Evolutionary Multi-objective Optimization of Business Process Designs with MA-NSGAII

Nadir Mahammed[1](✉), Sidi Mohamed Benslimane[1](✉), and Nesrine Hamdani[2](✉)

[1] LabRI-SBA, Ecole Supérieure en Informatique 08 Mai 1945,
BP 73 Bureau de poste EL WIAM, 22016 Sidi Bel Abbes, Algeria
{n.mahammed, s.benslimane}@esi-sba.dz

[2] LIO Laboratory, Ahmed Ben Bella University of Oran 1,
BP 1524 EL MNaouer, 31000 Oran, Algeria
hamdani.nesrine@edu.univ-oran1.dz

Abstract. Optimization is known as the process of finding the best possible solution to a problem given a set of constraints. The problem becomes challenging when dealing with conflicting objectives, which leads to a multiplicity of solutions. Evolutionary algorithms, which use a population approach in their search procedures, are advised to suitably solve the problem. In this article, we present an approach for an evolutionary combinatorial multi-objective optimization of business process designs using a variation of NSGAII, baptized MA-NSGAII. The variants of NSGAII are numerous. In fact, the vast majority deals either with the crossover operator or with the crowding distance. We discuss an optimization Framework that uses (i) a proposal of effective Fitness function, (ii) 02 contradictory criteria to optimize and (iii) an original selection technique. We test the proposed Framework with a real life case of multi-objective optimization of business process designs. The obtained results clearly indicate that an effectual Fitness function combined with the appropriate selection operator affects undeniably quality and quantity of solutions.

Keywords: Multi-objective optimization · Evolutionary computing
Genetic algorithm · Selection operator · Business process

1 Introduction

The optimizing of business processes (BP) is considered the problem of building feasible BPs while optimizing criteria such as reducing execution time and minimizing the resource cost. To achieve that goal, evolutionary algorithms (EA), and more specifically genetic algorithms (GA) have been much talked about. Generally, these optimization techniques are indoctrinated by 02 important questions: Exploration and Exploitation. If exploitation refers to the tendency of the algorithm to direct its search using information obtained in the past and to determine promising regions, for later research [1], exploration, for its part, explores new and unknown areas in the research space to find promising areas. In search of optimization, exploitation is carried out by selection operator, and exploration of the search space is carried out by other -search-operators in EAs [2].

A. Amine et al. (Eds.): CIIA 2018, IFIP AICT 522, pp. 341–351, 2018.
https://doi.org/10.1007/978-3-319-89743-1_30

Fogel [3] describes selection operator as the action of selecting more fitting individuals, by analogy to Darwin's theory of evolution (survival of the strongest). All individuals have a chance of being selected in the population, but there is a chance that an individual can be selected more than once depending on its Fitness [4]. In an optimization matter, these characteristic determine the convergence of GA [5].

This article proposes a Framework that deals with a MOO of BP designs (called BPMOO), by reducing the cost and minimizing the duration. The Framework uses an original EA: MA-NSGAII, for Mass selection based NSGAII. MA-NSGAII tests and experiments the influence of the proposed new selection operator; inspired by viticulture, Mass selection. Section 2 presents a state of the art on BPMOO exclusively with NSGAII. Section 3 presents the proposed Framework with its Fitness function aiming the optimization issue, and introduces MA-NSGAII. Section 4 presents the experimental phase of the Framework applied on a test scenario based on BP designs optimization, then evaluates and discusses the obtained results. Finally, Sect. 5 summarizes the proposed research and provides future work directions.

2 Related Work

To overcome the question of BPMOO, NSGAII is one of the most widely used EAs [6, 7]. The first work to quote is [8]. They focus on how to appropriately allocate resources to activities, in BP designs to ensure its high performance, using NSGAII.

A series of work on BPMOO with EAs, in general, and NSGAII, in particular are introduced in [9]. The proposed approach uses a formal definition for a BP (based on [8]). Multi-objectivity is expressed in terms of cost and duration of BP designs. They propose and test a Framework using NSGAII, for generating new BP designs. Thereafter, [9, 10] present the most important work in this field. They have improved [9] work by adding (i) the ability to review or reconfigure any unfeasible BP design resulting from NSGAII iterations, (ii) to compare the efficiency of NSGAII with other EAs. Finally, to reach to the work of [11, 12], they propose a Framework for optimizing BP designs, where each task composing a BP can be seen a Web service. Wibig [13] proposes a Framework for BPMOO using Petri networks for modeling. He uses dynamic programming to reduce the computation time required. Farsani et al. [14] modifies the mutation and crossover operators used by NSGAII. Mahammed et al. [6] are interested in a BPMOO (up to 03 criteria). They proposed a Framework that combines an original Fitness function with NSGAII with a modified crossover operator.

The work on a multi criteria optimization with NSGAII is legion, and few to have proposed to review or modify NSGAII's selection operator. Ishibuchi et al. [15] propose a new selection technique with NSGAII, in MOO. The proposed technique is a two-stage selection mechanism (i) a standard selection based on individual Fitness is applied, then, (ii) tournament selection is used. Emmerich et al. [16] use NSGAII to arrive at an evolutionary steady-state algorithm (it produces only an individual at each iteration). They combine NSGAII with a selection operator based on hyper-volume measurement. Trivedi et al. [17] propose to review all genetic operators of NSGAII. The binary tournament selection technique limited by constraints without parameters is used to efficiently manage constraints. Phan et al. [18] study a method to aggregate existing indicator-based selection operators. They show that a boosted selection operator outperforms exiting ones in optimality, diversity and convergence velocity. Zhong et al. [19] present an interesting study on the reduction of solutions diversity following a MOO (02 criteria) using NSGAII with truncation selection. Mahammed et al. [20] propose an evolutionary multi-criteria approach based on a modified EA for generating optimized business processes. They replace the binary tournament selection with uniform selection, roulette wheel selection and ranking selection, combining the whole with a proposed crossover operator. The proposed framework improves the results obtained by [6]. In the present work, the authors present an approach for a BPMOO using NSGAII with an unusual selection operator, Mass selection.

3 Proposed Approach

The current study gives rise to a Framework capable of combinatorial MOO of BP designs, using a modified NSGAII. It aims to generate a series of optimized designs, with reduced cost and minimized duration. Thereafter, the main steps of the proposed Framework are depicted. Then, MA-NSGAII with Mass selection is explained.

3.1 Overall Architecture of the Proposed Framework

Following a number of steps, the proposed Framework is able to generate a series of optimized designs from an initial BP. The authors define a BP as a set of activities that takes one or more kinds of input and creates an output that has value to the customer [20]. Throughout the Framework course (Fig. 1), each design must fulfill a certain amount of constraints. MA-NSGAII is used to generate BP designs, each one has (i) a feasible graphical representation and (ii) optimized' attributes values.

1. Create an initial population. The first step of the proposed optimization Framework is to generate a random population of BP designs. It takes place only once in the Framework's progress, then the population evolves for a defined number of generations. The steps 2–5 are repeated for a predefined number of iterations.
2. Create designs representation. For each design, a mathematical representation is generated. 02 distinguished matrixes are used (i) the first for representing the relationship between resources and tasks composing a potential BP design. (ii) The second matrix portrays the tasks attributes (i.e. optimization criteria) of each design.
3. Verify and apply the restraints. The proposed Framework checks constraints prior the evaluation of each individual, because its design might be modified, thereafter. As restraints, we quote:

   - Each design must have a bounded size.
   - A task must appear once, in each design.
   - Each task must take birth form one or more resources, or BP inputs resources.
   - Each task must be linked to another task by one or more resources, or BP outputs resources.

- Replace or delete any task or resource useless in a design.
- Verify inputs and outputs of each design.

4. Assess designs. It involves calculating each BP design's Fitness value. It is based on each solution attributes values in its design. Knowing that the Fitness function is at the heart of an evolutionary computing application, the proposed Framework uses a Fitness function dealing with 02 criteria, (i) minimizing the delivery price of the service (i.e. cost) and (ii) minimizing the delivery duration of the service (i.e. duration). The solutions are evaluated after the restraints verification because only tasks that really participate in a BP design are taken into account

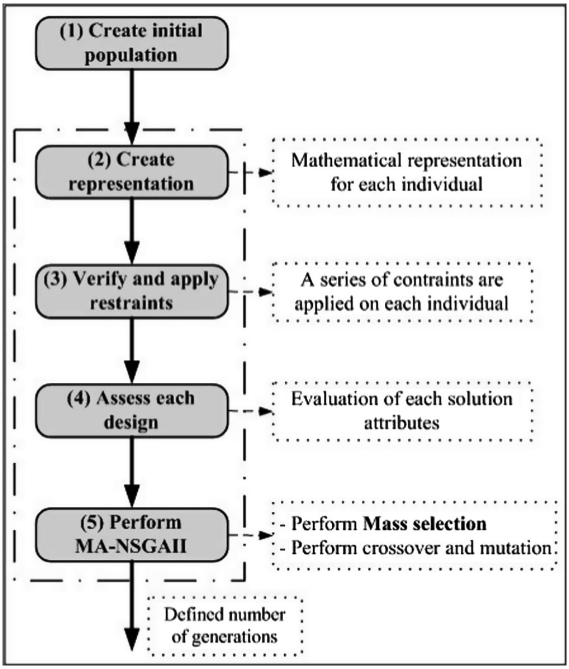
(1) Create initial population
(2) Create representation
Mathematical representation for each individual
(3) Verify and apply restraints
A series of contraints are applied on each individual
(4) Assess each design
Evaluation of each solution attributes
(5) Perform MA-NSGAII
- Perform **Mass selection**
- Perform crossover and mutation
Defined number of generations

Fig. 1. The proposed optimization Framework main steps.

To correctly represent the Fitness function F(x), the authors propose to use the Pythagorean Theorem[1]. F(x) can be symbolized geometrically as the hypotenuse of the right-angled triangle formed by C and D. This choice was decided by the absolute necessity of not favoring any particular objective, in an MOO [20]. The Fitness function may be formulated:

$$F(x) = \sqrt{C^2 + D^2}$$

With $C = \sum_1^{n_d} c_i$, $D = \sum_1^{n_d} d_i$ Where C and D are normalized

[1] Maor, E. (2007). *The Pythagorean Theorem: a 4,000-year history*. Princeton Univ. Press.

C: Cost of the BP design.
D: Duration of the BP design.
n_d: Number of tasks in the BP design.
c_i: Cost of a task i being a part of the BP design.
d_i: Duration of a task i from the BP design

5. Perform MA-NSGAII. After the evaluation, MA-NSGAII is applied. First, a simulated binary crossover is performed. The process does not check whether the solution is feasible, it is the concern of step 3. Then, mutation operator is performed. Finally, Mass selection is performed regarding to all new solutions. Subsection III-B introduces MA-NSGAII with Mass selection.

3.2 MA-NSGAII

MA-NSGAII is inspired from NSGAII proposed by [21], one of its main features is to ensure the diversity of the population throughout iterations. To do so, the authors propose to replace the simulated binary tournament selection operator recommended by [21] by Mass selection operator (see Fig. 2). This technique has never been used with NSGAII, by the past. The authors propose to add experiments comparison with other techniques of selection: Truncation selection [22] and roulette wheel selection [23], to demonstrate the effectiveness of MA-NSGAII.

Mass selection is native to traditional agriculture, mainly viticulture [24]. The use of such a technique goes back quite a long time [25–27] and never in the context of BPMOO. According to [24], Mass selection consists in choosing breeders according to their individual aspect performance(s). They explain that it is easy to achieve, and advised in the situation where no followed-up of individuals is available or required. To the knowledge of the authors, Mass selection has never been used with a GA, in a MOO context. This article proposes to replace the selection technique usually used within NSGAII by Mass selection, and compare the results using traditional tournament selection, truncation selection and roulette wheel selection. Knowing that Mass selection is purely used in biology research, the authors propose to implement it by taking inspiration from roulette wheel selection (see [28]).

Mass selection algorithm can be summarized as follows:

1. Evaluate Fitness f_i of each individual in the population.
2. Compute the probability of selecting each individual $p_i = f_i / \sum_{j=1}^{N} f_j$.
3. Calculate the sum of all individuals Fitnesses in the population $S = \sum_{i=1}^{N} f_i$.
4. Generate a uniform random number from interval $r \in]0, S]$.
5. Calculate the cumulative probability $q_i = \sum_{j=1}^{i} p_j + \frac{\sum_{j=1}^{m} c_j}{m} + m_{norm}$.
6. If $r \leq q_i$ then select individual i.
7. Repeat 4 to 6 N times.

Mass selection operator considers, in addition to Fitness function value of each individual, 02 parameters (i) the mean between the chosen criteria MOO of these

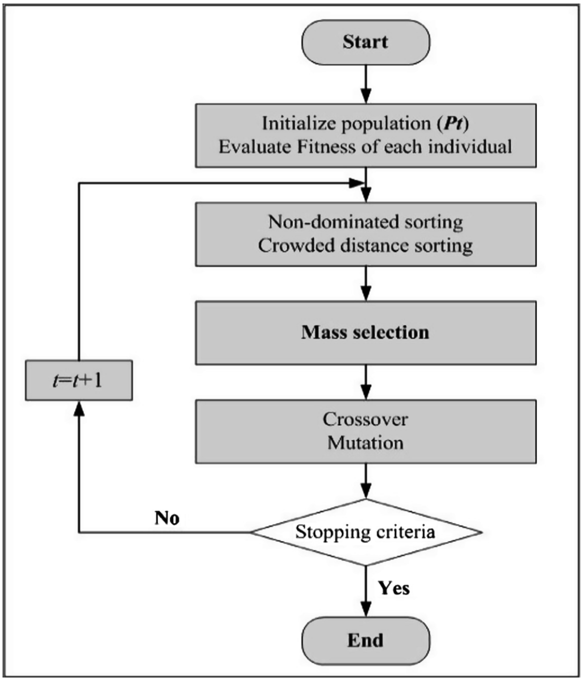

Fig. 2. MA-NSGAII's operating.

individuals and (ii) a value attributed to each solution for the appearance (e.g. aspect[2]). This implementation was chosen by the nature of Mass selection to take into account more external appearances of individuals. Which, compared to the others selection techniques adds more credibility, in its unfolding.

4 Experimentation and Results

The test scenario describes "Sales forecasting" from which new designs must be generated and optimized, proposed by [29]. The scenario takes as inputs (a) the company name and (b) the market update request, and produces as output (c) a report containing the forecast results of the contract.

The work required by the proposed Framework is to produce new optimized BP designs from the sales forecast scenario. The generation of these solutions is done using a tasks library (20 tasks), and a set of resources known and limited (09 resources). An evolutionary combinatorial MOO with 02 optimization criteria is used (i) the minimization of BP designs cost, and (ii) minimizing their duration.

[2] Genes' number of a solution.

Table 1. The proposed Framework parameters (inspired form [20]).

Parameter	Value
Tasks library size	20
Number of available resources	9
Number of attributes per task	2
Minimum BP design size	4
Maximum BP design size	6
Initial BP input resources	{a, b}
Initial BP output resources	{c}
Cost values' interval	[200 230]
Duration values' interval	[300 390]
Selection operator	Mass selection operator
Crossover operator	Simulated binary crossover
Mutation operator	One point mutation

Table 2. Fitness values according to selection techniques.

Population size	Selection operator	Solution size (tasks)		
		4	5	6
		Minimal Fitness		
500	Tournament	963	1192	1419
	Roulette wheel	963	1190	1424
	Truncation	958	1190	1421
	Mass	955	1185	1416
100	Tournament	963	1192	1419
	Roulette wheel	955	1188	1418
	Truncation	955	1183	1418
	Mass	954	1183	1411

Table 1 summarizes the parameters used by the proposed optimization Framework.

Table 2 shows the different Fitness values obtained by the proposed Framework according to the selection operator applied, for the BPMOO. The results vary according to the initial population size chosen (500 then 100 individuals) and solutions size obtained (depending on BP design's size: 4, 5 then 6 tasks).

A number of experiments have been performed to assess the capabilities of the proposed Framework using a MSI GT70 laptop with NetBeans 8.1 IDE and Java 8. The authors chose the minimum Fitness value of each solution as a parameter to evaluate each solution. The obtained results by [11] are resumed with the tournament selection. As explained in Subsection II-B, Mass selection is the only technique that gives importance to solution's appearance during the evaluation process. This feature makes it possible to add further criteria, for solutions estimation during BPMOO. Another interesting point is the proximity of resulting Fitness values. It can be explained by the

Table 3. Optimization criteria values according to selection techniques.

Selection	Solution size (task)	Cost	Duration
Tournament [11]	4	448	853
	5	553	1056
	6	662	1257
Roulette wheel	4	448	847
	5	550	1053
	6	659	1256
Truncation	4	447	844
	5	554	1050
	6	662	1252
Mass	4	446	843
		447	844
	5	555	1045
		558	1048
	6	658	1248
		658	1252

discrete nature of the values used in the BPMOO studied, even if the optimization Framework obtained the most interesting solutions.

Table 3 shows the optimization criteria values obtained by the proposed Framework, i.e. minimize BP design cost and duration. The results are obtained with an initial population of 500 then 100 individuals (see Table 2), a number of iterations up to 20 iterations. The design size ranges from 04 tasks to 06 tasks according to [11]. Table 3 shows -clearly- that BPMOO Framework provides the finest results with MA-NSGAII, in comparison with [11]. Figure 3 compares selection techniques used by the proposed

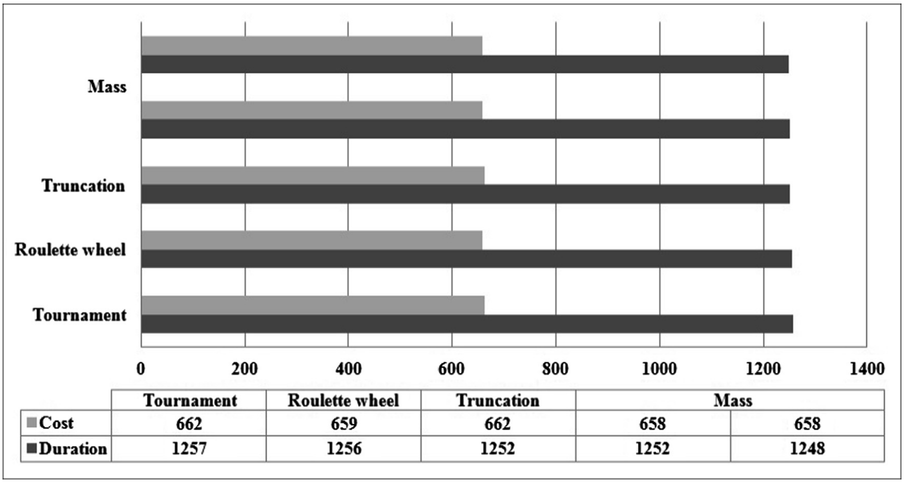

Fig. 3. Solutions comparison (Design size = 6 tasks).

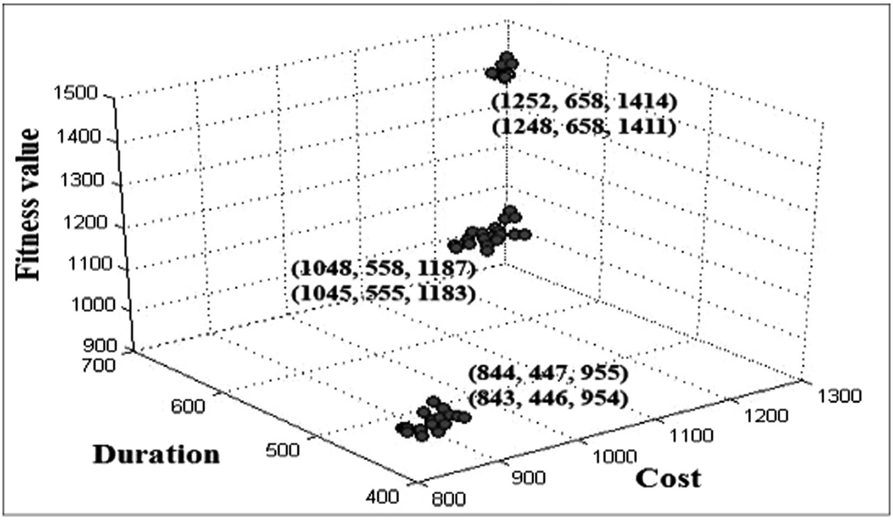

Fig. 4. Solutions and their Fitness (with MA-NSGAII).

Framework, with BP designs size equal to 06 tasks. The results show that using MA-NSGAII offers more solutions that meet optimization requirements (02 solutions per design size, against 01 solution with other selection techniques). The difference between the different solutions according to the selection technique used is certainly minimal. It may be explained by intervals' values (cost and duration) used by the Framework (Table 1). The results obtained by the proposed Framework with MA-NSGAII are shown in Fig. 4. Three (03) clusters of solutions are generated.

Each cluster corresponds to the average size of a solution, i.e. each BP design size should be between 04 and 06 tasks, making a cluster per value. A triplet as (843, 446, 951) represents: duration, cost and Fitness values of a BP design, respectively.

Although the results seem better than [11], it seems obvious that improvements can be made to the presented work. Adding other optimization criteria with a form of dependency between them could help to evaluate with confidence the quality of each potential solution. Another factor had a significant impact on the presented study is the tasks library size and the amount of resources. It turns out that a larger library (with more resources) could improve the obtained solutions (at least quantitatively). It appears to the authors that the selection operator as much as the crossover operator (see [20]) has a moderate effect on the generated solutions quality.

5 Conclusion

The presented work aims is to deal with a BPMOO using an EA, *i.e.* to generate feasible BP designs, in a combinatorial MOO environment, using an implemented Framework. The proposed Framework, apart from a mathematical representation of BP designs, uses MA-NSGAII, a modified version of NSGAII, with an original Fitness function proposal. One of MA-NSGAII specifications consists on replacing the traditional selection technique by a different one: Mass selection. To demonstrate the efficiency and effectiveness of the optimization Framework many selection techniques have been identified and used. It has an experimental case explaining an evolutionary combinatorial BPMOO problem. A rather confusing fact emerges. Mass selection, which has not been used in this field of research by the past, has yielded the most convincing results. The obtained outcomes are better either through the quality of solutions (Fitness value), the quantity (number of solutions) and even the population size used for experimentations. The results have demonstrated that the Framework with the help of MA-NSGAII, has enhanced its capability to generate diverse BP designs with optimal objective values.

Several issues can be addressed. For example, it is desirable to do more work on how an evolutionary bio inspired algorithm reacts within the proposed selection technique. Also, by incorporating information about the MOO context (e.g. servicing business processes). More experimentation (e.g. different real life scenarios) would be welcomed.

References

1. Thierens, D., Goldberg, D.: Convergence models of genetic algorithm selection schemes. In: Davidor, Y., Schwefel, H.-P., Männer, R. (eds.) PPSN 1994. LNCS, vol. 866, pp. 119–129. Springer, Heidelberg (1994). https://doi.org/10.1007/3-540-58484-6_256
2. Holland, J.H.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Harbor (1975)
3. Fogel, D.B.: Evolutionary Computation: Toward a New Philosophy of Machine Intelligence. IEEE, Piscataway (1995)
4. Goldberg, D.E., Deb, K.: A comparative analysis of selection schemes used in genetic algorithms. Found. Genet. Algorithms **1**, 69–93 (1991)
5. Mitchell, M.: An Introduction to Genetic Algorithms. MIT Press, Cambridge (1998)
6. Mahammed, N., Benslimane, S.M.: Toward Multi Criteria Optimization of Business Processes Design. In: Bellatreche, L., Pastor, Ó., Almendros Jiménez, J.M., Aït-Ameur, Y. (eds.) MEDI 2016. LNCS, vol. 9893, pp. 98–107. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45547-1_8
7. Zhou, Y., Chen, Y.: The methodology for business process optimized design. In: IECON Proceedings (Industrial Electronics Conference), vol. 2, pp. 1819–1824 (2003)
8. Hofacker, I., Vetschera, R.: Algorithmical approaches to business process design. Comput. Oper. Res. **28**(13), 1253–1275 (2001)
9. Vergidis, K., Tiwari, A., Majeed, B., Roy, R.: Optimisation of business process designs: an algorithmic approach with multiple objectives. Int. J. Prod. Econ. **109**(1), 105–121 (2007)
10. Vergidis, K., Saxena, D., Tiwari, A.: An evolutionary multi-objective framework for business process optimisation. Appl. Soft Comput. **12**(8), 2638–2653 (2012)
11. Vergidis, K., Turner, C., Alechnovic, A., Tiwari, A.: An automated optimisation framework for the development of re-configurable business processes: a web services approach. Int. J. Comput. Integr. Manufact. **28**(1), 41–58 (2015)
12. Georgoulakos, K., Vergidis, K., Tsakalidis, G., Samaras, N.: Evolutionary multi-objective optimization of business process designs with pre-processing. In: 2017 IEEE Congress on Evolutionary Computation (CEC), pp. 897–904. IEEE (2017)

13. Wibig, M.: Dynamic programming and genetic algorithm for business processes optimisation. Int. J. Intell. Syst. Appl. **5**(1), 44 (2012)
14. Farsani, S.T., Aboutalebi, M., Motameni, H.: Customizing NSGAII to optimize business processes designs. Res. J. Recent Sci. **2**, 74–79 (2013)
15. Ishibuchi, H., Shibata, Y.: A similarity-based mating scheme for evolutionary multiobjective optimization. In: Cantú-Paz, E., et al. (eds.) GECCO 2003. LNCS, vol. 2723, pp. 1065–1076. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-45105-6_116
16. Emmerich, M., Beume, N., Naujoks, B.: An EMO algorithm using the hypervolume measure as selection criterion. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 62–76. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-31880-4_5
17. Trivedi, A., Pindoriya, N.M., Srinivasan, D.: Modified NSGA-II for day-ahead multi-objective thermal generation scheduling. In: IPEC 2010, pp. 752–757. IEEE (2010)
18. Phan, D.H., Suzuki, J.: Boosting indicator-based selection operators for evolutionary multi objective optimization algorithms. In: 23rd IEEE International Conference on Tools with Artificial Intelligence (ICTAI), pp. 276–281. IEEE, November 2011
19. Zhong, X., Zhao, Y., Han, Q.: An improved non dominated sorting multi objective genetic algorithm and its application (2015)
20. Mahammed, N., Benslimane, S.M.: An evolutionary algorithm based approach for business process multi-criteria optimization. Int. J. Organ. Collect. Intell. (IJOCI) **7**(2), 34–53 (2017)
21. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: Schoenauer, M., Deb, K., Rudolph, G., Yao, X., Lutton, E., Merelo, J.J., Schwefel, H.-P. (eds.) PPSN 2000. LNCS, vol. 1917, pp. 849–858. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-45356-3_83
22. Crow, J.F., Kimura, M.: Efficiency of truncation selection. Proc. Natl. Acad. Sci. **76**(1), 396–399 (1979)
23. Mühlenbein, H., Voigt, H.M.: Gene pool recombination in genetic algorithms. In: Osman, I. H., Kelly, J.P. (eds.) Meta-Heuristics, pp. 53–62. Springer, Boston (1996). https://doi.org/10.1007/978-1-4613-1361-8_4
24. Leroy, G., Verrier, E., Meriaux, J.C., Rognon, X.: Genetic diversity of dog breeds: within-breed diversity comparing genealogical and molecular data. Anim. Genet. **40**(3), 323–332 (2009)
25. Liyanage, D.V.: Identification of genotypes of coconut palms suitable for breeding. Exp. Agric. **3**(03), 205–210 (1967)
26. Magnussen, S., Yeatman, C.W.: Predictions of genetic gain from various selection methods in open pollinated Pinus banksiana progeny trials. Silvae Genetica **39**(3–4), 140–153 (1990)
27. Berthouly, C., Leroy, G., Van, T.N., Thanh, H.H., Bed'Hom, B., Nguyen, B.T., Maillard, J.C., et al.: Genetic analysis of local Vietnamese chickens provides evidence of gene flow from wild to domestic populations. BMC Genet. **10**(1), 1 (2009)
28. Mahammed, N., Benslimane, S.M., Hamdani N.: An approach to addressing a business process multi-objective optimization issue with MA-NSGAII. In: EDiS 2017, University of Oran 1, Oran, Algeria, 17–18 December. IEEE (2017). (in press)
29. Grigori, D.: Business process intelligence. Comput. Ind. **53**, 321–343 (2004)