International Journal of Production Economics, Volume 109, Issues 1-2,
Special Section on Cost Engineering, September 2007, Pages 105-121.

# Optimisation of Business Process Designs: An algorithmic approach with multiple objectives

K. Vergidis[1*], A. Tiwari[1], B. Majeed[2] and R. Roy[1]

[1]*Manufacturing Department, School of Applied Sciences, Cranfield University, Cranfield, MK43 0AL, U.K.*

[2]*Computational Intelligence Group, IS Lab, BT Research and Venturing, Orion Building, Adastral Park, Martlesham, IP5 3RE, U.K.*

## Abstract

Most of the current attempts for business process optimisation are manual without involving any formal automated methodology. This paper proposes a framework for multi-objective optimisation of business process designs. The framework uses a generic business process model that is formally defined and specifies process cost and duration as objective functions. The business process model is programmed and incorporated into a software platform where a selection of multi-objective optimisation algorithms is applied to a range of test designs including a real example. The test business process designs are of varying complexity and are optimised with three popular optimisation techniques (NSGA2, SPEA2 and MOPSO algorithms). The results indicate that although business process optimisation is a highly constrained problem with fragmented search space; multi-objective optimisation algorithms such as NSGA2 and SPEA2 produce a satisfactory number of alternative optimised business process designs. However, the performance of the optimisation algorithms drops sharply as the complexity of the process designs increases. This paper also discusses the directions for future research in this particular area.

**Keywords:** Business process (bp), bp optimisation, bp re-design, bp modelling and analysis.

---

[*] Corresponding author. Phone: +44 (0) 1234 754194. Fax: +44 (0) 1234 750852. E-mail address: k.vergidis@cranfield.ac.uk

## 1. Introduction

In the modern competitive business world there is a frequent need for enterprises to modify their design of business processes to become more successful in the marketplace. The design and management of business processes is a key factor for companies to effectively compete in today's volatile business environment. By focusing on the optimisation and continuous improvement of business processes, organisations can establish a solid competitive advantage by reducing cost, improving quality and efficiency, and enabling adaptation to changing requirements. Multi-objective optimisation of business processes can result in novel approaches and more efficient ways of business process improvement since more than one optimisation criteria can be selected and satisfied concurrently. The next section examines the relevant work in the specific subject area while the rest of the paper introduces a multi-objective optimisation framework for formally defined business process designs.

## 2. Related work

Process modelling methodologies, such as the IDEF family, Computer Integrated Manufacturing – Open Systems Architecture (CIM-OSA), Object-oriented Modelling and Petri-nets, allow for a systematic and a well-defined representation of processes (see Aguilar-Saven, 2004). Based on some of the above methodologies, a number of process modelling tools have been developed, such as ARIS, FirstStep, PrimeObjects and TEMAS (Zakarian, 2001). These approaches provide powerful methods for visualising business processes, evaluating their particular characteristics (such as resource utilisation, cost and speed) and checking their structural and resource consistency (Sadiq and Orlowska, 2000; Li *et al.,* 2004). Zakarian (2001) also integrated the Fuzzy-rule-based Reasoning Approach with IDEF methodology for quantitative analysis of process models to model efficiently the uncertain and incomplete information of process variables that exist in most of the traditional modelling techniques. Vaughan (2006) presents a linear model that with its robustness can contribute to business process modelling. Grigori *et al.* (2004) proposed a Business Process Intelligence tool suite that uses Business Intelligence Technologies (in particular data mining) for analysing business processes. Also Cheung and Bal (1998) presented an overview of business process analysis techniques and tools and Byrne and Hossain (2005) an improved hybrid approach for production planning.

The qualitative nature of business processes explains the difficulty in developing their parametric models. Therefore, although a considerable number of algorithms exist for dealing with process optimisation problems in areas such as Logistics (Yu and Li, 2000), there is a lack of algorithmic approaches for the optimisation of business processes. McKay and Radnor (1998) attempt to apply product data engineering principles and methods to the representation of business processes but there is no formal optimisation attempt. Much of the recent research in the area of business process optimisation has dealt with either selection of a process model from a set of alternatives (Shimizu and Sahara, 2000) or simple single-objective optimisation (Hofacker and Vetschera, 2001) that does not address the strong synergistic/anti-synergistic effects among individual activities that constitute a process design. Therefore, the current research suffers from serious limitations in dealing with the scalability requirements and complexity of real-life processes. Hofacker and Vetschera (2001) attribute this to the lack of formal methods to support the design of business processes. One of the main reasons for this is that the design elements and constraints on process designs are hard to characterise in a formal way amenable to analytical methods. These authors propose a business process modelling approach that can be optimised with three different techniques. They describe a business process using a single-objective mathematical model that can be minimised or maximised according to each optimisation concept.

In summary, the optimisation attempts for business processes have still a long way to go due to three main problems:

1. There is no widely recognised and adopted business process representation in terms of quantitative modelling. Most business process modelling techniques (such as flowcharts and IDEF) use visual diagrammatic approaches not capable of quantitative analysis and structured optimisation. These diagrammatic approaches use standard graphical notations for visual representation and analysis of business processes.
2. The business process optimisation attempts have been mostly manual and based on simplistic or isolated cases without generalisation capabilities.
3. There are no attempts to optimise a business process under multiple criteria. The lack of multi-objectivity makes any algorithmic approach less attractive without tangible benefits.

## 3. Multi-objective optimisation of business processes

This section introduces the basic steps of the proposed multi-objective optimisation framework. The framework's main aim is to introduce a methodology for applying multi-objective optimisation algorithms to business process designs. The first stage of the framework is the business process model specification. The model is formulated on a mathematical basis to ensure formality, consistency and rigour. The second stage of the framework involves the application of the optimisation algorithms to the business process model.

### 3.1 Model construction

As mentioned above, the first stage of the framework is concerned with the business process model construction and the necessity for formal representation of the business process. The business process model is limited by a series of mathematical constraints that define the feasibility boundaries of the business process and a set of objective functions that consist of the various business process objectives. To follow a formally correct, repeatable and verifiable approach using a mathematical model guarantees the construction of consistent and rigorous business process models (Koubarakis and Plexousakis, 2002).

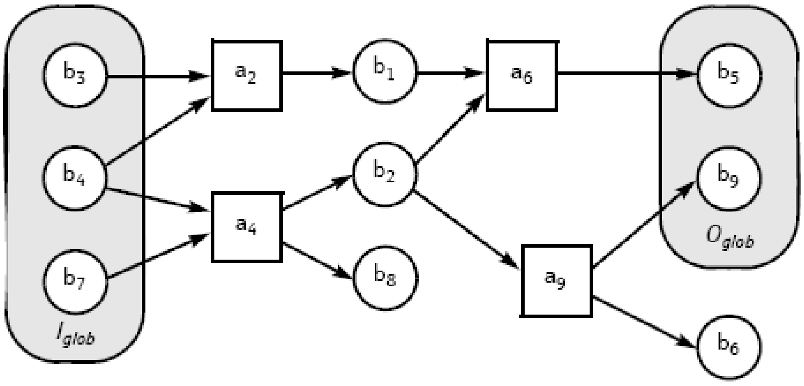



Figure 1. A feasible business process design using activities and resources
(source: Hofacker & Vetschera (2001))

Figure 1 sketches a feasible business process design using two key concepts: activities and resources. This design is in line with Bititci and Muir (1997) suggestion that business activities that together fulfil a goal form a business process. The business process design of figure 1 has two sets of resources, the initial ($I_{glob}$) and the final ($O_{glob}$) resources. The initial resources are available at the beginning of the business process while the final resources form the final output. The resources flow through the process and belong to two main categories: physical and information resources. The activities are perceived as the transformation steps within the process that make use of some resources as inputs and produce others as outputs. In a feasible process design all the activities are in a defined sequence, the resources are adequate and most importantly the final resources are produced by the participating activities.

The business process model is optimised by defining the model's optimisation variables and objectives as demonstrated in figure 2. The optimisation variables of the business process model are the participating activities and their starting times. The aim is to produce an improved process by minimising the two objectives, the process duration and cost. For each process design, there is a library of candidate activities with attributes such as activity duration and activity cost. These activities are defined in terms of their input and output resources and their duration and cost attributes. Some activities that have similar requirements in terms of their input and output resources are interchangeable in a business process design. Changing activities in a process design affects directly the total business process cost and duration. For an optimised business process design to be produced a set of activities that generate minimum business process cost and duration needs to be selected.

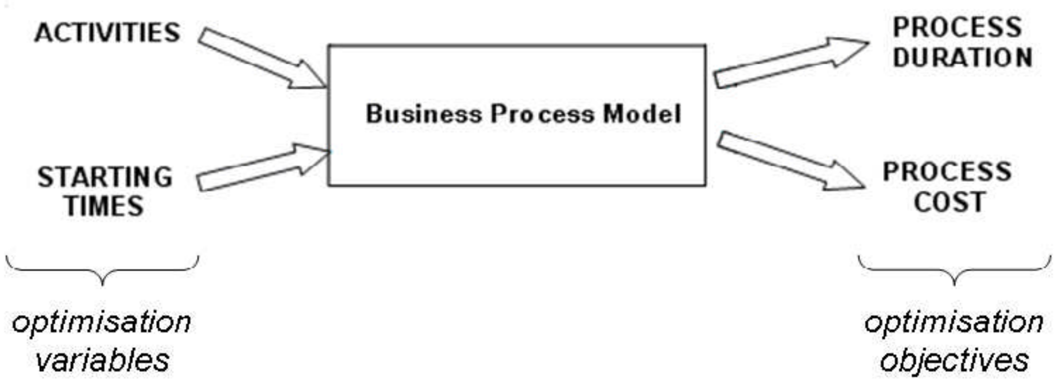



Figure 2. Optimisation variables and objectives of the business process model

It is important to note that the framework works independently of the number of process objectives. Process duration and cost were chosen as two common objectives for business process improvement. The business process model of figure 2 is formally defined with a mathematical model. The mathematical model defines the optimisation objectives with two objective functions and ensures the business process consistency and feasibility with thirteen constraints. It is possible to add further process objectives by formulating more objective functions. The complete mathematical model is the following:

$$f_1(P) = \max(q_j) \to \min, \forall j : b_j \in go_j$$
$$f_2(P) = \sum u_{i1} x_i \to \min$$
$s.t.$
$$1. x_i \le r_{ij}, \forall i, j : b_j \in I_i, b_j \in B_P,$$
$$2. x_i \le y_j, \forall i, j : b_j \in I_i, b_j \in B_I,$$
$$3. go_i + \sum_i r_{ij} \le M \cdot gi_j + \sum_i t_{ij} x_i, \forall j : b_j \in B_P,$$
$$4. y_j \le gi_j + \sum_i t_{ij} x_i, \forall j : b_j \in B_I,$$
$$5. y_j \ge go_j,$$
$$6. p_i \ge q_j - M(1 - x_i), \forall i, j : b_j \in I_i,$$
$$7. q_j \le p_i + \delta_i + M(1 - x_i), \forall i : b_j \in O_i$$
$$8. q_j \ge p_i + \delta_i - M(1 - x_i) - M(1 - \lambda_{ij}), \forall i : b_j \in O_i,$$
$$9. \lambda_{ij} \le x_i, \forall i, j : b_j \in O_i,$$
$$10. \sum_{i : b_j \in O_i} \lambda_{ij} \ge \sum_i r_{ij} + og_i - M(1 - y_j), \forall j : B_P, gi_j = 0,$$
$$11. \sum_{i : b_j \in O_i} \lambda_{ij} \ge 1 - M(1 - y_j), \forall j : b_j \in B_P, gi_j = 0,$$
$$12. x_i \in \{0,1\}, \forall i,$$
$$13. \lambda_{ij} \in \{0,1\}, \forall i, j : b_j \in O_i.$$

where:

$u_{il}$ = cost of execution for activity $a_i$

$x_i$ = binary variable that indicates whether a candidate activity $a_i$ participates in the business process design

$y_j$ = binary variable that indicates whether resource $b_j$ is or becomes available during the business process

$t_{i,j}$ = matrix of binary variables that links the activities with their output resources

$r_{ij}$ = matrix of binary variables that indicate if a unit of physical resource $b_j$ is available for use by activity $a_i$

$g_{ij}$ and $g_{oj}$ = one-dimensional binary constants that indicate which resources belong to global inputs and/or global outputs

$M$ = large constant indicating that physical resources contained in the set of global inputs are available in unlimited amounts

$p_i$ = starting time of activity $a_i$

$q_j$ = the time resource $b_j$ becomes available

$\delta_i$ = duration of activity $a_i$

$\lambda_{ij}$ = binary variable indicating that activity $a_i$ is used to create resource $b_j$
$I_i$ / $O_i$ = sets of input/output resources of activity $a_i$
$B_P$/$B_I$ = set of physical/information resources $b_j$.

The mathematical expression of process model appears to be complicated in contrast with its visual representation (figure 1) where it consisted of activities and resources. The mathematical model consists of a number of binary variables and binary matrices that have a serious impact on the production of feasible process designs since they result in a highly fragmented search space. The first objective function ($f_1$) of the model calculates the duration of the business process. The total duration for a feasible process equals the time the last resource that belongs to global outputs is produced. The second objective function ($f_2$) calculates the business process cost as the sum of costs of all participating activities. The mathematical model constraints ensure that the model produces feasible business processes by examining different aspects of the business process model.

Table 1 provides a short description of each constraint of the mathematical model. It is important to highlight two features of the business process model. The mathematical model consists of many discrete binary variables that significantly increase the complexity of even a simple process design as the search space for feasible solutions is highly fragmented. Another feature of the business process model is that although it is simple to conceive, understand and visualise, it proves to be complex and highly constrained when it comes to formal mathematical definition. This can create serious difficulties in locating the optimum solutions among the feasible ones since even feasible solutions are hard to be produced. The concepts that describe the business process and its mathematical model are inspired by Hofacker and Vetschera (2001). The aim of the paper is to extend their model to multi-objectivity and optimise it using evolutionary algorithms.

| | |
|---|---|
| $x_i \le r_{ij}, \forall i, j : b_j \in I_i, b_j \in B_P$ …..(1)<br>All input physical resources of an activity must be available ($r_{ij}$=1) at some stage of the process if the activity is participating ($x_i$=1). | |
| $x_i \le y_j, \forall i, j : b_j \in I_i, b_j \in B_I$ …..(2)<br>All input information resources ($y_j$) of an activity must be available at some stage of the process if the activity is participating ($x_i$=1). | |
| $go_i + \sum_i r_{ij} \le M \cdot gi_j + \sum_i t_{ij} x_i, \forall j : b_j \in B_P$ ….(3)<br>The output physical resources -final or not- must not exceed the sum of initial and produced -during the process. | |
| $y_j \le gi_j + \sum_i t_{ij} x_i, \forall j : b_j \in B_I$ …..(4)<br>An information resource ($y_j$) can be available either at the beginning of the process -as initial resource ($gi_j$) - or as an output resource of a participating activity. | |
| $y_j \ge go_j$ …..(5)<br>A resource ($y_j$) cannot be part of the output without first being available at some stage of the process ($go_j$). | |
| $p_i \ge q_j - M(1 - x_i), \forall i, j : b_j \in I_i$ …..(6)<br>In terms of time, a participating activity must start ($p_i$) only after the time that all its input resources have become available. | |
| $q_j \le p_i + \delta_i + M(1 - x_i), \forall i : b_j \in O_i$ …..(7)<br>$q_j \ge p_i + \delta_i - M(1 - x_i) - M(1 - \lambda_{ij}), \forall i : b_j \in O_i$ …..(8)<br>In terms of time, an output resource must become available exactly when the generating activity has been completed ($q_j$=$p_i$). | |
| $\lambda_{ij} \le x_i, \forall i, j : b_j \in O_i$ …..(9)<br>A non-participating activity ($x_i$=0) cannot have output resources ($\lambda_{ij}$=1). | |
| $\sum_{i:b_j \in O_i} \lambda_{ij} \ge \sum_i r_{ij} + og_i - M(1 - y_j), \forall j : B_P, gi_j = 0,$ …..(10)<br>When a physical resource does not belong to initial resources, it must be produced during the process in greater or equal amounts to the required resource inputs of the participating activities. | |
| $\sum_{i:b_j \in O_i} \lambda_{ij} \ge 1 - M(1 - y_j), \forall j : b_j \in B_P, gi_j = 0$ …..(11)<br>Each physical resource that does not belong to initial resources but appears in the output of a participating activity must be produced at least once. | |
| $x_i \in \{0,1\}, \forall i$ …..(12)<br>The variable x (indicating participating activities) must be binary. | |
| $\lambda_{ij} \in \{0,1\}, \forall i, j : b_j \in O_i$ …..(13)<br>The variable λ (indicating output resource *j* of activity *i*) must be binary. | |

Table 1. Description of the mathematical model constraints

### 3.2 Optimisation algorithms

The second stage of the framework deals with the identification and application of multi-objective optimisation algorithms to the business process model. Although there are many optimisation techniques in literature such as nonlinear programming, tabu search and simulated annealing to name a few, the authors decided to focus exclusively on evolutionary optimisation techniques in an attempt to investigate their suitability on the business process domain. Evolutionary algorithms have proved to be efficient in dealing with multi-objective and highly constrained problems in other relevant areas such as scheduling, supply chain optimisation etc.

A number of algorithms exist in literature that can deal with multi-objective optimisation problems expressed in the form of mathematical models. These algorithms can deal with any criterion expressed using a mathematical model. After relevant research, three algorithms that allow multi-objective optimisation have been selected to optimise the business process model. This selection was based on the requirement of dealing with fragmented search space and constraints. The selected algorithms are Non-Dominated Sorting Genetic Algorithm II (NSGA2), Strength Pareto Evolutionary Algorithm II (SPEA2) and Multi-Objective Particle Swarm Optimisation (MOPSO). According to Deb (2001), NSGA2 and SPEA2 are popular due to their robustness in solving fragmented search space problems. MOPSO–a multi-objective extension of the Particle Swarm Algorithm- is a new algorithm that to the authors' knowledge has never been used on such a constrained problem (Kennedy and Eberhart, 1999). This guided the selection of NSGA2, SPEA2 and MOPSO for this research. Since more than one optimisation method is applied to the business process model, the opportunity of comparing the performance of the different algorithms in the particular problem context is created. NSGA2 and SPEA2 are demonstrated in a number of papers and although similar, they compete each other in the quality of results produced in different subject domains. In more detail:

1. *Non-dominated Sorting Genetic Algorithm II.* NSGA2 is a non-dominated, sorting-based, multi-objective evolutionary algorithm. The first version of NSGA (Deb, 2000) received criticism because like other genetic algorithms that use non-dominated shorting and sharing, has high computational complexity. NSGA2 overcomes many of its predecessor's problems by having less complexity as it uses fast non-dominated sorting approach (Deb, 2001). Its elitist approach and parameter-less sharing approach can be also taken as significant improvements. NSGA2 has been popular and has

been applied to many problems on a number of research areas. A recent example involve Chafekar *et al.* (2003) who explores the potential of constrained multi-objective optimisation.

2. *Strength Pareto Evolutionary Algorithm II.* In 1999, Zitzler and Thiele proposed SPEA as another elitist evolutionary algorithm. The improved version, namely SPEA2, evolved from the same group of authors (Zitzler *et al.* 2001) and incorporates in contrast to its predecessor a fine-grained fitness assignment strategy, a density estimation technique, and an enhanced archive truncation method. SPEA2 has also been popular and used in a variety of optimisation problems.

3. *Multi-Objective Particle Swarm Optimisation (MOPSO).* Different from most evolutionary computation techniques, Particle Swarm Optimisation (PSO) method is motivated from the simulation of social behaviour of bird flocking and fish schooling. PSO was originally designed and developed by Kennedy and Eberhart (1995). It shares many similarities with evolutionary computation techniques. The particles fly through the problem space by following an optimum particle called guide. Changing PSO to optimise a multi-objective problem requires a redefinition of what a guide is in order to obtain a front of optimal solutions. In Multi-Objective Particle Swarm Optimization (MOPSO), the Pareto-optimal solutions should be used to determine the guide for each particle. Kennedy and Eberhart (1999) point out that MOPSO has demonstrated good performance in problems that have continuous search space.

## 4. Test business process designs

This section describes the construction of five test business process designs. These five test designs are defined and optimised based a generic business process design that is described below. Only one of these test designs is presented in this section and related to a real example of business process. The descriptions and results of the remaining four can be found in the Appendix.

### 4.1 Generic business process design

The generic process design demonstrated in figure 3 has a set of predefined input ($I_{glob}$) and output ($O_{glob}$) resources ($b_j$). Also, the number of activities ($a_i$) that can participate in the business process is fixed. The activities that can potentially participate in the business process are selected from a library with alternatives. The size of the library varies for each particular test design.

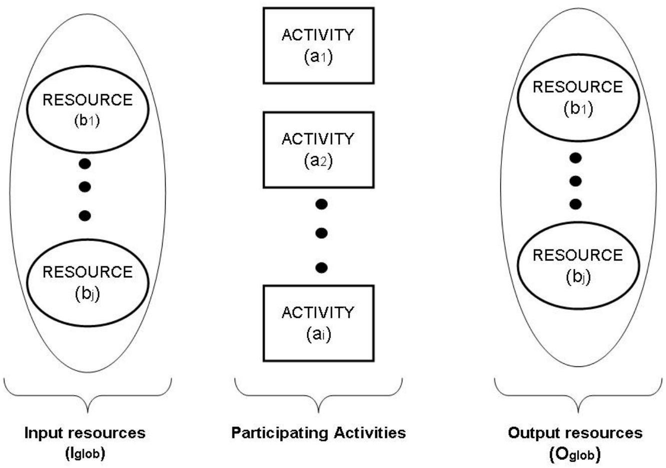



Figure 3. Structure of the generic business process design

Business process design optimisation involves the generation of alternative process designs with optimised objective values (process duration and cost). These improved designs contain different combinations of the library activities with their starting times appropriately set to result in reduced objective values for the business process.

The aim of creating the test designs from the generic design is to assess the performance of the optimisation algorithms in optimising these designs. The test designs that are utilised have an increasing number of participating activities, variations on the number of input and output resources, and also different library sizes. The structure of one of the test designs is described below and discussed in parallel with a real example. The structure of the other test designs is presented in table 4 in the Appendix.

## 4.2 A test design and a real example

The name of the test design discussed is ActivitiesST4. This test design involves 4 participating activities and a library of 10 alternatives. It also has two global input resources to start the process. These two resources together with the two global outputs are predefined. The combination of the 4 activities selected to participate in the process design must allow the process output resources to be produced. The optimisation variables of the problem are eight, the 4 activities to be selected from the library and the 4 four starting times of these activities. In case that two or more activities are linked sequentially, their corresponding starting times need to be properly adjusted by the algorithm to reflect this sequence. As can be seen from figure 2, both starting times and activities are optimisation variables having a significant impact on the design complexity.

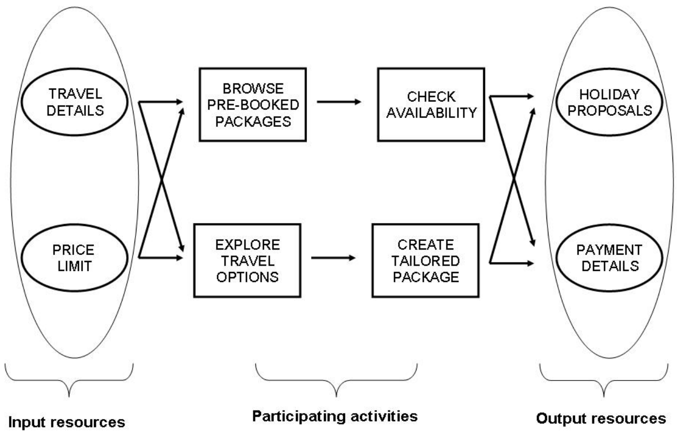



Figure 4. Travel agent process of offering holidays

A real example of this process is discussed below. The ActivitiesST4 process design can be perceived as a travel agent's process of offering holidays to a customer. The process starts with the customer providing the price that he/she is willing to pay and also the details of his/her desired holiday, e.g. travel destination, dates and trip duration. The travel agent then takes two simultaneous actions: he/she searches and checks the availability of existing holiday packages that meet the customer requirements and at the same time he/she explores individual travel options in order to create a tailored package that suits the specific customer needs. The result of these actions is to provide different holiday proposals to the customer along with the payment details for each one of them. The travel agent process of offering holidays is demonstrated in figure 4.

| Object name | Process element | Alternatives | Cost | Duration |
|---|---|---|---|---|
| Travel details | *Input resource* | - | - | - |
| Price limit | *Input resource* | - | - | - |
| Browse pre-booked packages | *Activity* | 1. Search from brochures | 2 | 9 |
| | | 2. Search company intranet | 7 | 5 |
| Explore travel options | *Activity* | 1. Browse past cases | 4 | 8 |
| | | 2. Explore new options | 6 | 6 |
| Check availability | *Activity* | 1. Via intranet/e-mail | 10 | 1 |
| | | 2. Via phone/post | 5 | 7 |
| Create tailored package | *Activity* | 1. Use specific software | 11 | 2 |
| | | 2. Combine options manually | 5 | 6 |
| Holiday proposals | *Output resource* | - | - | - |
| Payment details | *Output resource* | - | - | - |

Table 2. Explanation of the travel agent process elements

Table 2 provides an explanation of the process elements. 'Travel details' and 'price limit' are the two input resources that are necessary to initiate the process of offering holidays. Then for each activity of figure 4 there are two alternatives with different cost and duration values. In this example the alternatives for each activity are the different ways that the activity can be executed. Each activity in the process design uses input resource(s) and transforms them so that the following activities can utilise them until the final output resources have been produced. In this example, the customer's preferences and details are fed to the process and transformed by the activities to the actual holiday proposal and payment details. For each alternative activity the values (provided in arbitrary units) reflect the trade-off between cost and duration, i.e. manual activities tend to cost less but last longer while automated activities carry a higher cost but have much shorter duration. These cost and duration values are the average for each particular activity. The output resources of the process are considered as the 'Holiday proposals' and the 'Payment details'. There are two input and output resources in the example, the participating activities are four and the library contains 8 alternative activities thus making this real example similar to ActivitiesST4 test process design. The alternative activities named in this particular example are considered as mutually exclusive, however for each activity another alternative can be potentially added that utilises both alternatives together. It is expected though that such alternative will have greater values both in cost and duration. This alternative has not been included here. However, the framework is generic and capable of handling additional alternatives.

## 5. Experimental results

This section describes the experimental results for the process design discussed in the previous section and the remaining four that are presented in the Appendix. The aim of this section is to evaluate both the algorithms' optimisation performance in this constrained problem and to demonstrate the alternative optimised processes that are generated for the travel agent process example.

### 5.1 Performance of optimisation algorithms

The test problems are incorporated in KEA toolbox (Bartz-Beielstein, 2004), a software optimisation platform that utilises NSGA2, SPEA2 and MOPSO algorithms to optimise user-defined problems. Each of the five test designs were imported in the platform and optimised with each of the three evolutionary algorithms. Table 3 presents the parameters for each of the algorithms as these were defined by the authors. These parameters are typically used in literature for comparing the algorithms (see Freschi and Repetto, 2005). These are also the default values in the KEA toolbox (Bartz-Beielstein, 2004).

| NSGA2 | SPEA2 | MOPSO |
|---|---|---|
| Population size = 100<br>Generations = 250<br>Mutation prob. = 0.2<br>Crossover probability = 0.8 | Population size = 100<br>Generations = 1000<br>Mutation prob. = 0.2<br>Archive size = 100<br>Recombination prob. = 0.8 | Particles = 20<br>Hypercubes = 10000<br>Generations = 1000<br>Repository = 100<br>Inertia = 0.4<br>Personal best = 1.0<br>Repository weight = 1.2 |

Table 3. Main parameters for the optimisation algorithms

In order to evaluate the results produced by optimisation algorithms and acquire a picture of the search space, a random population of 10,000 feasible solutions was initially generated. Random solutions are those, for which the variable values are randomly generated, i.e. in the real example random activities are selected from the library and their starting times are also randomly selected. Feasible solutions are those that satisfy all the constraints of the mathematical model and produce the actual process outputs; for the real example these are the holiday proposal and payment details. In order to produce the actual results (i.e. feasible optimised processes) each of the optimisation algorithms was executed 30 times with different random seed values. Most of these 30 runs produced similar results. The results presented here belong to one of the typical runs. The graphs in figure 5 demonstrate the solutions that each of the three optimisation techniques generated for the ActivitiesST4 process design.

The solutions for the remaining four test designs are in the Appendix (figures 8 and 9). The generated solutions consist of feasible business processes with minimised process duration and cost. The graphs depict the process duration and cost values for both the random and the optimised population. The dotted points represent the solutions that the techniques produced while the ‘x points’ the randomly generated solutions. To evaluate the results a performance metric is introduced. In general, the performance metrics for multi-objective optimisation algorithms focus on two main aspects:

1. The closeness of solutions to the Pareto-optimal front and
2. The diversity of solutions.

Deb (2001) introduced a series of evaluation metrics such as error ratio, set coverage metric, maximum Pareto-optimal front error and hypervolume. For the business process designs the primary aim is to measure the number of optimal solutions that are generated. For this reason we used a variation of the error ratio. The error ratio is an illustrative metric that is easily understood by business end users. As opposed to this, most other metrics used in multi-objective optimisation literature are designed for computer scientists. The error ratio measures the percentage of solutions that do not belong to the

Pareto-optimal front against the total number of produced solutions. To demonstrate better the quality of the experimental results, we designed the opposite of error ratio.

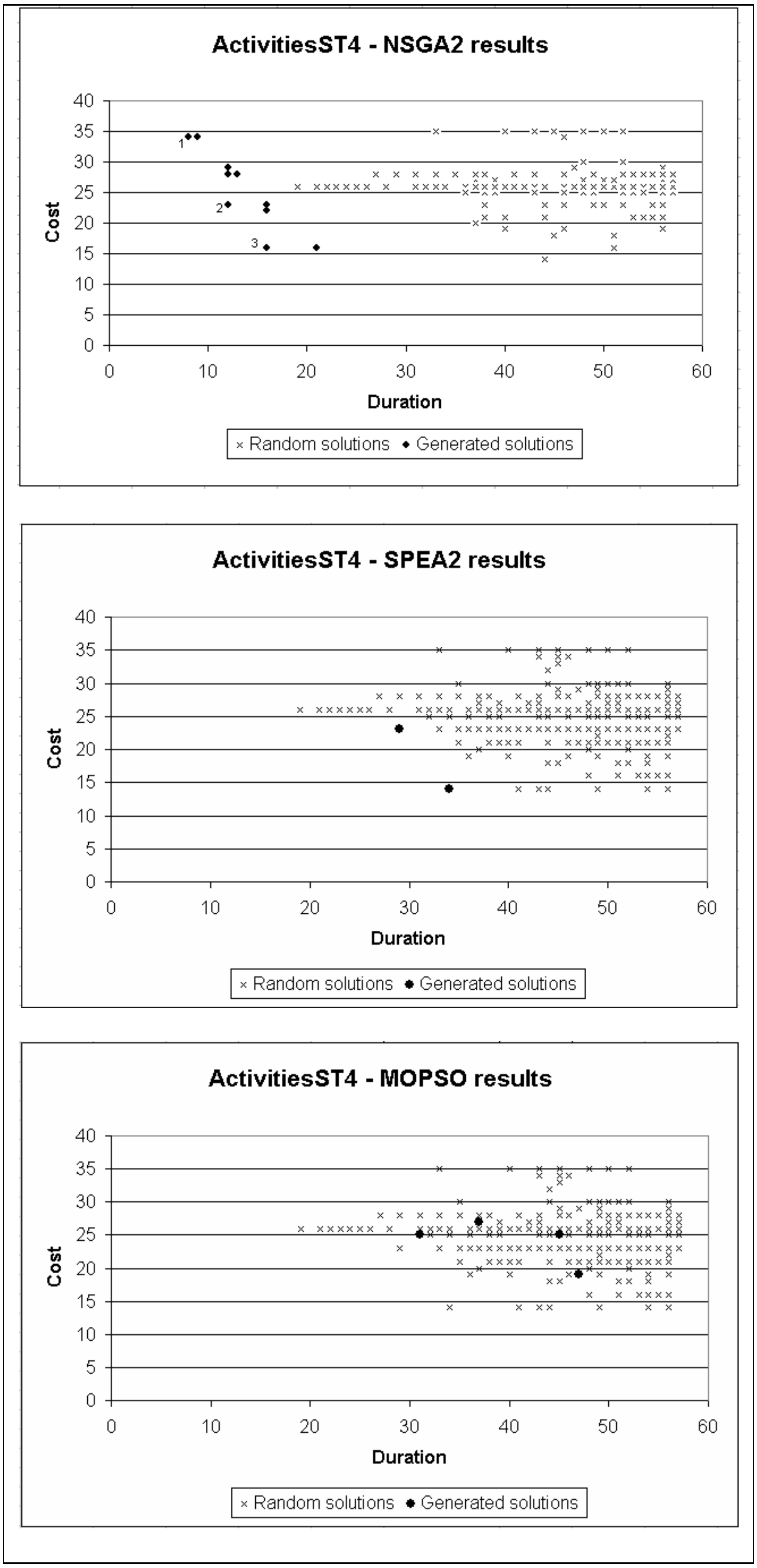
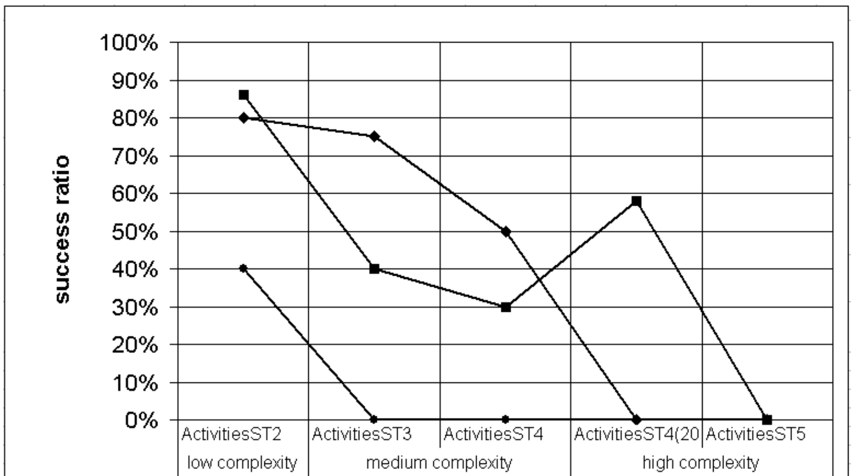



Figure 5. Generated solutions for ActivitiesST4 by the optimisation techniques

The ‘success ratio’ is calculated as the percentage of generated solutions that belong to the Pareto front against the total number of solutions.

The formula of the success ratio is: $$s_R = \frac{no\_of\_solutions \in P^*}{total\_solutions}\%$$

The numerator of the success ratio equation holds the number of generated solutions that belong to P* (Pareto-optimal front) while the denominator holds the total number of generated solutions. The success ratio ($s_R$) calculates the percentage of Pareto-optimal solutions that the optimisation algorithm has generated. Being a real-life situation the Pareto-optimal front is not known for the test problem. Therefore in this case, Pareto-optimality of a solution is defined if it is non-dominated with respect to the large set of randomly generated solutions as shown in figure 5.




| | ActivitiesST2 | ActivitiesST3 | ActivitiesST4 | ActivitiesST4(20 | ActivitiesST5 |
|---|---|---|---|---|---|
| | low complexity | medium complexity | | high complexity | |
| NSGA2 | 86% | 40% | 30% | 58% | 0% |
| SPEA2 | 80% | 75% | 50% | 0% | 0% |
| MOPSO | 40% | 0% | 0% | 0% | 0% |

Figure 6. Performance evaluation of the test process designs based on success ratio

Figure 6 demonstrates the performance of the five test designs based on the success ratio evaluation. For ActivitiesST2 process design, both NSGA2 and SPEA2 perform very well unlike MOPSO that identifies only 40% of the Pareto-optimal solutions. SPEA2 also produces very good results for ActivitiesST3 problem (75%) while NSGA2 gives a satisfactory number of optimum solutions (40%).

Nevertheless their performance drops significantly with the addition of an extra activity in ActivitiesST4. MOPSO performs poorly as apart from the first test problem it does not seem to be able to locate optimum solutions for the rest. Moving to test problems with bigger activity libraries, NSGA2 produced satisfactory results for ActivitiesST4(20), while for ActivitiesST5 problem none of the algorithms was able to locate solutions on the Pareto front due to its increased complexity. The average success ratio for both NSGA2 and SPEA2 is approximately 40%, while for MOPSO it is only 8%.

Before these results are further discussed, the features of the search space need to be highlighted once more as they have a significant impact on the quality of the results. The mathematical model of the business process design consists of discrete binary variables that increase the complexity of optimising even a simple process design as the search space is highly fragmented. Also the business process models are highly constrained having 13 constraints to check for every possible solution, decreasing the performance of the algorithms. The optimisation algorithms have a difficult task even to produce sets of feasible solutions. Given the complex nature of the business process designs, the overall performance of NSGA2 can be characterised as good and can be attributed to its elitism. As NSGA2 archives the optimum solutions of each generation and compares them with the ones it produces, it manages to preserve the identified feasible solutions. SPEA2 is also an elitist algorithm that provides bigger spread of the solutions. It also preserves feasible individuals through generation evolution and this justifies its satisfactory results. As opposed to this, MOPSO seems to have a serious problem in successfully defining the guide that combines the two objectives. The algorithm demonstrates poor performance as the solutions that it generates do not belong to the Pareto optimal front for most of the test problems. This supports the claim by Kennedy and Eberhart (1999) that MOPSO has better performance in problems with continuous search space which is not the case here.

Figure 6 also shows that as the complexity of the problems increases, the performance of the optimisation algorithms declines significantly. The simplest of test problems (ActivitiesST2) is handled well by all three algorithms. Moving to medium complexity problems, SPEA2 provides better results while NSGA2 hits back on high complexity problems with 58% success on one of the problems (ActivitiesST4(20)). On average performance NSGA2 holds the best position with slightly better results that SPEA2 which has also performed above 40% on average. This supports the claim by Zitzler *et al.* (2001) that SPEA2 and NSGA2 tend to behave very similar on various problems. Many applications of the NSGA2, SPEA2 and MOPSO are not as successful in dealing with large dimensional problems and extremely disconnected Pareto fronts.

### 5.2 Evaluation of alternative business process designs

After evaluating the ability of the three optimisation algorithms in producing optimised process designs, it is of equal importance to examine the practical implications of the results on producing alternative business process designs. We selected three NSGA2 solutions (the numbered dots in figure 5) in order to visually demonstrate the optimised business processes that were generated based on the travel agent business process example that was presented in section 4. Figure 7 sketches the three optimised business processes. These processes have different trade-offs between process duration and cost, thus none of them is better than the others.

At the beginning (time = 0) the two input resources are available and the process starts. The grey boxes represent the activities and their length depicts their duration. The process cost is calculated by adding all the activity costs, while the process duration is defined by the time that the last activity finishes its execution. For the duration and cost values of each activity refer to table 2. Figure 7 proves that the same business process can have different performances in terms of time and cost -by selecting different sets from amongst the library activities.

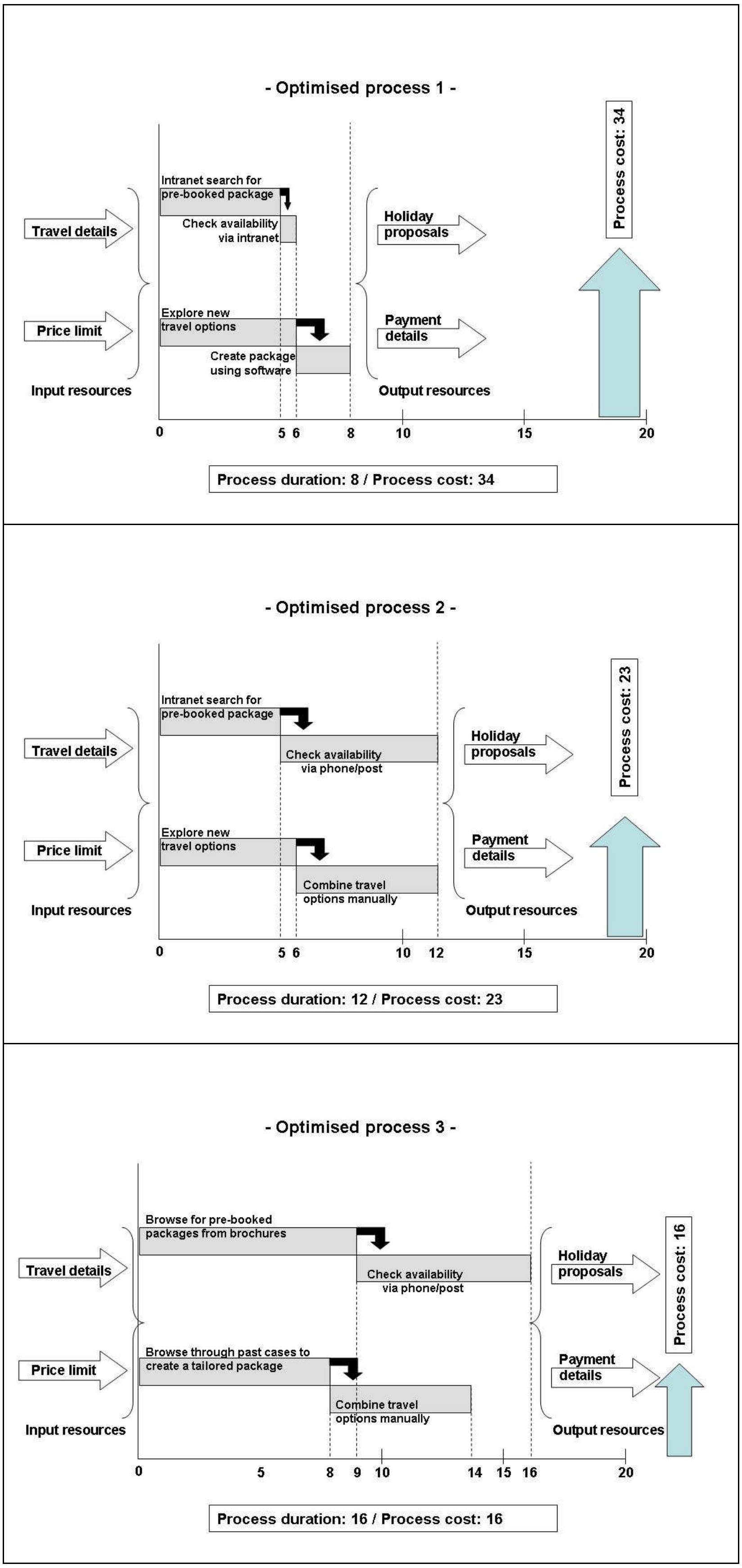



Figure 7. Optimised alternatives for the ‘travel agent’ business process

Optimised process 1 for example has the shortest duration (8 units) but it is also the most expensive as it is using mostly technological means to produce the resources. Optimised process 2 has reduced cost but has a 50% increase in process duration due to different combinations of the activities. Finally, optimised process 3 costs less than half of the first instance but it lasts twice as long. Therefore, the optimised solutions provide a range of selection to the process analyst to make a decision. The decision making criteria could be the company's priorities or policy at a given time or external factors such as competitor's performance. The risk of selecting a particular alternative over the others needs also to be taken into account. Having the opportunity to shape a business process according to two or more objectives and being able to review the trade-offs between these objectives empowers the process analyst when it comes to business process selection and realisation as there is a range of alternatives in his hand.

## 6. Discussion

This section discusses the practical implications of the framework, along with its limitations and directions for future research. The test problems demonstrated that the proposed framework is capable of applying multi-objective optimisation to various business process designs and generating Pareto-front solutions. The ability to produce an overall 40% of optimum solutions provides a good set of optimised alternative business processes with different trade-offs in process duration and cost. This gives the capability to the process owner to select according to decision making priorities a business process from a range of optimised ones with different objective values rather than having a single optimised alternative. The results are indicative but also promising and future research can lead to better quality results.

Nevertheless, during the development of the multi-objective optimisation methodology a number of limitations were unveiled. The first limitation originates from the mathematical model of the business process. The mathematical model focuses on activities and resources as its two main concepts and it ignores the participating (physical or mechanical) actors. This consequently results in what is criticised by Lindsay *et al.* (2003) as 'a mechanistic viewpoint of business processes'. However, it is more difficult for a formal business process modelling technique to capture the roles of the participants than a diagrammatic approach which visualises the flow of the process. Another limitation lies in the selection of the test process designs. In order to better assess the optimisation techniques used, an approach with a scalable range of problems was selected. To evaluate the algorithms' performance using a larger series of problems can better demonstrate the algorithms' behaviour by providing a more apparent performance overview. Another limitation is linked with the evaluation metric that was simple and did not take into account the diversity of the generated solutions. In the authors' opinion the limitations described above do not reduce the significance of the methodology that was followed or the quality of results. Addressing these limitations, would lead to improved results perhaps even more encouraging in terms of the optimisation algorithms' performance on multi-objective business processes optimisation.

Future research in the relevant area could focus on areas such as building more complete process models, testing more complicated process designs and exploring more efficient metrics. The construction of a business process model that can cover more aspects of a 'closer to real world' business process can be a complicated research area. Business processes in real world have features such as feedback loops or decision points. Modelling and optimisation of these aspects can prove a complicated task. Future research should also focus on selecting and customising the most appropriate techniques for business process multi-objective optimisation from a wider set of techniques and algorithms and thus locating more accurately the most suitable optimisation method in order to produce better results.

## 7. Conclusions

The paper presented a framework for applying multi-objective optimisation to business process designs. By developing a formal business process model and orienting it to multi-objectivity, the generation of optimised business process designs was facilitated and demonstrated using the travel agent example. What makes the business process optimisation problem distinctive is its highly constrained nature and the fragmented search space that has a significant impact on locating the optimum solutions. It is shown that state-of-the-art multi-objective optimisation algorithms, such as NSGA2 and SPEA2, produced satisfactory results by generating and preserving optimal solutions on process designs of different complexities. That provides an adequate number of alternative optimised process designs for the business analyst to decide the trade-offs between the different objectives. The results presented here are indicative and encouraging for further research in the area of business process multi-objective optimisation.

## References


Aguilar-Saven R. S., 2004. Business process modelling: Review and framework. International Journal of Production Economics 90, 129-149.

Bartz-Beielstein T., Mehnen J., Naujoks B., Schmitt K. and Zibold D., 2004. KEA - A software package for development, analysis and application of multiple objective evolutionary algorithms, viewed 20 December 2005, <http://www-isf.maschinenbau.uni-dortmund.de/veroeff/documents/2004_mehn_kea_a.pdf>.

Bititci U.S. and Muir D.,1997. Business process definition: a bottom-up approach. The International Journal of Operations and Production Management 17 (4), 365-374.

Byrne M.D. and Hossain M.M., 2005. Production planning: An improved hybrid approach. International Journal of Production Economics 93-94, 225-229.

Cheung Y. and Bal J., 1998. Process analysis techniques and tools for business improvements. Business Process Management 4 (4), 274-290.

Chafekar D., Xuan J. and Rasheed K., 2003. Constrained Multi-objective Optimization Using Steady State Genetic Algorithms. *In Proceedings of GECCO 2003,* Chicago, Illinois, USA, .pp. 813-824.

Deb K., Agrawal S., Pratap A. and Meyarivan T., 2000. A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. *KanGAL Report No. 200001*.

Deb K., 2001. Multi-objective optimisation using evolutionary algorithms, New York: John Wiley &

Sons.

Freschi F. and Repetto M., 2005. Multiobjective Optimization by a Modified Artificial Immune System Algorithm. Lecture Notes in Computer Science 3627, 248-261.

Grigori D., Casati F., Castellanos M., Dayal U., Sayal M. and Shan M.C., 2004. Business Process Intelligence. Computers in Industry 53, 321-343.

Hofacker I. and Vetschera R., 2001. Algorithmical approaches to business process design. Computers & Operations Research 28, 1253-1275.

Kennedy J. and Eberhart R., 1995. Particle Swarm Optimisation. IEEE International Conference on Neural Networks, Conference Proceedings 4, 942-1948.

Kennedy J. and Eberhart R., 1999. The Particle Swarm: social adaptation in information-processing systems. In: Corne D., Dorigo M. and Glover F., (Eds.) New Ideas in Optimisation, 379-388. Cambridge: McGraw-Hill.

Koubarakis M. and Plexousakis D., 2002). A Formal Framework for Business Process Modelling and Design. Information Systems 27, 299-319.

Li H., Yang Y. and Chen T.Y., 2004. Resource constraints analysis of workflow specifications. The Journal of Systems and Software 73, 271-285.

Lindsay A., Downs D. and Lunn K., 2003. Business processes - attempts to find a definition. Information and Software Technology 45, 1015-1019.

McKay A. and Radnor Z., 1998. A characterization of a business process. The International Journal of Operations and Production Management18 (9/10), 924-936.

Sadiq W. and Orlowska E. M., 2000. Analysing process models using graph reduction techniques. Information Systems 25, 117-134.

Shimizu Y. and Sahara Y., 2000. A supporting system for evaluation and review of business process through activity-based approach. Computers and Chemical Engineering 24, 997-1003.

Vaughan T.S., 2006. Lot size effects on process lead time, lead time demand, and safety stock. International Journal of Production Economics 100, 1-9.

Yu C-S. and Li H-L., 2000. A robust optimization model for stochastic logistic problems. International

Journal of Production Economics 64, 385-397.

Zakarian A., 2001. Analysis of process models: A fuzzy logic approach. The International Journal of Advanced Manufacturing Technology 17, 444-452.

Zitzler, E. and Thiele L., 1999. Multi-objective evolutionary algorithms: A comparative case study and the strength pareto approach. IEEE Transactions on Evolutionary Computation 3, 257-71.

Zitzler E., Laumanns M. and Thiele L., 2001. SPEA2: Improving the Strength Pareto Evolutionary Algorithm. TIK Report Nr. 103, Computer Engineering and Networks Lab (TIK), Swiss Federal Institute of Technology (ETH) Zurich.

## Appendix

The Appendix describes the remaining four test business process designs and their results. Table 4 presents a summary of the designs' main parameters demonstrating also their increasing complexity. The optimisation results of these test problems are demonstrated in figures 8 and 9 below.

| Parameters | ActivitiesST2 | ActivitiesST3 | ActivitiesST4(20) | ActivitiesST5 |
|---|---|---|---|---|
| No. of input resources: | 2 | 2 | 2 | 3 |
| No. of output resources: | 2 | 2 | 2 | 3 |
| No. of participating activities: | 2 | 3 | 4 | 5 |
| Library size: | 10 | 10 | 20 | 20 |

Table 4. Summary of parameters for the remaining test business process designs

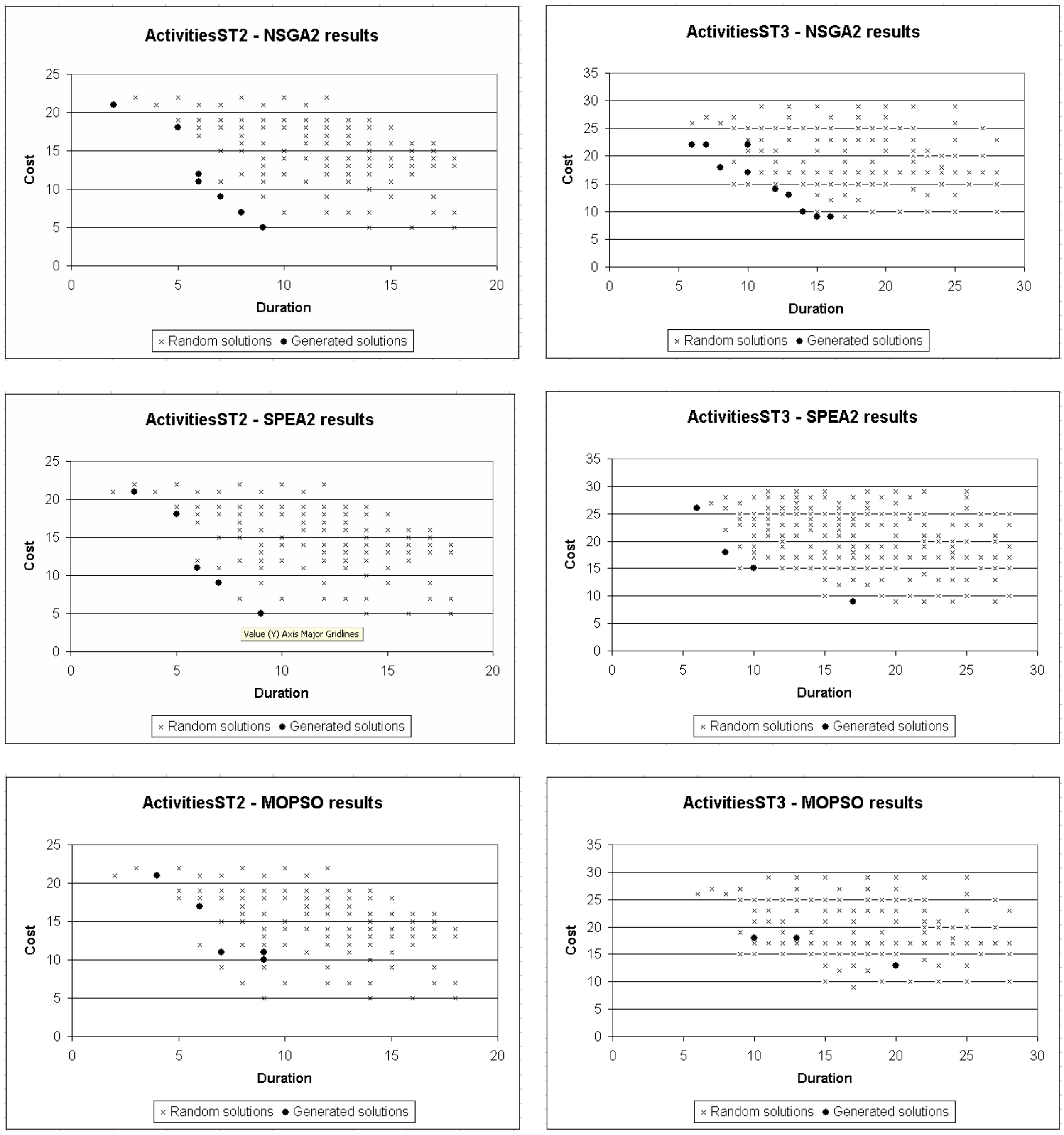



Figure 8. Generated solutions for ActivitiesST2 and ActivitiesST3

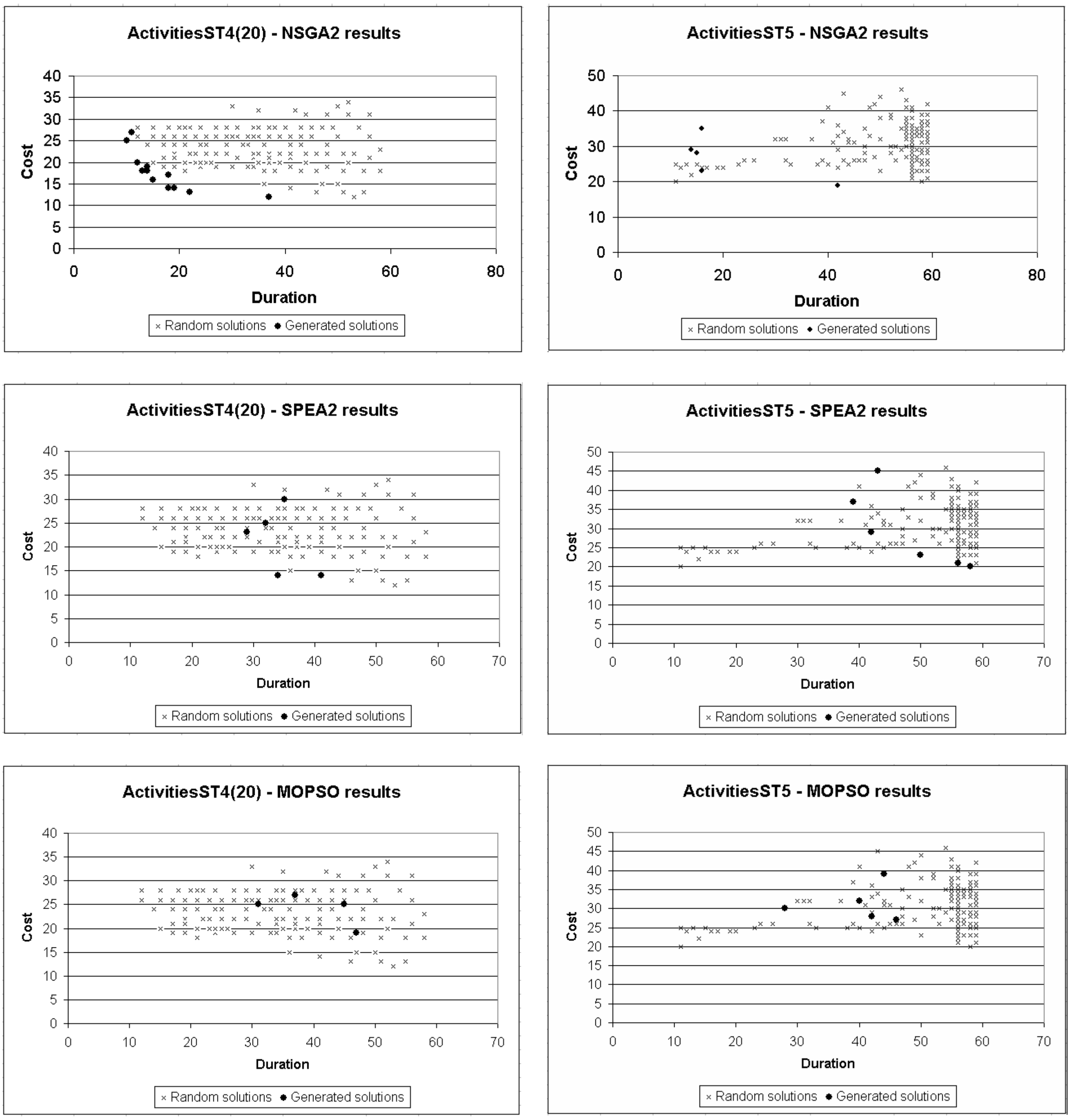



Figure 9. Generated solutions for ActivitiesST4(20) and ActivitiesST5

# Optimisation of business process designs: An algorithmic approach with multiple objectives.

Vergidis, K.

2007-09

K. Vergidis, A. Tiwari, B. Majeed, R. Roy, Optimisation of business process designs: An algorithmic approach with multiple objectives, International Journal of Production Economics, Volume 109, Issues 1-2, Special Section on Cost Engineering, September 2007, Pages 105-121.

http://dx.doi.org/10.1016/j.ijpe.2006.12.032