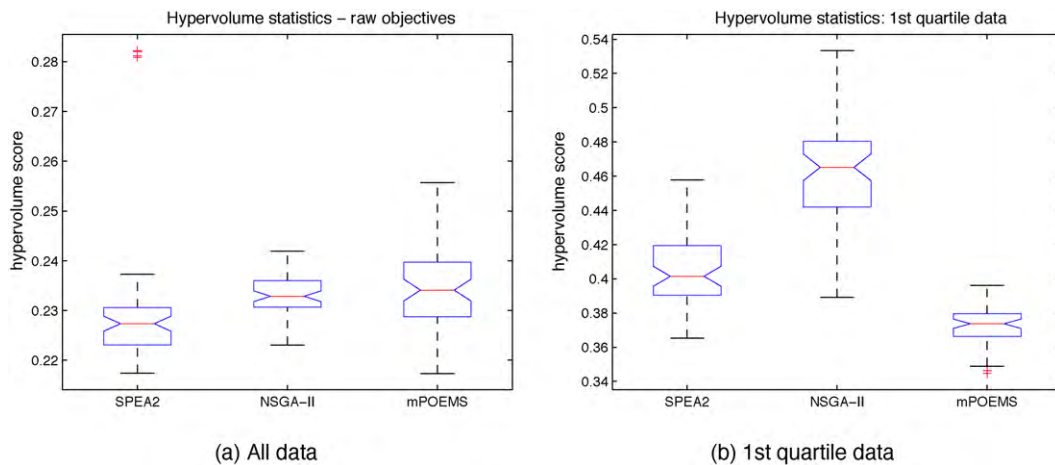


Fig. 3. Notched boxplots generated from 60 hypervolume values obtained for SPEA2, NSGA-II and mPOEMS. (a) shows boxplots generated from all non-dominated solutions and (b) shows boxplots generated from solutions composed of 1st quartile objective values.

bad in other objective(s) at the same time. Instead, the user should rather prefer non-dominated solutions which are not inferior with respect to any of the considered optimization objectives. Hence, in the following analysis only the solutions whose all objective values belong to the best 75% values observed in $NDS_{final}^{(a)}(i)$ for all $i = 1 \dots 60$ and $a \in \{\text{mPOEMS, SPEA2, NSGA-II}\}$ are considered. So the non-dominated solutions that have some objective value really bad (not belonging to the 1st quartile range) are filtered out. Statistics calculated for these post-processed sets $NDS_{final}^{(a)}$ are presented in Figs. 2 and 3, and Tables 1 and 2, denoted as 1st quartile data. One can see that a dominance of mPOEMS over SPEA2 and NSGA-II with respect to the coverage measure is even bigger now. Note that mPOEMS generated 1140 non-dominated 1st quartile solutions in total while SPEA2 and NSGA-II only 885 and 715, respectively. Moreover, mPOEMS significantly outperforms the other two algorithms in hypervolume measure as well (confirmed by t -test as well as Wilcoxon rank-sum test, both at the 1% significance level). Note, the choice for the mPOEMS algorithm was made based on the fact that mPOEMS has been shown to outperform the state-of-the-art multiobjective evolutionary algorithms NSGA-II and SPEA2 on the multiobjective 0/1 knapsack problem [10], whose formal specification is very similar to the one of the project portfolio optimization problem. The following analyses justify the correctness of our choice as they show the mPOEMS outperforms the NSGA-II and SPEA2 on this problem as well. A distribution of 60 hypervolume values observed for each algorithm is shown in Fig. 2a and corresponding statistics are in Table 1. Boxplots generated from the hypervolume values are presented in Fig. 3a. They show that SPEA2 has the smallest (i.e. the best) median hypervolume value out of the three compared algorithms. Since the notches in the SPEA2 and NSGA-II boxplots and the notches in the SPEA2 and mPOEMS boxplots do not overlap, we can conclude, with 95% confidence, that the true SPEA2 and NSGA-II medians as well as SPEA2 and mPOEMS medians do differ. These differences were also approved by Wilcoxon rank-sum test at the 1% significance level while the two-sample t -test suggests that the null hypothesis that the means are equal cannot be rejected at the 1% significance level.

Thus, we may conclude that SPEA2 and NSGA-II achieve good hypervolume values largely thank to the extremal solutions that are very good in some objectives and really bad in the others at the same time. On the other hand, the mPOEMS is significantly better than SPEA2 and NSGA-II at generating the well-fit solutions³ with respect to both quality measures. We argue that this is an important achievement from the practical point of view. Given we do not know in advance what objective values can be considered already good enough for the problem at hand, we have to use approaches that seek for the whole Pareto-optimal set (such as SPEA2, NSGA-II or mPOEMS) and then extract from the final set of non-dominated solutions just those solutions that are not inferior in any objective. Our experiments show that mPOEMS is the best choice for this optimization strategy for the considered project selection problem as it attains the best set of non-dominated well-fit solutions.

8. Conclusions

In this paper we presented an effective approach to model the complex project portfolio selection environment in a formal model, with special focus on the adaptability to multiobjective optimization algorithms. The model comprises all characteristics found in the literature about project portfolio optimization. Selection of projects and optimization of portfolios was found very useful and the presented tools regarding project selection, project sequencing,

goal maximization respectively minimization, resource optimization, and balancing of preferences have proven to work efficiently. The quality of the presented optimization approach is especially based on the combined number of optimization tools a project portfolio manager can use with the proposed approach. Furthermore the presented approach is highly modular and scalable. Goals, restrictions, and preferences may be added or removed, and the number of projects to optimize is not restricted.

The proposed mPOEMS optimization approach proved to perform comparably to or even better than the state-of-the-art multiobjective optimization evolutionary algorithms. Particularly, mPOEMS is significantly better at generating good trade-off solutions than the compared algorithms.

Findings can be briefly described as follows:

- A new project selection method is proposed to enable the portfolio optimization to not only conduct optimization in the selection of projects, but also in the sequencing of projects, thus performing resource usage optimization.
- A method to deal with objective criteria like number of categories assigned to a specific category in a portfolio is proposed. This method considers such criteria as another objective to optimize and combine it with restrictions which can be relaxed.
- The evaluated optimization algorithms proved to be capable to deal with a complex set of goals and restriction.
- The mPOEMS approach exhibits better capabilities of finding sets of high-quality solutions than the compared state-of-the-art algorithms.

Various topics are subject to further work. The test set with which the approach was tested included 50 projects. It should be investigated how the performance of the optimization algorithms under consideration change for test sets with a greater number of projects, and more interdependencies to take care off. Regarding the test set it would be important to test the approach on a test set with real-world data. The relevance of the proposed project sequencing optimization and resource usage optimization for real-world businesses would be also an interesting topic for further research. The presented approach is unable to deal with incomplete data sets and uncertainty is only considered in the risk metric. Therefore it should be investigated how this optimization approach could deal with incomplete and uncertain data. Furthermore the second phase of the project portfolio decision support framework should be defined. The second phase comprise a decision support process for the selection of one portfolio out of the Pareto-optimal front, especially designed to be applicable to a group of decision makers. This group decision support has to use sophisticated visualization techniques and data exploration tools in order to provide insight into the multi-dimensional portfolio data, and support the group of decision makers in choosing the best alternative.

There are also several open issues regarding the mPOEMS that we plan to focus on in the future. Particularly, different prototype selection strategies and means for boosting the exploration capabilities of the algorithm will be studied first as they seems to have the greatest impact on the efficiency of the algorithm.

Acknowledgements

Research described in this paper has been partly supported by a Marie Curie Transfer of Knowledge Fellowship of the European Community's 6th Framework Program under the contract MTKD-CT-2005-029755: CzechVMXT. Jiří Kubalík has been supported by the research program No. MSM 6840770012 "Transdisciplinary Research in Biomedical Engineering II" of the CTU in Prague. Stefan Biffl has been supported by the Christian Doppler Forschungsgesellschaft and the BMWFJ, Austria.

³ By the well-fit solutions we mean the 1st quartile solutions.

References

- [1] N.P. Archer, F. Ghasemzadeh, An integrated framework for project portfolio selection, *International Journal of Project Management*. (1999) 207–216.
- [2] B. Boehm, *Software Cost Estimation with COCOMO II*, Prentice Hall PTR, Upper Saddle River, 2000, p. 17.
- [3] K. Deb, *Multi-objective Optimization Using Evolutionary Algorithms*, John Wiley & Sons, Ltd., New York, 2002.
- [4] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Transactions on Evolutionary Computation* 6 (2) (2002) 182–197.
- [5] K. Doerner, W.J. Gutjahr, R.F. Hartl, C. Strauss, C. Stummer, Pareto ant colony optimization: a metaheuristic approach to multiobjective portfolio selection, *Annals of Operations Research* (2004) 79–99.
- [6] F. Ghasemzadeh, N.P. Archer, Project portfolio selection through decision support, *Decision Support Systems* 29 (2000) 73–88.
- [7] S. Gueorguiev, M. Harman, G. Antoniol, Software project planning for robustness and completion time in the presence of uncertainty using multi objective search based software engineering, in: *Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation Conference, GECCO 2009*, Montreal, Quebec, Canada 131, 2009, pp. 1673–1680.
- [8] S. Iamratanakul, D.Z. Milosevic, Using strategic fit for portfolio management., in: *Proc. Portland International Center for Management of Engineering and Technology*, 2007, pp. 2089–2095.
- [9] J. Kubalik, J. Faigl, Iterative prototype optimisation with evolved improvement steps, in: P. Collet, M. Tomassini, M. Ebner, A. Ekart, S. Gustafson (Eds.), *Proceedings of the 9th European Conference on Genetic Programming, EuroGP 2006*, Heidelberg, Springer, 2006, pp. 154–165.
- [10] J. Kubalik, R. Mordinyi, S. Biffl, Multiobjective prototype optimization with evolved improvement steps, in: *Evolutionary Computation in Combinatorial Optimization*, Heidelberg, Springer, 2008, pp. 218–229, ISBN 978-3-540-78603-0.
- [11] J. Kubalik, Solving the sorting network problem using iterative optimization with evolved hypermutations, in: *Genetic and Evolutionary Computation Conference 2009 [CD-ROM]*, New York, ACM, 2009, pp. 301–308.
- [12] J. Kubalik, Solving the multiple sequence alignment problem using prototype optimization with evolved improvement steps, in: *Adaptive and Natural Computing Algorithms*, Heidelberg, Springer, 2009, pp. 183–192, ISBN 978-3-642-04920-0.
- [13] J. Kubalik, F. Železný, Solving the shortest common supersequence problem using iterative optimization with evolved hypermutations, in: R. Trapp (Ed.), *Proceedings of the Twentieth European Meeting on Cybernetics and Systems Research, EMCSR 2010*, Austrian Society for Cybernetic Studies, Vienna, 2010, pp. 579–584.
- [14] H.A. Levine, *A practical guide to selecting projects managing portfolios and maximizing benefits*, John Wiley & Sons, San Francisco, 2005.
- [15] A.L. Medaglia, S.B. Graves, J.L. Ringuest, A multiobjective evolutionary approach for linearly constrained project selection under uncertainty, *European Journal of Operational Research* 179 (3) (2007) 869–894.
- [16] *Project Management Institute: The Standard for Portfolio Management*, Pennsylvania, 2006.
- [17] B. De Reyck, Y. Grushka-Cockayne, M. Lockett, S.R. Calderini, M. Moura, A. Sloper, The impact of project portfolio management on information technology projects, *Proc. International Journal of Project Management* 23 (2005) 524–537.
- [18] C. Stummer, R. Vetschera, Decentralized planning for multiobjective resource allocation and project selection, *Central European Journal of Operations Research* 11 (2003) 253–279.
- [19] C. Stummer, K. Heidenberger, Interactive R & D portfolio analysis with project interdependencies and time profiles of multiple objectives, *IEEE Transactions on Engineering Management* 50 (2003) 181–184.
- [20] E. Zitzler, L. Thiele, Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach, *IEEE Transactions on Evolutionary Computation* 3 (4) (1999) 257–271.
- [21] E. Zitzler, M. Laumanns, L. Thiele, SPEA2: improving the strength Pareto evolutionary algorithm for multiobjective optimization, in: *Evolutionary Methods for Design, Optimisation, and Control*, Barcelona, Spain, 2002, pp. 19–26.
- [22] E. Zitzler, M. Laumanns, L. Thiele, S. Bleuler, PISA—A Platform and Programming Language Independent Interface for Search Algorithms, *Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science, Berlin, 2003, pp. 494–508.