Learning fitness function in a combinatorial optimization process

Frédéric Clerc[1][2], Ricco Rakotomalala[1], and David Farrusseng[2]

[1] Laboratoire ERIC – Université Lyon2
5 avenue Pierre Mendès France
69676 Bron CEDEX, France
(e-mail: ricco.rakotomalala@univ-lyon2.fr)

[2] Institut de Recherches sur la Catalyse – CRNS
2 avenue Albert Einstein
69626 Villeurbanne CEDEX, France
(e-mail: fclerc@catalyse.cnrs.fr, farrusseng@catalyse.cnrs.fr)

Abstract. Combinatorial optimization is a well known technique to solve problems in various fields such as jet engine design, factory and project scheduling or image recognition. Evolutionary computation and particularly genetic algorithms are commonly used to solve problems defined by complex and high dimensional mathematical expressions. Nevertheless, in some cases, domain experts cannot define this function exactly because of its complexity. In this paper we show that it is possible to solve such optimization problems, where the so called fitness function is unknown. To do this, we hybridize a classic genetic algorithm with a knowledge discovery system which extracts information from a database containing known observations allowing to build a model replacing the fitness function. We use the k nearest neighbours algorithm to solve such a problem sat in heterogeneous catalysis, a division of chemical science where a compound shall be optimized to favour a reaction.

Keywords: datamining, combinatorial optimization, genetic algorithm, fitness function.

1 Introduction

In drug design for medical applications as well as in catalyst development for oil refinery, the discovery and optimization of new formulations is based on the trial and error process. The state of knowledge in both biochemistry and solid state chemistry does not enable to build a model which would give guidelines for the design of formulations with targeted performance. In the vocabulary of optimization it means that the fitness function is not *a priori* known : each formulation must be first synthesized and then its performance measured with specific equipments. At the light of the results, chemists can draw new hypothesis and can design new formulations. A cycle is usually a day and years are required to end up with a final formulation. The new research methodology named high-throughput experimentation now enables to synthesize and test several dozen to hundreds of samples in parallel fashion in order to speed up the research process [B.Jandeleit *et al.*, 1998]. But now the question is : what are the experiments to be performed among an infinite possible number, which maximizes the chance of discovery and/or speeds up the optimization process? [Isar *and al.*, 2002], [Isar and Moga, 2004].

Computer assisted issues were recently reported to develop new catalysts. In [Wolf *et al.*, 2000], libraries of samples corresponding to populations are synthesized and tested in an iterative manner, using an evolutionary strategy. After typically 10 generations, the targeted compound presenting the best performance, the *optimum*, was found. Nevertheless, the total number of catalysts synthesized is still too high and shall be reduced. We present a system which enable to save experiments by hybridizing an optimization process with a knowledge discovery (KD) system. The concept was already reported in [Farrusseng *et al.*, 2003] and [Hanagandi and Kargupta, 1996]. The starting point consists in a real catalyst library which is synthesized and then tested. The corresponding information is stored in a database (DB) which is used by a KD algorithm to *estimate* new virtual individuals. The best estimated are *evaluated* (synthesized and tested) and the resulting information is added to DB so the prediction will be finer. This process is repeated until the checking of a given criterion. The creation of statistical models after each generation shall enable to direct the design of the libraries (i.e. population) by a virtual pre-screening.

In a first section we describe the hybrid optimization process, in a second section, the constraints and issues of the learning process are detailed. The experimental methodology and the results are presented in the third section, before concluding.

2 Hybridizing an optimization process with a knowledge discovery algorithm

Among several optimization processes such as tabu search [Laguna and Glover, 1998] and simulated annealing [S.Kirkpatrick *et al.*, 1983], we decided to use genetic algorithms (GA) [Holland, 1975],[Goldberg, 1989] as this technique was already known and used in the field of heterogeneous catalysis. The mechanics of a genetic algorithm are conceptually simple: (1) maintain a population of individuals (library or generation), (2) select the better for crossover operator, (3) perform mutation operator, and (4) use the offspring to replace poorer individuals. The hybridization consists at inserting a learning process in the genetic algorithm as described in Fig.1.

1. Initialization : Generating randomly a first population of n individuals which are potential solutions to the catalysis problem.
2. Evaluation : Giving a real value to each individual by synthesizing and testing the catalyst. The information produced is stored in DB. Each catalyst is defined by (1) a set of parameters and (2) its performance.

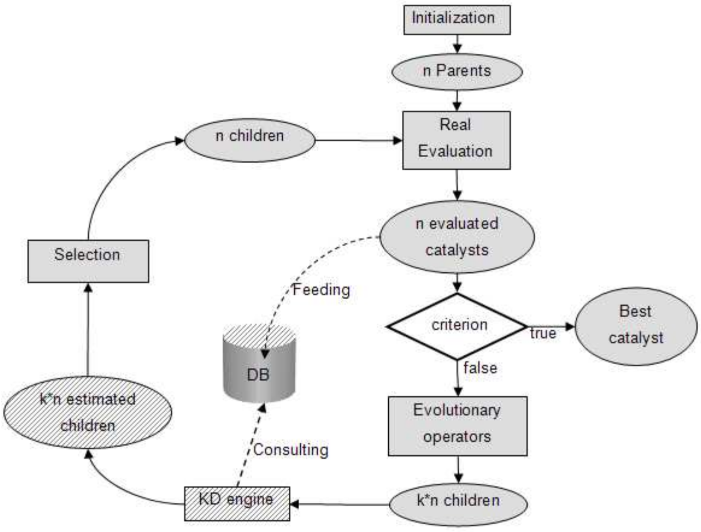


Fig. 1. Hybrid GA with KD system. The hatched bricks are the elements of the KD engine, the remainder are traditional elements of GA.

(number of loops*size of population) individuals are really evaluated and stored in DB.

3. Criterion : Stopping the evolution if verified. Usually, a number of loops is used.
4. Evolutionary operators : Applying traditional GA operators (crossover and mutation) on the parameters of the catalysts which has just been evaluated. During this operation, M*n virtual catalysts are generated in order to maximize the chances of obtaining the optimum quickly. This operation is costless as no individual is really synthesized.
5. KD engine : Mining the database DB so as to estimate the virtual library proposed by evolutionary operators. This quantitative prediction of the fitness involves the use of a supervised learning technique which is described in the next section. This *virtual screening* which is used as a first pass filter is the added value to classical GA.
6. Selection : Applying a selection which extracts n individuals among M*n from the virtual estimated ones. The two best are always picked up and the remaining (n-2) ones are selected using their rank. The selection and its role in genetic algorithms is complex and of importance. It is developed in [Miller and Goldberg, 1996].
7. Loop : Returning at step (2), individuals resulting from selection will be evaluated.

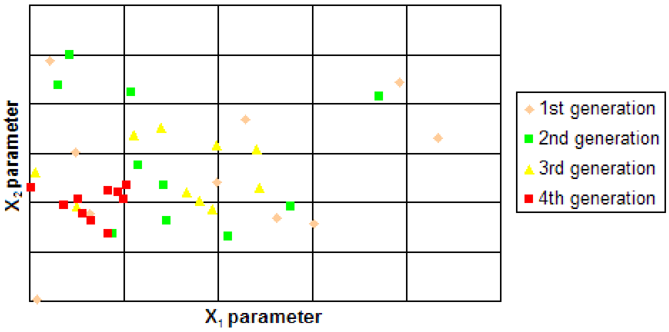


Fig. 2. Schematic evolution of database during a 4 generation process. From a generation to the other, the individuals get closer to the optimum. Thus, some zones of the space are well known, others are almost unknown. This training sample is not homogeneous and the algorithm must take this into account

3 The knowledge discovery algorithm

The knowledge discovery algorithm is integrated in an optimization process, so it needs to be adapted to this particular use. In the field of application the constraints are the following: (1) the search space is usually defined by 10 to 20 predictive continuous or categorical variables, (2) the representation space is non linear, (3) a maximum of 400 individuals can be screened and the less the better (4) the predicted variable is continuous. In addition, the learning algorithm has to face the issue of non homogeneous sampling of the search space. Indeed, because the optimization process focuses on specific zones of the search space (see Fig 2) the data set is usually biased.

Among various datamining algorithms [D.Hand *et al.*, 2001], we use the k nearest neighbours algorithm (k-nn)[D.W.Aha *et al.*, 1991]. To estimate a new individual, the algorithm searches among the known individuals (DB) its k nearest neighbours and attributes it their average performance. This algorithm fulfils the main requirements e.g. the learning is (1) adapted to overcome the problem of evolution and convergence as k-nn algorithm itself doesn't require complex update like neural networks or decision trees, (2) nonlinear.

4 Methodology and experiments

4.1 Benchmark

Because there is no open database in the field of catalysis, and because of the cost, the validation of optimization algorithms is performed through simulation using virtual benchmarks. We consider in this study the one presented in

$$\text{Catalyst} = (x_{V}, x_{Mg}, x_{B}, x_{Mo}, x_{La}, x_{Mn}, x_{Fe}, x_{Ga}, \text{method})$$

$$\text{With } x_{element} \in [0..1] \text{ and method} \in \{0,1\}$$

$$Y_{catalyst} = \begin{cases} X_1S_1, & \text{if method=0} \\ X_2S_2, & \text{if method=1} \\ 0, & \text{if } (x_{La} > 0) \text{ or } (x_B > 0) \end{cases}$$

$$S_1 = 66x_V \,.\, xMg\,(1 - x_V - x_{Mg}) + 2x_{Mo} - 0.1x_{Mn} - 0.1x_{Fe}$$

$$X_1 = 66x_V \,.\, x_{Mg}(1 - x_V - x_{Mg}) - 0.1x_{Mo} + 1.5x_{Mn} + 1.5x_{Fe}$$

$$S_2 = 60x_V \,.x_{Mg}\,(1 - 1.3x_V - x_{Mg})$$

$$X_2 = 60x_V \,.x_{Mg}\,(1 - 1.3x_V - x_{Mg})$$

Fig. 3. The virtual benchmark

[Wolf *et al.*, 2000]. It is composed of 9 predictive variables : 8 percentages of elements for the composition of the catalyst (V, Mg, B, Mo, La, Mn, Fe and Ga) represented by continuous variables from zero to one and a preparation method (coprecipitation or impregnation) represented by a discrete variable (0 or 1). The performance, named Y for Yield, is the continuous variable to predict and is defined in Fig.3. The optimum is a compound containing 32% of Vanadium, 32% of Magnesium and 36% of Molybdenum, the method being coprecipitation. According to the benchmark, we calculate that Y(0.32 , 0.32 , 0 , 0.36 , 0 , 0 , 0, 0) = 7.55

4.2 Conditions

We hybridize a very simple and classical GA for two major reasons. First, the application of computer based optimization methods in the field of heterogeneous catalysis is something quite new and before examining complex issues, we have to experiment the simple ones. Second, in this paper, we're aiming at measuring the performance of an hybrid GA and specially the KD algorithm. This GA, used to generate relevant new virtual individuals, uses a rank selection associated with an elitist selection (2 best individuals kept), a 3 point crossover (probability = 0.8) and a bit-flip mutation (probability = 0.01). Furthermore, the value of the multiplier M is arbitrarily fixed at 15.

In real conditions, the evaluation of a single catalyst is very costly so we have a strong constraint to respect. We consider the optimization finished at the end of 10 generations of 40 individuals, meaning 400 individuals evaluated during the whole experiment. This constitutes the stopping criterion we used. We call one optimization experiment a *run*. GA being stochastic processes only an average value is significative. Thus all the following results are based on 30 runs.

The behaviour of the k-nn algorithm is compared to the behaviour of trivial learning algorithms, the "learning limits" : no and perfect estimation. The upper limit is the perfect learning : the real value of the individual. The

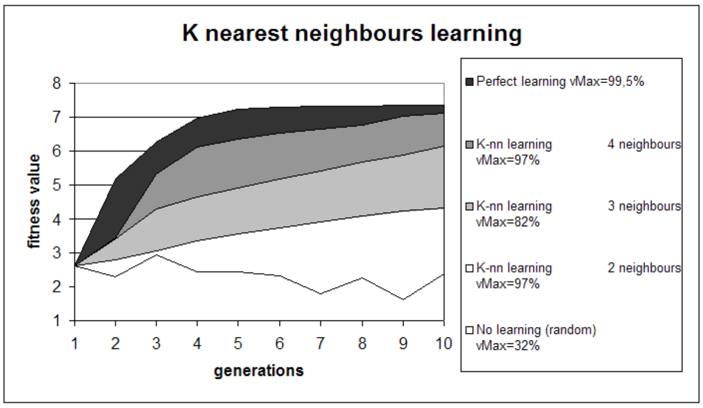


Fig. 4. Evolutionary behaviour of a hybridized GA with a k-nn algorithm, $k \in \{2, 3, 4\}$. Comparison with the learning limits. Note: the starting value of each curve depends on the benchmark and on the size of the population. Here, its value is 2.6 ± 0.5

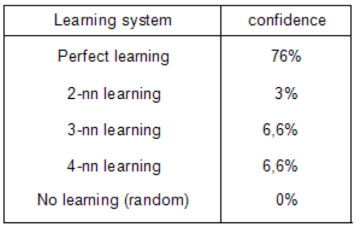
lower limit is the absence of learning, a random value with respect to the range of the benchmark (from 0 to 7.5).

4.3 Results

The quality of the algorithm is assessed by 2 criteria. First, the *performance* (vMax) is the average maximum value reached at the tenth generation. It is a percentage of the real optimum, for instance vMax = 50% means $7.5/2 = 3.75$. The Fig.4 presents the results of a hybrid for various k values. Whatever it is, the performance is manifestly better than using no learning. We expect that the upper bound is unreachable.

Second, the reliability of each algorithm is computed. Indeed a stochastic algorithm presents a different behaviour from one run to another. The *confidence* (conf) illustrates the percentage of runs where at least 98% of the optimum is really obtained on the whole the 30 runs. For instance, if 3 runs out of 30 reached at least 7.4 (98% of the optimum) then conf = 10%. The Fig.5 summarizes the values of this indicator according to the learning algorithm. The hybridization of a GA with no learning never reaches the optimum. In the opposite, the perfect learning fully benefits from the multiplication mechanism, the optimum is reached 23 times out of 30 at the tenth. Our proposal using k nearest neighbours occupies an intermediate position, whatever the value of k.

The results are in average better than no learning, either in terms of performance or in terms of confidence and the use of 4 neighbours gives the best results considering both criteria. In a real experiment, we would favour the confidence because the cost of a catalyst would not allow failure. The use



Learning system	confidence
Perfect learning	76%
2-nn learning	3%
3-nn learning	6,6%
4-nn learning	6,6%
No learning (random)	0%

Fig. 5. Percentage of runs which reaches at least 98% of the global optimum. Confidence of the hybrid GA/KD algorithm.

of a KD system makes it possible to improve the behaviour of a simple optimization algorithm, without increasing the global cost of experimentation. It makes it possible to choose in a relevant way among multiplied virtual individuals those which present truly good real performances.

5 Conclusion

We empirically studied in this article the hybridization of a genetic algorithm with a knowledge discovery system and its application to a heterogeneous catalysis problem whose fitness function is unknown. Its objective is to estimate the value of a potential solution to a problem which is not defined by a mathematical expression but by a set of observations, each of high monetary cost. We compare the results obtained by hybridizing a genetic algorithm with (1) a learning process using k nearest neighbours algorithm, (2) a perfect learning and (3) no learning. We show that the use of k-nn increases the optimization speed and improves the robustness compared to random learning.

There remains opened interrogations concerning the role of the number of neighbours. Increasing k value means that the k-nn algorithm is more linear and so the hybrid GA/KD would become less efficient for this application, but this remains to be demonstrated. Another question concerns the population multiplier, one expects that the higher, the more the chances to gain the optimum quickly are large. But however, this postulate is limited by the learning process. A ceiling value probably exists giving the best results possible for each KD algorithm.

Combinatorial catalysis is a vast field of investigation for applying new types of computer based optimizations and knowledge discovery systems. The actors of the domain are currently acquiring and storing data in vast databases. Combinatorial optimization methods are in total adequacy with experts needs and the expansion of such techniques is ensured. For this kind of hybrids, we are particularly interested in knowledge discovery methods which extract association rules. Indeed, the interactions between the predictive variables are often badly known and their description would be of a great interest. This constitutes the Knowledge Discovery in Genetic Algorithms (KDGA) project, materialized by a self made free software : OptiCat [IRC and ERIC, 2005] which has been used to perform all experiments presented here.

References

[B.Jandeleit *et al.*, 1998]B.Jandeleit, D.J.Schaefer, T.S.Powers, H.W.Turner, and W.H.Weinberg. Combinatorial materials science and catalysis. *Angew. Chem*, pages 2494,2532, 1998.

[D.Hand *et al.*, 2001]D.Hand, H.Mannila, and P.Smyth. *Principles of Data Mining.* Massachusetts Institute of Technology, 2001.

[Duff *et al.*, 2002]DG Duff, A Ohrenberg, S Voelkening, and M Boll. A screening workflow for synthesis and testing of 10,000 heterogeneous catalysts per day – lessons learned. *Macromolecular Rapid Communications*, pages 169–177, 2002.

[D.W.Aha *et al.*, 1991]D.W.Aha, D.Kibler, and M.K.Albert. *Instance-based learning algorithms.* Machine Learning, 1991.

[Farrusseng *et al.*, 2003]D Farrusseng, L Baumes, and C Mirodatos. In high-throughput analysis: A tool for combinatorial materials science. *Potyrailo*, pages 551–579, 2003.

[Goldberg, 1989]DE Goldberg. *Genetic Algorithms in Search, Optimization and machine learning.* Addison- Wesley, 1989.

[Hanagandi and Kargupta, 1996]V Hanagandi and H Kargupta. Unconstrained blackbox optimization: The search perspective. *Institute for Operations Research and the Management Sciences (INFORMS)*, 1996.

[Holland, 1975]J Holland. Adaptation in natural and artificial systems. 1975.

[IRC and ERIC, 2005]IRC and ERIC. `http://eric.univ-lyon2.fr/~fclerc/` OptiCat, 2005.

[Klanner *et al.*, 2003]C Klanner, D Farrusseng, L Baumes, C Mirodatos, and F Schuth. How to design diverse libraries of solid catalysts? *QSAR & Combinatorial Science*, 2003.

[Laguna and Glover, 1998]M Laguna and F Glover. *Handbook of Combinatorial Optimization.* Kluwer, Colorado Business Review, 1998.

[Miller and Goldberg, 1996]BL Miller and DE Goldberg. Genetic algorithms, selection schemes, and the varying effects of noise. *Evolutionary Computation*, 1996.

[S.Kirkpatrick *et al.*, 1983]S.Kirkpatrick, C.D.Gelatt, and M.P.Vecchi. Optimization by simulated annealing. *Science*, 1983.

[Wolf *et al.*, 2000]D Wolf, OV Buyevskaya, and M Baerns. An evolutionary approach in the combinatorial selection and optimization of catalytic materials. *Applied Catalysis A: General*, pages 63–77, 2000.