Controlling the NSGA-II Algorithm convergence toward a fixed Pareto-optimal solution for the Gross Domestic Product quarterly disaggregation

Raïmi Aboudou Essessinou, Guy Degla

Institute of Mathematics and Physical Sciences, University of Abomey-Calavi (UAC) , Dangbo, Benin.

Email: raimi.essessinou@imsp-uac.org; gdegla@imsp-uac.org

Abstract - **In this paper, we test the convergence speed of the fast elitist Non-Dominated Sorting Genetic Algorithm (NSGA-II) on the Gross Domestic Product (GDP) quarterly disaggregation problem. In fact, we perform many simulations by considering various inputs of the NSGA-II parameters with respect to the population size (*pop_size*), the iteration number (*no_rum*), the maximum number of generations (*gen_max)* and the mutation distribution index in the polynomial mutation (*etam*). It turns out that for suitably large values of the parameters *pop_size, no_rum* and *gen_max*, the NSGA-II algorithm converges to a single Pareto-optimal solution located in a fixed cuboid. Due to the elitism in the NSGA-II and our simulation results, we suspect that the mutation distribution index alone doesn't influence the time of algorithm convergence and the number of Pareto-optimal solutions. Therefore, we reach the most likely conclusion that only an appropriate choice of the parameter triplet (*pop_size, no_rum, gen_max)* can ensure the convergence of the algorithm toward a fixed Pareto-optimal solution for the Quadratic Multi-objective Programming of GDP quarterly disaggregation. It is worth noted that our work is an extension of previous works in the same field.**

Keywords* - *Simulation, Convergence, Multiobjective Optimization, NSGA-II Algorithm, Pareto-optimal front, Quarterly Gross Domestic Product

I. INTRODUCTION

Several algorithms have been developed in the literature to solve multi-objective optimization problems, but most of these multi-objective optimization problems are Non Polynomial (NP)-difficult according to their size. Generally, all the algorithms can be classified into two types of approaches: (i) non-Pareto approaches that seek to reduce the initial problem to single-objective problems and (ii) Pareto approaches that do not transform the objectives of the problem and treat them simultaneously by seeking better compromise solutions [2] according to the resolution process.

A multi-objective programming problem is generally formulated as presented in equation (1):

$$\min_{x \in U} F(x) = \left(f_1(x), f_2(x), \dots, f_Q(x)\right) \tag{1}$$

where $F: \mathbb{R}^N \longrightarrow \mathbb{R}^Q$ is the vector of objective functions, f_i represents the q^{th} objective function, Q is the number of objective functions and generally $Q \geq 2$ and $U \subseteq \mathbb{R}^N$ the set of all the equality and inequality constraints. A solution x is a vector of N decision variables: $x = (x_1, x_2, \dots x_N)$, $x \in \mathbb{R}^N$.

In the literature, many multi-objective evolutionary algorithms have been suggested for solving multi-objective programming. Recently, it has been proved that simulation results on difficult test problems showed that the Non-dominated Sorting Genetic Algorithm (NSGA)-II, in most problems, is able to find much better range of solutions and better convergence speed near the true Pareto-optimal front [4]. NSGA-II is a very effective algorithm but is often criticized for its computational complexity.

The selected population and the chosen parameters generate off springs from crossover and mutation operators. So a bad choice of the parameters could affect negatively the convergence of the algorithm.

The step-by-step process shows that NSGA-II algorithm is simple and straightforward. NSGA-II is a better sorting algorithm which incorporates elitism and no sharing parameter needs to be chosen a priori. Since it is not possible to find a solution that simultaneously optimizes all objective functions in the case of a multi-objective program, the dominance in the sense of Pareto is introduced in the algorithm.

The fastness of NSGA-II and the convergence speed of the algorithm depend on the choice of the parameters. A test on the parameters will be discussed in detail in this paper.

In fact, it is difficult to ensure perfect cooperation between objective functions in the search for Pareto optimal solutions with a lot of objective functions. In fact, a multi-objective optimization program gives multiple Pareto optimal solutions in general.

For the sake for looking for a suitable algorithm that solving the quarterly disaggregation of Gross Domestic Product (GDP) as a multi-objective programming, our aim in this paper is to test the robustness of the convergence and the speed towards single Pareto-optimal solution, using the solving algorithm proposed in [8]. Therefore, we test how fast the elitist multi-objective NSGA-II can converge towards a fixed Pareto optimal point if we control the key parameters, mainly, the mutation distribution index in the polynomial mutation. For this purpose, we use the multi-objective programming problem of GDP quarterly disaggregation developed in [7]. This paper is an extension of previous works in the same field. In section II, we present the quarterly disaggregation problem of GDP. The structure of solving algorithm in described in section III and the simulation results a discussed in section IV.

II. THE QUARTERLY DISAGGREGATION PROBLEM OF GDP

The quarterly disaggregation of Gross Domestic Product (GDP) problem is the new one proposed and developed in [7]. All the elementary objective functions identified are grouped in equation (2) with the constraints given by equations (3)-(7) to form the above multi-objective programming:

$$\min_X \{(f_1(X), f_2(X), f_3(X))\} \tag{2}$$

Subject to

$$\mathrm{X} = (X_1, X_2, X_3) \tag{3}$$

$$X_k = (X_{k,t})_t \;\; ; \; \forall\, t \, ; \; k \tag{4}$$

$$-X_{k,t} \le 0 \;\; ; \quad \forall \; t\,,k \tag{5}$$

$$\sum_{t=4y-3}^{4y} X_{k,t} - Z_{k,y} = 0 \;\; ; \;\; \forall\, y \;, \;\; k \tag{6}$$

$$\sum_{t=4y-3}^{4y} \frac{X_{k,t}}{I_{k,t}} \eta_{k,t} = \frac{Z_{k,y}}{\sum_{t=4y-3}^{4y} I_{k,t}} \; ; \; \forall \; k\,, y \tag{7}$$

Where

- T: the number of years for national accounts observed,
- $y \in \{1, 2, 3, \dots, T\}$, year generic index,
- $i \in \{1, 2, 3, 4\}$, quarterly index,
- $t \in \{1, 2, 3, \dots, 4T\}$: generic index of quarters on the T years' period obtained from i and y using an operator proposed in [6]:
- $k \in \{1, 2, 3\}$, generic branch index of the quarterly accounts nomenclature;
- Value of the annual account per branch: let $Z_{k,y}$ be the (known) value added of the branch account k , for the year y;
- quarterly indicator: let $I_{k,t}$ be the value of the branch indicator k , for the quarter $t = 1, 2, 3, \dots., 4T$;
- $I_k = (I_{k,t})_{t=1,2,..4T}$: the vector of quarterly indicator related the branch k over the entire period;
- Inter-branch interaction: we note $\overline{W}_{j,k}$ the interaction of the branch j on the branch k , considered as the average share of the branch k demand of product coming from the branch j : $\overline{W}_{j,k} = 1 \; if \; j = k$ and $0 \le \overline{W}_{j,k} < 1 \; if \; j \ne k$.
- Value added per branch: let $X_{k,t}$ be the value of national account for branch k at quarter $t = 1, 2, 3, \dots., 4T$
- The vector of national account value for branch k over the entire period is noted: $X_k = (X_{k,t})_{t=1,2,..4T}$, $X_k \in \mathbb{R}^{4T}$; $for \; k = 1, 2, 3$;
- The vector of quarterly national accounts value for all branches over the entire period is noted: $\mathrm{X} = (X_1, \; X_2, \; X_3), \; X \in \mathbb{R}^{4T \times 3}$;
- Thus, the objective functions are given by equation (8):

$$f_k(X_1, X_2, X_3) = \sum_{j=1}^{3} \sum_{t=2}^{4T} \overline{W}_{j,k} \left(\frac{X_{j,t}}{I_{j,t}} - \frac{X_{j,t-1}}{I_{j,t-1}} \right)^2 \tag{8}$$

III. THE ALGORITHM DESIGN

The algorithm for GDP quarterly disaggregation problem presented as a multi-objective programming in [8], [9], has the architecture as presented in Figure 1.

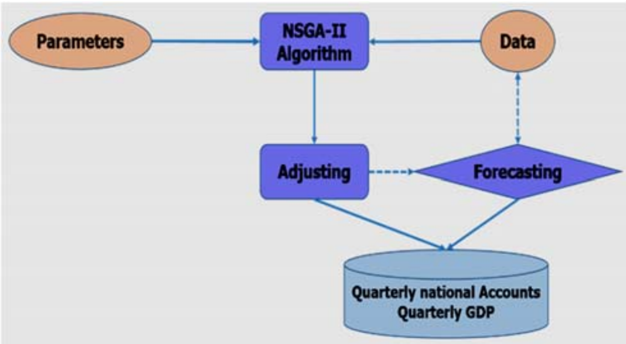

Figure 1. Architecture of the GDP quarterly disaggregation algorithm

The core of the model simulation is based on the NSGA-II algorithm developed in [1], [4], [10] which is adapted to the problem of GDP quarterly disaggregation in [8]. The algorithm is a fast elitist multi-objective programming algorithm. We provide the bounded values (minimum and maximum) of the target variables in the NSGA-II algorithm. Finally, the pseudo code of the complete algorithm is given by the above steps:

Step 1: Setting the decision space upper bound and lower bound for the target variables.
Step 2: Initializing all general parameters
Step 3: Interpolation of accounts with quarterly indicators
Step 4: Initialization of the other NSGA-II input parameters number of constraints, number of variables, the size of initial population to be generated, the number of iteration, the maximum number of generations

Step 5: Calling the Model function
Step 6: Computing the NSGA-II algorithm
Step 7: Recovering of the optimal solution provided by NSGA
Step 8: Adjusting the optimal solution provided by NSGA-II
Step 9: Predicting if necessary the quarterly GDP for next quarters.

IV. SIMULATION, RESULTS AND DISCUSSION

Several tests were performed on the key parameters of the algorithm for evaluating the convergence speed. The key parameters as input parameters are: the population size (*pop_size*), the iteration number (*no_rum*), the maximum number of generations (*gen_max*) and the mutation distribution index in the polynomial mutation (*etam*).

Based on the findings in literature, several values were tested for the parameters *pop_size* as in [5], *no_rum*, *gen_max* and *etam*. The results of these tests have been presented in different figures of the Pareto-optimal front. The Pareto-optimal solutions obtained with the simulation are illustrated through Figure 2 and Figure 3.

The analysis of the results shows that if the parameter *no_run* is set to 1, the simulation gives many Pareto optimal solutions but the number of solutions increases with the population size (*pop_size*) and this regardless of the *gen_max* value, this situation is illustrated by the panels (a1) (b1), (c1) and (c2) of Figure 2. It was used pop_size=100 in the following simulations, as it was adopted in [3], [8], [9], [11]. The Pareto optimal front presented in the panels (a2), (b2), (d1) and (d2) indicate that the algorithm converges to a single Pareto optimal point when *no_run*=100 for any *gen_max* value with *pop_size*=100. However, the Pareto optimal front obtained with *gen_max*=25 and *gen_max*=100 have the same configuration; the solutions obtained for different cases are all located in practically the same restricted space (cuboid).

It should be noted that overall, the *gen_max* values set at 25 and 100 give Pareto front with the same characteristics; however, *gen_max*=100 takes relatively more time than when *gen_max*=25, for displaying results.

For the convergence speed testing by controlling the mutation distribution index, many values of *etam* was used with a fixed value or parameter triplet (*pop_size, no_rum, gen_max*). The Pareto front are illustrated in panels (e1)-(h2) of Figure 3. The Table 1 show how fast the algorithm converges toward the Pareto-optimal solutions and present the number of Pareto-optimal solution given by the algorithm. We test different values of the parameters *pop_size, no_rum, gen_max* combined with the parameter *etam. Our aim is to* evaluate the speed of the algorithm convergence and to observe the number of Pareto-optimal solution we get with each value of the parameters. As the results show in Table 1 and Figure 3, it is noted that for a fixed value of (*pop_size, no_rum, gen_max*), a change in *etam* value doesn't modify perceptibly the convergence speed (in term of number of seconds for computing) of the algorithm and the number of Pareto-optimal solution.

V. CONCLUSION

We use in this paper a computationally fast and elitist multi-objective evolutionary algorithm based on NSAG-II. On the test problem taken from [7]. NSGA-II is able to converge closer to a single fixed Pareto-optimal front with the GDP quarterly disaggregation problem. However, NSGA-II maintains diversity among solutions by controlling the mutation distribution index in the polynomial mutation.

In the convergence speed test by controlling both the mutation distribution index and the parameter triplet (pop_size, no_rum, gen_max), it is clear that the mutation distribution index alone doesn't influence the speed of algorithm convergence. However, we note that such a deterministic crowding coupled with the effect of mutation-based approach has been beneficial for convergence close to a fixed Pareto-optimal solution.

Finally, our conclusion is that, if the parameter triplet (pop_size, no_rum, gen_max) is judiciously chosen so that pop_size ≥ 100, no_rum > 1 and gen_max > 2, the fast and elitist NSGA-II algorithm converges to a fixed Pareto optimal solution located in a fixed cuboid for the test problem for any value of the mutation distribution index.

TABLE 1: SENSIBILITY OF MUTATION DISTRIBUTION INDEX ON THE CONVERGENCE AF THE GDP PROGRAM ALGORITHM

(*pop_size, no_rum, gen_max*)	**Mutation distribution index (etam)**	**Running Time (in second)**	**Number of Pareto-optimal solutions**
(10, 1, 25)	0	18.292	10
(10, 1, 25)	1	19.2736	8
(10, 1, 25)	10	19.6863	9
(10, 1, 25)	100	17.6145	9
(10, 1, 25)	500	18.4421	9
(10, 1, 25)	1000	19.0442	8
(10, 1, 25)	2000	17.6917	6
(10, 1, 25)	5000	18.6984	8
(100, 10, 25)	0	1656.67	1
(100, 10, 25)	100	1703.82	1
(100, 10, 25)	200	1733.95	1
(100, 10, 25)	500	1769.96	1
(100, 10, 25)	1000	1751.14	1

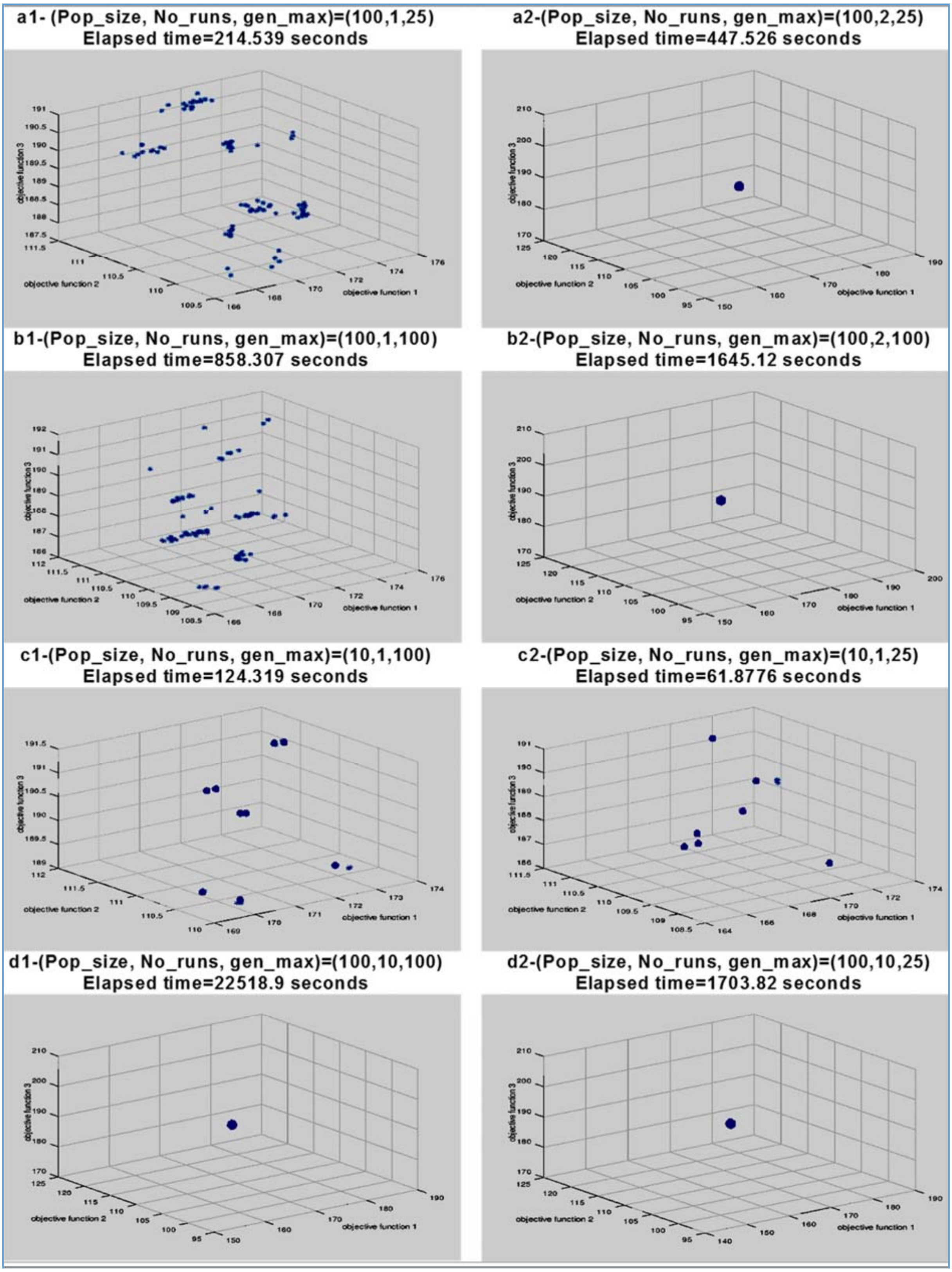

Figure 2. Optimal Pareto solutions for different simulation parameters (*pop_size, no_rum*, *gen_max*) with etam=100

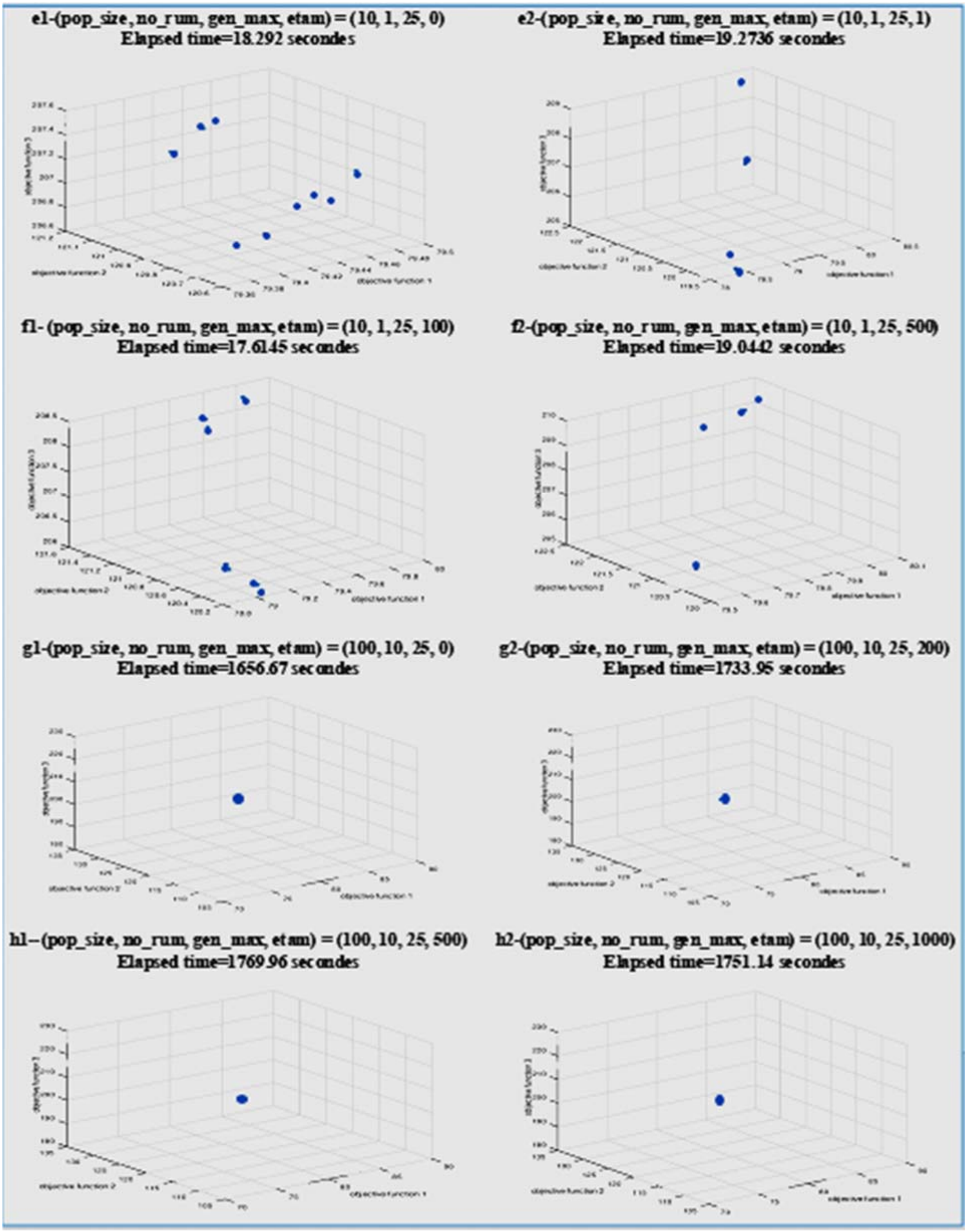

Figure 3. Various Optimal Pareto solutions for different simulation parameter etam values with afixed (*pop_size, no_rum, gen_max*)

ACKNOWLEDGMENT

All simulations are performed with Octave software installed on HP Intel Core i7 (vPro) notebook PCs, 16 Gb RAM, under Windows system.

We thank all the reviewers for their comment. We would like to send special thanks to the African Centre of Excellence in Mathematical Sciences and Application (ACE-MSA) at Institute of Mathematics and Physics Sciences (IMSP) for all its supports.

REFERENCES

[1] Aravind, S. : A Fast Elitist Multiobjective Genetic Algorithm: NSGA-II. Mathlab source code (2012). http://www. mathworks. com/matlabcentral/fileexchange/10429-nsga-iia-multi-objective-optimization-algorithm/content

[2] Barichard, V. : Approches hybrides pour les problèmes multiobjectif. Thèse de Doctorat Num. d'ordre : 593, école doctorale d'Angers, Laboratoire d'Etude et de Recherche en Informatique d'Angers, Univesite d'Angers, 1, 7, 22, 32-38 (2003)

[3] Belaidia, I., Mohammedib K., Brachemi, B. : Mise en œuvre de l'Algorithme NSGA-II pour l'Optimisation Multi-Objectif des paramètres de coupe en Fraisage en bout. 19ème Congrès Français de Mécanique, Marseille, 24-28 août 2009, 3-5 (2009) https://www.researchgate.net/publication/260553237_Mise_en_oeuvre_de_l'Algorithme_NSGAII_pour_l'Optimisation_Multi-Objectif_des_parametres_de_coupe_en_Fraisage_en_bout

[4] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T. : A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. IEEE Trans on Evol. Comp, Vol. 6, No. 2, april 2002, 183 - 188 (2002), https://www.iitk.ac.in/kangal/Deb_NSGA-II.pdf

[5] Hafid, Z. : Représentation de solution en optimisation continue, multi-objectif et applications. Thèse de Doctorat cotutelle, Ecole Mohammadia d'Ingenieurs, LERMA- Rabat-Maroc, Institut National des sciences appliquées de Rouen, HAL Id: tel-00939980,30-31, 50-51 (2013), https://tel.archives-ouvertes.fr/tel-00939980/document, , 31 Jan (2014)

[6] R. A. Essessinou, G. Degla and Babacar Mbaye Ndiaye: From Two to One Index Isomorphism in Optimization Program for Quarterly Disaggregation of Annual Times Series. Journal of Advances in Mathematics and Computer Science, 34(1&2): 1-15, 2019; Article no. JAMCS.51632, http://www.journaljamcs.com /index.php/JAMCS/article /view/30199

[7] R. A. Essessinou and G. Degla: A Proposed New Model for Denton Proportional Method Generalization in Quarterly Disaggregation of the Gross Domestic Product. Mathematics Letters, 5(4): 47-53 (2019), doi: 10.11648/j.ml.20190504.12. http://www.sciencepublishinggroup.com/j/ml

[8] R. A. Essessinou1 and G. Degla: Using a fast elitist non-dominated genetic algorithm on multiobjective programming for quarterly disaggregation of the Gross Domestic Product. 5th International Conference on Engineering and Formal Sciences, 24-25 January 2020, Brussels. European Journal of Engineering and Formal Sciences. DOI: 10.26417/ejef.v4i1.p24-45.

[9] R. A. Essessinou and G. Degla: An illustration of the convergence of NSGA-II algorithm to a single Pareto optimal solution with a big size continuous and quadratic multiobjective optimization, To appear in

[10] Robin C. P., Kalyanmoy D., Maszatul M. M., Sanaz M., Rui W. : A Review of Hybrid Evolutionary Multiple Criteria Decision Making Methods. COIN Report Number 2014005, 1-3, 6 (2014)., https://www.egr.msu.edu/~kdeb/papers/c2014005.pdf

[11] Srinivas N., Kalyanmoy D. : Multiobjective optimization using Nondominated Sorting Genetic Algorithm. Journal of Evo. Comput. Vol. 2, No. 3, Pages 221-248, 7-12 (1994), https://web.njit.edu/~horacio/Math451H/download/SrinivasDeb_GA.pdf.