On the Landscape of a Problem of Finding Satisfactory Metaheuristics

José M. Cecilia, Baldomero Imbernón
Bioinformatics and HPC Research Group (BIO-HPC)
Polytechnic School, Univ. Católica San Antonio of Murcia (UCAM)
Murcia, Spain
{jmcecilia,bimbernon}@ucam.edu

José-Matías Cutillas-Lozano, Domingo Giménez
Department of Computing and Systems
University of Murcia
Murcia, Spain
{josematias.cutillas,domingo}@um.es

***Abstract*—Fitness landscape analysis is used to determine the distribution and shape of the fitness function for the space of solutions of optimization problems. It is a hard task which can help in the design of efficient metaheuristics to approach the solution of the problem in hand. This paper studies the landscape of a problem of finding satisfactory metaheuristics. A molecule-docking problem and a problem of determination of kinetic constants in a chemical reaction are used as case studies. We are not working at the level of the case study problems, but at a higher abstraction level, which makes the landscape analysis an even harder task. The metaheuristics being analyzed are developed with a parameterized schema, with each combination of parameters in the schema being an element of the solutions space and a metaheuristic to be applied to the base problem. The fitness for a metaheuristic is measured by applying it to some training instance of that problem. Some highlights are given on how fitness landscape analysis can be conducted for this type of problems. The ideas analyzed with the two case studies can be extended for the development of hyperheuristics searching for satisfactory metaheuristics for other optimization problems.**

***Index Terms*—fitness landscape analysis, metaheuristics, hyperheuristics, molecule-docking, reaction kinetics**

I. Introduction

Metaheuristics are used to approach the solution of complex optimization problems, but the selection of a satisfactory metaheuristic for a particular problem is a difficult task. Fitness landscape analysis has been applied to the field of metaheuristics since the 1980s [1]–[3]. It studies the shape of the fitness in the space of solutions, and gives guidelines for the design of metaheuristics for the problem in hand.

Here we are interested in the landscape of an optimization problem built on top of another optimization problem. Two optimization problems are used at the bottom as case studies: a molecule-docking problem [4] and determining the kinetic constants of a chemical reaction [5]. Several metaheuristics can be applied, and a hyperheuristic (a metaheuristic in itself) searching in the space of metaheuristics is the optimization problem at the top of our hierarchy.

A parameterized schema of metaheuristics is used. It was initially used for the application of several metaheuristics to some optimization problems [6]. The next step led us to the development of hyperheuristics on top of the metaheuristic schema [7]. These hyperheuristics has high computational costs due to the repeated application of numerous metaheuristic configurations to the problem in hand. The base problem may in turn be computationally demanding, and so the combination of the high computational cost of metaheuristics with that of the hyperheuristics working on top of metaheuristics advises the exploitation of parallelism at both levels [8]. But even the efficient exploitation of parallelism is not enough to reduce the search to an acceptable time, and so fitness landscape analysis can guide the search for satisfactory metaheuristics. The conclusions drawn here can be applied to the problem of searching for satisfactory metaheuristics for other optimization problems.

The rest of the paper is organized as follows. Section II briefly describes the two case studies. The general ideas of the parameterized metaheuristic schema and the hyperheuristic developed on top of it are presented in Section III. The fitness landscape analysis is shown in Section IV. Finally, Section V summarizes the conclusions and gives some directions for future work.

II. Case Study Problems

A. The Molecule-Docking Problem (MDP)

Virtual Screening (VS) methods analyze large libraries of small molecules (*ligands*) to search for those structures which are most likely to bind to a protein receptor or enzyme [9]. VS is computationally demanding and requires the application of approximate optimization and high performance computing techniques [10], [11].

Molecular docking is used to predict non-covalent binding of molecules. Normally, the molecules to be bound are a macromolecule (receptor) and a small molecule (ligand). There are several scoring functions that can be used to measure the strength of the bond [12]. The Lennard-Jones potential is used here. The sum of the interactions of each atom of an active site of the protein with each atom of the ligand is computed. So, the computing cost is quadratic in the sizes of the molecules.

The ligand can bind at several spots on the receptor molecule. There are six degrees of freedom for the ligand, three for its translation and three for the rotation. The spot, translation and rotation which give the lowest value of the

Supported by the Spanish MINECO and European Commission FEDER funds, under grants TIN2015-66972-C5-3-R and TIN2016-78799-P (AEI/FEDER, UE).

scoring function are sought. The values of the movements and rotations of the ligand can be approached with metaheuristics, and a search is conducted in the neighborhood of each spot. So, independent metaheuristics are applied at each spot and the reference set for the search is composed of a subset for each spot.

B. Kinetic Constants of a Chemical Reaction (KCP)

When chemical reactions occur in heterogeneous phase the variables that affect the reaction rate are temperature, pressure, composition and mass transfer rate. The kinetics of the dissolution of calcium carbonate is used for the experiments. It is a function of the concentration of the various carbonate species in the solution and, therefore, a function of the partial pressure of carbon dioxide and pH. Depending on the value of the pH, there are different ways in which the dissolution occurs: by reaction with acetic acid, by reaction with carbonic acid, and by the hydrolysis reaction [13]. Therefore, the variation of moles of calcium over time is

$$\frac{1}{V}\frac{dN_{Ca^{2+}}}{dt} = -k_1 a^{n_1}\left[H_3O^+\right]^{n_2} - k_2 a^{n_3}\left[H_2CO_3\right]^{n_4} - k_3 \quad (1)$$

where k_i are the combined reaction rate constants, n_i are the reaction orders, and a is the area of the tablet, which is known for each increase of time. The integration of the equation for a certain value of time, t_j, gives the increase of calcium in the solution as a function of the concentrations of protons and carbonic acid. An individual is represented by a real vector of size seven. Every time the fitness of an individual is computed, the whole chemical system is solved with an iterative process, and the fitness is obtained by comparing the values obtained with the simulation with those obtained experimentally.

III. Parameterized Metaheuristics and Hyperheuristics

The application of a parameterized schema of metaheuristics (Algorithm 1) to the two case studies is considered. Schemas of this type have been applied to other problems, and the fitness landscape analysis techniques here used can be extended to other hard optimization problems.

Algorithm 1 Parameterized metaheuristic schema

```
Initialize(S,ParamIni) //Generate initial set and improve elements
while (not EndCondition(S,ParamEnd)) do
    SS=Select(S,ParamSel) //Select elements for combination
    SS1=Combine(SS,ParamCom) //Combine pairs of elements
    SS2=Improve(S,SS1,ParamImp) //Improve and diversify
    S=Include(S,SS1,SS2,ParamInc) //Update the reference set
end while
```

The schema comprises some basic functions which work on sets (S, SS, $SS1$ and $SS2$) of configurations. Some metaheuristic parameters ($ParamX$) are included in each basic function, which can be implemented in different ways, and the number of parameters and their meanings also change. Landscape analysis is conducted to study the influence of the parameters on the fitness. The sets, the basic functions and the meaning of the metaheuristic parameters in the schema are commented on.

`Initialize`: Sets of candidate solutions are randomly generated. For the MDP, a subset is generated at each spot, but in the experiments only one spot is considered for low execution times. The number of elements is given by the parameter Initial Number of Elements ($INEIni$), which determines if the metaheuristic is closer to a local search ($INEIni = 1$) or to a population based method.

Some of the generated elements are improved, and the intensity of the improvement can be varied to determine the importance of the local search in the metaheuristic. A percentage ($PEIIni$) of the elements is improved by comparing its fitness with that of a neighbor configuration, which substitutes the original element if the fitness is improved. The intensity of the improvement is determined by parameter $IIEIni$, which represents the maximum number of neighborhood analysis steps.

Another intensification parameter is considered for MDP. A ligand has a number of flexibility junctions determined by its crystallographic structure. In the intensification, $NEIFlex$ represents the number of configurations obtained by rotation of the junctions to be evaluated for each candidate element. This parameter depends on the characteristics of the ligand, and so the same value is considered for improvements in the functions of the schema (after initialization and in the improvements inside the loop).

A number of the best elements is selected for the following iteration. The implementations for the two problems differ slightly at this point. In the KCP only one parameter is used, for the Number of Best Elements to include ($NBEIni$). But in the MDP a Number of Best Elements ($NBEIni$) and of Worst Elements ($NWEIni$) is established, in order to not concentrate the search on only the most promising areas. The selection can be made in different ways (random, roulette, etc). Our implementation selects the $NBEIni$ best elements, and $NWEIni$ elements are randomly selected from the half elements with the worst scores.

`EndCondition`: Typically, the end condition of a metaheuristic is a maximum number of iterations ($MNIEnd$) or a maximum number of iterations without improving the best solution ($MIREnd$). Because we are interested in the study of the influence of the parameters on the fitness, the experiments are carried out for fixed times.

`Select`: A number of the best ($NBESel$) and the worst ($NWESel$) elements of the reference set are selected. The selection is made in the same way as the elements to be included in the reference set are selected for the MDP after the improvement in the initialization.

`Combine`: A number of combinations is carried out between best ($NBBCom$), worst ($NWWCom$) and best-worst ($NBWCom$) pairs of elements. The inclusion of worst elements diversifies the search. In the MDP the combination of a pair of elements gives a new element obtained as the mean of the translation and rotation parameters of the two elements. In the KCP, the classical crossing by a middle point is used.

`Improve`: Some of the elements obtained by combination are improved in the same way as in the improvement in the initialization. The percentage of elements to be improved is $PEIImp$, and the intensification of the improvement is determined by $IIEImp$.

A diversification strategy is applied to avoid falling into local minima. The percentage of elements from the reference subsets to be diversified is $PEDImp$, and the elements obtained by diversification are improved with an intensification $IIDImp$. An element is diversified by randomly generating a new value for one of its parameters, which is selected randomly. The elements generated by diversification are improved to avoid early death.

`Include`: The $NBEInc$ best elements from all those in the reference set and those generated by combination, diversification and improvement are included in the reference set for the next iteration. The reference set is completed with $FNEIni - NBEInc$ elements randomly selected from the remaining ones.

There are eighteen metaheuristic parameters in the implementation for the MDP and sixteen for the KCP, and their influence on the fitness was analyzed varying the execution time and without considering the end condition parameters.

Hyperheuristics are methods for the selection of satisfactory metaheuristics [14]. The schema in Algorithm 1 can be used for the development of hyperheuristics. The optimization problem now consists in selecting the values of the metaheuristic parameters which give the best fitness for some training problems. The hyperheuristic is therefore another metaheuristic that can in turn be developed as a parameterized metaheuristic with the parameterized schema. The general ideas of this type of hyperheuristics when applied for the two case studies are:

- The possible solutions for the hyperheuristic are vectors of metaheuristic parameters (metaheuristics). The sizes of the vectors depend on the implementation of the parameterized schema: 18 or 16 parameters for the MDP and the KCP, with only 16 or 14 if the end condition is a fixed execution time. The hyperheuristic searches in a predetermined search range for each parameter. For the experiments with the MDP, $INEIni$ varies between 20 and 200, and a range between 0 and 100 is established for the rest of the parameters; and with the KCP, $INEIni$ and $FNEIni$ are between 20 and 200, the intensification parameters between 0 and 50 and the rest of the parameters between 0 and 100.
- The fitness for each metaheuristic is obtained through its application to some training problems. Only one training problem can give problem-dependent results, and a large number of training problems means large execution times. Fitness landscape analysis can help in determining how a particular problem is representative of the behavior of other problems.
- The execution time of a hyperheuristic of this type is very high due to the repeated application of metaheuristics to the training instances. So, the analysis of the fitness landscape could help to guide the search of the hyperheuristic for low search times.
- In general, to reduce the high execution times, the metaheuristic parameters for the hyperheuristic will be smaller than those of the metaheuristics it searches for. The sets in the search are smaller, as are the number of elements to be improved and the intensification of the improvement.
- The improvements in the initialization and inside each iteration are carried out by analyzing elements in the neighborhood, with the neighbors of an element being those which are obtained from it by changing one element of the metaheuristic parameter vector (increasing or decreasing by one).
- The combination of a pair of metaheuristics is carried out with the typical method of selecting a crossing point and combining the initial part from one ascendant with the second part from the other. If an invalid configuration of the metaheuristic parameters is generated, it is discarded.
- The diversification consists of the substitution of the value of a randomly selected metaheuristic parameter by another value randomly selected from the search range for this parameter.

There is an optimization problem at the bottom (MDP or KCP), and the fitness landscape analysis of metaheuristics for this problem studies the shape of the fitness function for some instances of the problem. At a higher abstraction level, the hyperheuristic is now the metaheuristic to be guided with the results of the fitness landscape analysis.

IV. Fitness Landscape Analysis

The fitness landscape analysis of the problem of finding satisfactory metaheuristics for an optimization problem is a hard task. The fitness of each metaheuristic is measured through its application to various training instances of the problem. The optimization problem can have high computational costs, which, together with the number of training problems, means high execution times. Furthermore, the results obtained with one or several training problems should be representative for other problems. For simplification, for the two case studies experiments were carried out for three instances of the problem and for 100 randomly generated metaheuristics, with the values of the metaheuristic parameters in the ranges previously indicated.

The characteristics of the pairs for the MDP are given in Table I; they correspond to the Angiotensin-converting enzyme (ACE), the Glycogen phosphorylase beta (GPB) and the Poly(ADP-ribose) polymerase (PARP), and the table shows the number of atoms of the receptor and the ligand and the number of junctions of the ligand. The best fitness when applying each metaheuristic to the corresponding receptor-ligand pair was stored at intervals of 30 seconds, starting at 30 seconds and finishing at 600 seconds. Three experiments for which laboratory data are available are used for the KCP. In this case, each metaheuristic is run for 100 seconds, and the fitnesses obtained at intervals of 5 seconds are stored. The results of the experiments are analyzed to study three aspects:

TABLE I
Characteristics of the receptor-ligand pairs used in the experiments.

pair	#atoms receptor	#atoms ligand	#junctions
ACE	9198	59	13
GPB	13261	29	1
PARP	5588	32	3

- The influence of the parameters in the fitness is analyzed, both individually (which parameters influence the fitness most and if the influence is positive or negative) and globally (if the distance between metaheuristics is related to the difference of their fitnesses). The hyperheuristics could concentrate the search in the parameters which prove to be the most influential, and in the ranges where the best fitnesses are obtained.
- The influence of the execution time on the distribution of the fitness and on the influence of the parameters in the fitness. This study can help establish a time limit for the application of the metaheuristics to the training problem, so reducing training time.
- Whether the results obtained for one instance are similar to the others. If they are similar or if some relation between them can be deduced, the number of training problems could be low (in the best case only one), so reducing training time.

The influence of the metaheuristic parameters on the fitness can be analyzed to determine the parameters with most influence and the ranges where the search should be concentrated. The evolution of the correlation coefficient of the metaheuristic parameters with respect to the fitness is shown in Figures 1 (MDP) and 2 (KCP). A negative value indicates that when the value of the parameter increases the fitness improves (decreases). The figures on the left show the coefficient for the parameters with the greatest positive influence on the fitness, and those on the right correspond to the other parameters. The colors identify the basic functions in the schema: blue for initialization, red for selection, black for combination, brown for improvement, cyan for inclusion, and green for *NEIFlex*.

In the MDP, the parameter which most positively influences the fitness is not always the same. For ACE it is $NEIFlex$, which also influences PARP, but with no influence on GPB. It can be logical if we consider the number of flexible junctions in each case, which is 13 for ACE, 3 for PARP and 1 for GPB (Table I). So, it seems the search by flexing the ligand is more advisable for more flexible junctions. Another influential parameter is $NBEInc$, with a correlation coefficient of around -0.2 in ACE and PARP, and a little higher than -0.1 in GPB. It is also normal that considering a large number of promising elements will provide better fitness. The percentage of elements to diversify ($PEDImp$) is the best parameter for GPB and PARP, and the value is slightly higher than -0.2 in ACE. So, it seems that diversification is better than exploring the neighborhood of promising metaheuristics. Surprisingly, the combination of non promising elements ($NWWCom$) seems to be the fourth parameter in importance for good fitness, maybe because combining these elements avoids intensification in their neighborhood.

The relevance of the parameters changes for the KCP. The most relevant parameter is always clearly $IDEImp$, which has an inverse relation with the fitness, so it is advisable not to work too much on the improvement of elements after diversification. The other parameters have little influence. The initialization parameters have in general a positive influence and, in particular, the improvement parameters in the initialization. The improvement of the elements in the successive steps ($PEIImp$ and $IIEImp$) also has a positive influence. So, the local searches lead to better results for this problem.

In any case, independently of which parameters have most influence on the fitness, the analysis can serve to guide the hyperheuristic according to the problem it is being applied to and the particular implementation of the parameterized metaheuristic schema for that problem.

There is a need to reduce the training time of hyperheuristics. Figures 1 and 2 show the relation of the metaheuristic parameters with the fitness at several time-steps. After a time (around 250 and 50 seconds) the influence is stable. Furthermore, as can be seen in Figure 3, the correlation coefficient of the fitnesses obtained at one time step with respect to those obtained at the final time is higher than 0.9 after 250 seconds for the MDP and after 60 seconds for the KCP. With the application of the metaheuristics to a training problem we are not searching for optimum solutions for this particular problem, but for satisfactory metaheuristics for the general problem. So, the training time could be reduced to approximately half the time used in the experiments.

There are other possibilities to reduce the training time. For example, with the 100 metaheuristics experimented with in 20 time-steps, if the five metaheuristics with the worst fitness are discarded at each step, the total training time is halved. The difference in the behavior with the different problems and instances considered can be seen in Figure 4, where the mean fitness of the five metaheuristics discarded at each time step is shown. The similarity in the results for GPB is observed in its flat shape. For the MDP, the slope of the lines seems to be related to the number of flexible junctions (Table I) and the difficulty of approaching the optimum.

V. Conclusions and Future Work

This paper presents a first approach to the analysis of the fitness landscape for the problem of finding satisfactory metaheuristics for hard optimization problems. A molecule-docking problem and a problem of obtaining the values of kinetic constants are used as case studies, but the same methodology can be applied to the search for metaheuristics for other problems. The metaheuristics considered are implemented with a parameterized schema, which is used for the development of a hyperheuristic which searches in the space of metaheuristics. The fitness for each metaheuristic is obtained with its application to some training problems. The main difficulty in the fitness landscape analysis is the

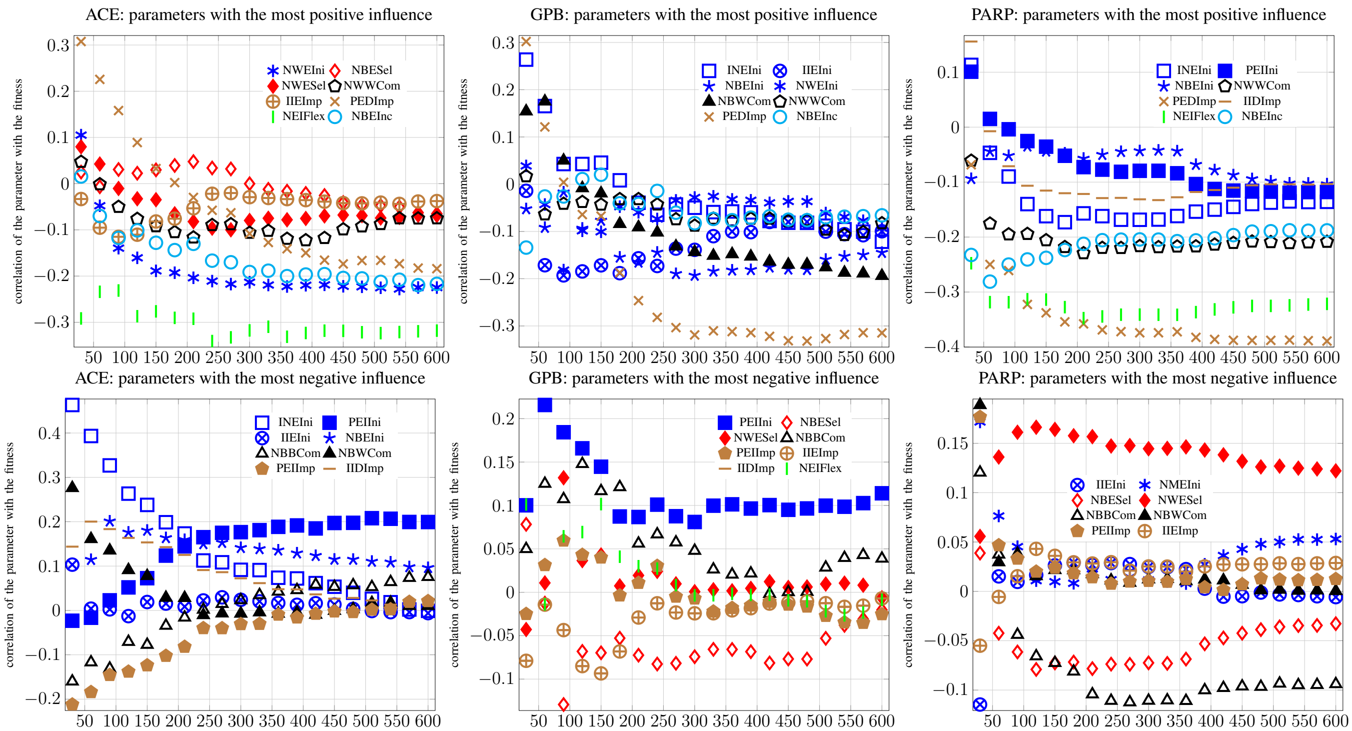

Fig. 1. Evolution of the correlation coefficient of the metaheuristic parameters with the fitness, for the MDP, for several execution times for the receptor-ligands ACE (left), GPB (middle) and PARP (right). Top: the eight parameters with the most direct relation with the fitness after 600 seconds; Bottom: the eight parameters with the most inverse relation.

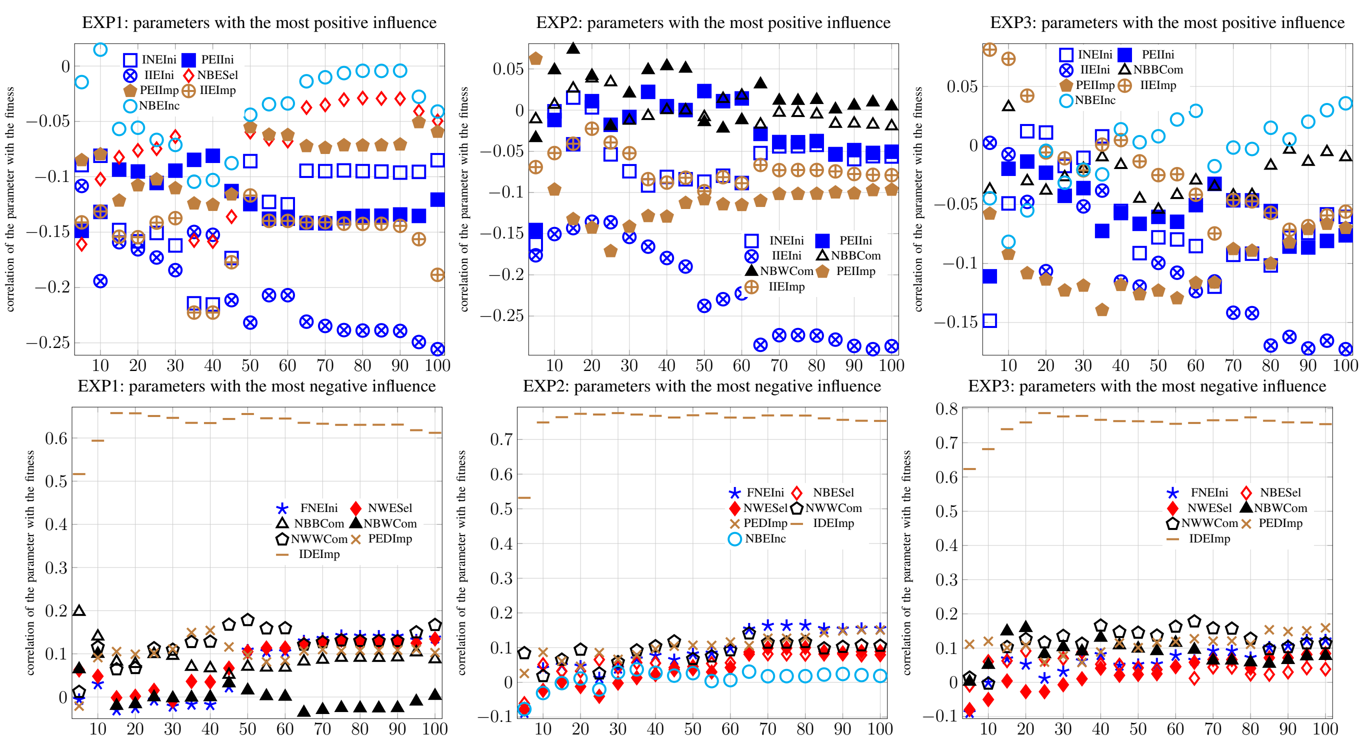

Fig. 2. Evolution of the correlation coefficient of the metaheuristic parameters with the fitness, for the KCP, for several execution times for the experiments EXP1 (left), EXP2 (middle) and EXP3 (right). Top: the seven parameters with the most direct relation with the fitness after 100 seconds; Bottom: the seven parameters with the most inverse relation.

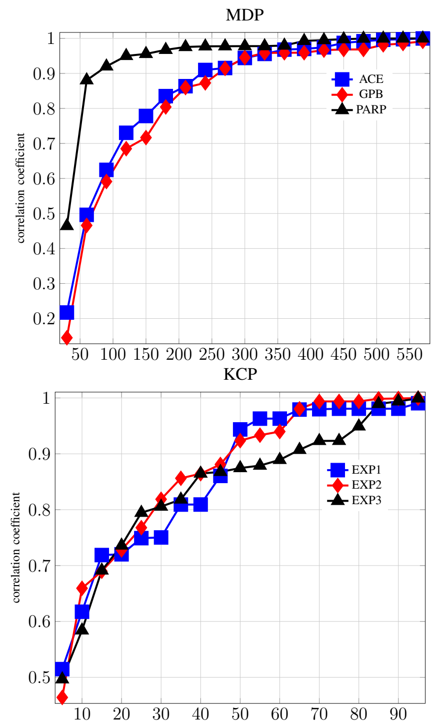

Fig. 3. Evolution of the correlation coefficient of the fitness at different time-steps with respect to the fitness. Top: MDP after 600 seconds; Bottom: KCP after 100 seconds.

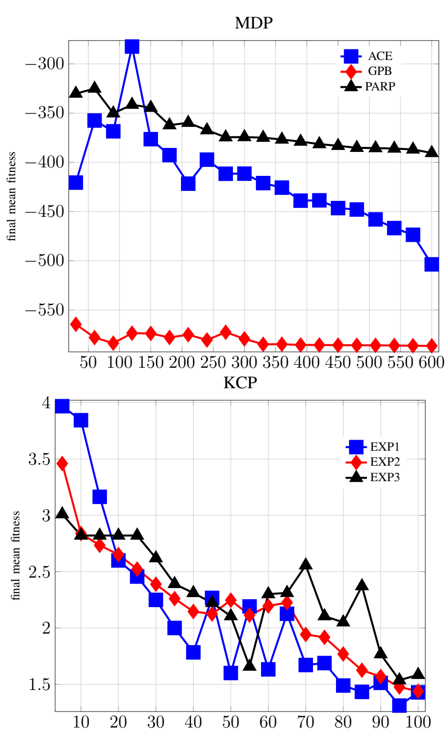

Fig. 4. Mean of the final fitness of the metaheuristics discarded at different times-steps. Top: MDP, plot each 30 seconds; Bottom: KCP, plot each 5 seconds.

randomness of the application of the metaheuristics, which makes the surface of the fitness function dependent on the random execution and on the instances used in the training process. The influence of the parameters and of the training time on the fitness has been experimentally analyzed.

The results of our analysis are being used to guide the implementation of hyperheuristics on top of the parameterized metaheuristic. More experiments with more instances of the case study problems are needed, and similar studies for other optimization problems at the bottom of the metaheuristics hierarchy will be conducted.

References

[1] W. Hordijk, “A measure of landscapes,” *Evol. Comput.*, vol. 4, pp. 335–360, Dec. 1996.

[2] K. Malan and A. P. Engelbrecht, “A survey of techniques for characterising fitness landscapes and some possible ways forward,” *Inf. Sci.*, vol. 241, pp. 148–163, 2013.

[3] P. F. Stadle, “Towards a theory of landscapes,” in *Complex Systems and Binary Networks*, vol. 461 of *Lecture Notes in Physics*, ch. 10, pp. 78–163, 2007.

[4] E. López-Camacho, M. J. García-Godoy, J. García-Nieto, A. J. Nebro, and J. F. A. Montes, “Solving molecular flexible docking problems with metaheuristics: A comparative study,” *Appl. Soft Comput.*, vol. 28, pp. 379–393, 2015.

[5] J. Cutillas-Lozano and D. Giménez, “Determination of the kinetic constants of a chemical reaction in heterogeneous phase using parameterized metaheuristics,” in *Proceedings of the International Conference on Computational Science*, pp. 787–796, 2013.

[6] F. Almeida, D. Giménez, J.-J. López-Espín, and M. Pérez-Pérez, “Parameterised schemes of metaheuristics: basic ideas and applications with Genetic algorithms, Scatter Search and GRASP,” *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 43, no. 3, pp. 570–586, 2013.

[7] J. Cutillas-Lozano, D. Giménez, and F. Almeida, “Hyperheuristics based on parametrized metaheuristic schemes,” in *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 361–368, 2015.

[8] J. M. Cecilia, J.-M. Cutillas-Lozano, D. Giménez, and B. Imbernón, “Exploiting multilevel parallelism on a many-core system for the application of hyperheuristics to a molecular docking problem,” *The Journal of Supercomputing*, Mar 2017.

[9] U. Rester, “From virtuality to reality-Virtual screening in lead discovery and lead optimization: a medicinal chemistry perspective,” *Current Opinion in Drug Discovery & Development*, vol. 11, no. 4, pp. 559–568, 2008.

[10] S. Garcia Sanchez, E. Ramirez Aportela, J. I. Garzón, P. Chacón, A. S. Montemayor, and R. Cabido, “FRODRUG: A virtual screening GPU accelerated approach for drug discovery,” in *22nd Euromicro International Conference on Parallel, Distributed, and Network-Based Processing*, pp. 594–600, 2014.

[11] B. Imbernón, J. M. Cecilia, and D. Giménez, “Enhancing metaheuristic-based virtual screening methods on massively parallel and heterogeneous systems,” in *Proceedings of the 7th International Workshop on Programming Models and Applications for Multicores and Manycores*, pp. 50–58, 2016.

[12] E. Yuriev, M. Agostino, and P. A. Ramsland, “Challenges and advances in computational docking: 2009 in review,” *Journal of Molecular Recognition*, vol. 24, no. 2, pp. 149–164, 2011.

[13] L. N. Plummer, T. M. L. Wigley, and D. L. Parkhurst, “The kinetics of calcite dissolution in CO_2–water systems at 5 to 60^{o}C and 0.0 to 1.0 atm CO_2,” *American Journal of Science*, vol. 278, pp. 179–216, 1978.

[14] E. K. Burke, M. Hyde, G. Kendall, G. Ochoa, E. Özcan, and J. Woodward, *A Classification of Hyper-heuristic Approaches*, pp. 449–468. Springer, 2010. In Michel Gendreau, Jean-Yves Potvin, editors, Handbook of Meta-heuristics.