PRES LUNAM
Ecole Doctorale STIM
Sciences et Technologies de l'Information et Mathématiques

Field : Informatique
Laboratory : LERIA
Team : Metaheuristics, Optimization and Applications

A Dynamic Island Model for Adaptive Operator Selection

Candan, Caner
Email : caner.candan@univ-angers.fr

Abstract: In this paper we propose a generic framework for Dynamic Island Models, which can be used as an original approach for the adaptive selection of operators in evolutionary algorithms. Assigning a variation operator to each island, we show that the dynamic regulation of migrations, which takes into account the pertinence of recent migrations, distributes the individuals on the most promising islands, i.e., the most efficient operators, at each stage of the search. The efficiency of this approach is assessed with a few problems of the literature by comparing theoretical expected results to those obtained by our dynamic island model. Experiments show that the model provides the expected behavior.

Keywords: *Island Models, Adaptive Operator Selection, Evolutionary Computation*
Cooperations: Computer Science Laboratory of the Faculty of Sciences of the l'University of Angers.

1 Evolutionary algorithms

Evolutionary algorithms (EA) have been widely used for tackling NP-hard problems [1]. Basically, an EA manages a population of individuals encoding possible configuration of the problem, in order to optimize a given fitness function. These individuals evolve by means of variations operators and selection processes. Although the efficiency of EAs is well-established on numerous optimization problems, their performance and robustness may depend on the correct setting of its components. Moreover, these technical choices often lead to the design of *ad hoc* EAs, dedicated to specific problem instances, and become out of the scope of non-specialist users. Therefore, parameter setting in EA has deserved much attention during recent years [2] in order to provide more generic and adaptive algorithms. Parameter setting may indeed be considered from two complementary points of view: the design of a suitable EA for a given problem and the improvement of the behavior of an EA to reach an acceptable performance. While the design will focus on the structural parameters of the EA such as its variation operators, its behavioral parameters, such as the application probabilities of these operators, will be adjusted to improve its solving efficiency. This efficiency greatly depends on the management of the balance between exploration and exploitation of the search space, which relies on the correct combination of suitable components with suitable behavioral parameters values. The contribution of this paper is thus to use Islands Models in order to achieve such an adaptive operators management for EAs.

2 Island Models

Island Models [3] consider simultaneously a set of populations, clustered on islands, which are evolving independently during some search steps and interacting periodically. This model, which constitutes an additional abstraction level in comparison to classical EAs, provides an improved management of the diversity and simplify the parallel implementation of EAs.

Most of the time, island models are used in a static way, where individuals are migrating from populations to populations following a fixed predefined scheme [4], or are specifically chosen in order to reinforce the populations characteristics [5]. Nevertheless, it is possible to dynamically regulate migrations between islands by considering a transition matrix [6]. Such a model is used to increase or decrease the migration probabilities during the evolutionary process according to the impact of previous analogue migrations. The purpose is to control migration in order to dynamically regulate the diversity of the individuals in the populations, according to the search progress, and, consequently, to control the population sizes. In classical uniform island models, each island uses the same EA and differs only by its individuals. Considering now a different algorithm on each island, a dynamic model allows to regulate interactions between individuals or groups of individuals.

3 Generic Dynamic Island Model Representation

Since twenty years and the first distributed evolutionary algorithms [7], island-based genetic algorithms (or island models [3]) are more and more studied in evolutionary computation. The main original problem consists in defining the model topology and the migration policies in order to reduce premature convergence of the population and to insure a global sharing of promising individuals. Numerous migration policies and model topologies have been proposed (see [8, 9]), and it is not obvious to figure out which topology and policies are the most suitable for a given purpose.

Let us so consider a previously proposed island model, which dynamically supervises the commonly-used EA parameters [10] such as population size, migration policy for the individuals, selection policy for immigrants or the topology of the communication between populations. A n-island model topology can be represented by a complete labeled digraph (see example in Figure 1). Migration policies are given by a transition (stochastic) matrix T of size n, where $T(i, j)$ represents the probability for an individual to migrate from island i to island j (or to stay on the same island if $i = j$). In order to make the model (possibly) independent to any global processing, T is actually given by n vectors that represent the migration policies for each island $V^i, i \in \{1, \cdots, n\}$, with $\Sigma_{j=1}^{n} V_j^i = 1$.

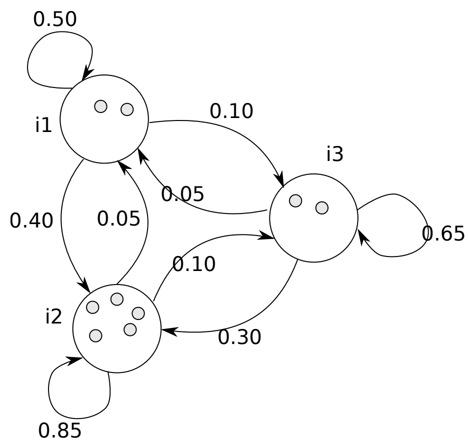

Figure 1: Island Model as a Complete Digraph

In this dynamic adaptation process, one has to determine pertinent migration probabilities at each step of the search process, considering a classical multi-population based EA. The dynamic regulation of the migration policies may result in different population sizes for the different islands, which prevent from assigning too much computational resources to poor-quality sub-populations or islands. However, if different islands use different variation operators, the control process should dynamically provide a well-balanced distribution of the individuals according to these operators and the current search state. As mentioned above, this can obviously be viewed as an adaptive operator selection process, which will be further detailed in Section 3.3.

3.1 Island processes

Figure 2 describes the mechanism that we propose for a generic Dynamic Island Models (DIM).

The vertical reading highlights the three divided layers defined as follows:

- upstream: the input part of the DIM with the received feedback data and individuals followed by a learning process;
- midstream: the evolution process (basic EA on the island) that is applied on the sub-population given as parameter;
- downstream: the output part of the DIM that uses intermediate functions (analysis of the sub-population and transition vector updating) before distributing the data to the other islands.

The left-side of the figure shows the feedback data flow processes while the right-side shows the individuals flow processes.

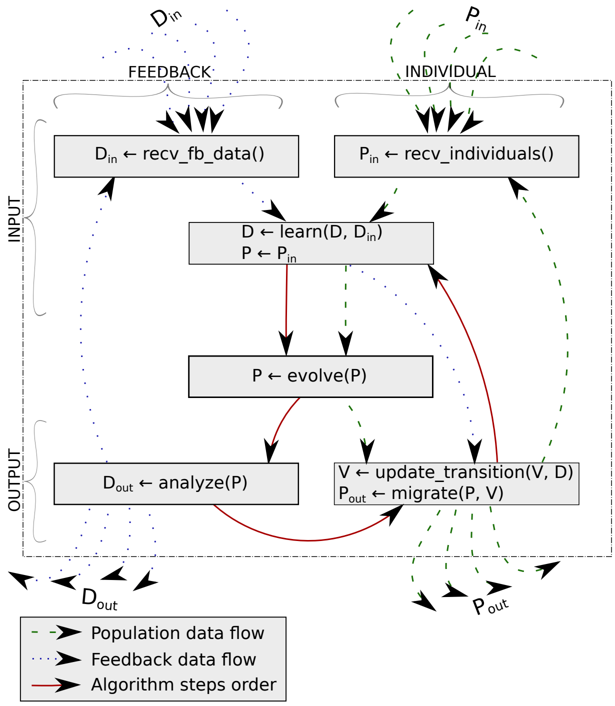

Figure 2: Dynamic island mechanism: population flow, feedback data flow and order of the basic steps.

Note that this DIM overall process is iterated on each island into a loop. The different blocks are ordered according to the arrows: learn, evolve, analyze and migrate. Each block is now described in Section 3.2.

3.2 Blocks description

$D_{in} \leftarrow$ **recv_fb_data():** this continuously running process collects the feedback data from other islands.

$P_{in} \leftarrow$ **recv_individuals():** this continuously running process collects individuals migrating to (or staying on) the current island.

Both `recv_fb_data()` and `recv_individuals()` functions do not cause any break during the execution of the main loop process since they should be processed in parallel. They fill out their own stacks each time they receive feedback data as well as new migrated individuals from other islands. An equivalent process can be simulated sequentially.

$D \leftarrow$ **learn(D, D_{in}):** the current knowledge vector D (n values representing the comparative pertinence of the last migrations, for all islands) is updated using data received since its last update that are recorded in D_{in}. Note that D_{in} may not have a fixed size.

$P \leftarrow$ **evolve(P):** a basic EA process is applied to the sub-population P.

$D_{out} \leftarrow$ **analyze(P):** according to the origin of individuals, feedback information characterizing the pertinence of last migrations are sent to corresponding islands.

$V \leftarrow$ **update_transition(V, D):** reward/penalty values given by D are used to update V with the following reinforcement learning process:

$$V = (1-\beta)(\alpha.V + (1-\alpha).D) + \beta.N$$

where N is a stochastic vector ($\Sigma_{j=1}^{n} N_j = 1$) with random values. α represents the importance of the knowledge accumulated during the last migrations (inertia or exploitation) and β is the amount of noise, which is necessary to explore alternative search space areas by means of individuals.

$P_{out} \leftarrow$ **migrate(P, V):** each individual migrates to an island regarding to V (in this representation, the particular case of staying on the same island is also considered as a migration). P_{out} is a vector of n sets of individuals.

3.3 DIM for Adaptive Operators Selection

The generic DIM introduced in this section (3) can easily be instantiated in order to manage classical evolutionary algorithms like genetic, memetics or population-based local search algorithms. As mentioned in introduction, DIM can be used for the adaptive operator selection in an EA. This can be achieved with a specific instantiation of the model proposed in Figure 2. Each island represents a particular operator and has the following specifications:

- The *evolve* process uses the operator assigned to this island.
- The *analyze* process computes the feedback information and sends it to each islands (including its own). In our case, this information will be the average improvement of all individuals in function of their previous localization, during the last evolution step.
- The *learn* process receives feedback information from all other islands. D is a *reward vector* computed using an *intensification* strategy: only the best island is rewarded.

4 Conclusion

This paper presents an original generic model for dynamic islands model and shows that this may constitute an efficient approach for the adaptive selection of operators in EAs. Each island is assigned to a single variation operator EA, and the dynamic regulation of migrations distributes the individuals on the most promising islands according to recent background information. At each stage of the search, the more efficient operators receive the greatest part of the computational resources, while the model is able to auto-adapt the attractive power of each islands. To assess the real efficiency of the model, we used an experimental protocol, comparing a theoretically optimal selection scheme of operators, for the One-Max problem, to empirical obtained values (see [11]). The next step is to apply this operator selection strategy to more difficult problems and to compare it with other adaptive operator selection methods.

References

[1] Zbigniew Michalewicz. *Genetic algorithms + data structures = evolution programs (3rd ed.)*. Springer-Verlag, London, UK, 1996.

[2] Fernando G. Lobo, Cláudio F. Lima, and Zbigniew Michalewicz, editors. *Parameter Setting in Evolutionary Algorithms*, volume 54 of *Studies in Computational Intelligence*. Springer, 2007.

[3] Darrell Whitley, Soraya Rana, and Robert B. Heckendorn. The island model genetic algorithm: On separability, population size and convergence. *Journal of Computing and Information Technology*, 7:33–47, 1998.

[4] Marek Rucinski, Dario Izzo, and Francesco Biscani. On the impact of the migration topology on the island model. *CoRR*, abs/1004.4541, 2010.

[5] Zbigniew Skolicki and Kenneth A. De Jong. The influence of migration sizes and intervals on island models. In *GECCO*, pages 1295–1302, 2005.

[6] Frédéric Lardeux and Adrien Goëffon. A dynamic island-based genetic algorithms framework. In *SEAL*, pages 156–165, 2010.

[7] Reiko Tanese. Distributed genetic algorithms. In *Proceedings of the third international conference on Genetic algorithms*, pages 434–439, San Francisco, CA, USA, 1989. Morgan Kaufmann Publishers Inc.

[8] Erick Cantú-Paz. Migration policies, selection pressure, and parallel evolutionary algorithms. *Journal of Heuristics*, 7(4):311–334, 2001.

[9] A. Coelho, I. Ricarte, A. Yamakami, E. Noda, and A. Freitas. Devising adaptive migration policies for cooperative distributed genetic algorithms. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, pages 438–443, 2002.

[10] Lourdes Araujo, Juan Julián Merelo Guervós, Antonio Mora, and Carlos Cotta. Genotypic differences and migration policies in an island model. In *GECCO*, pages 1331–1338, 2009.

[11] Caner Candan, Adrien Goëffon, Frédéric Lardeux, and Frédéric Saubion. A dynamic island model for adaptive operator selection. In *Genetic and Evolutionary Computation Conference (GECCO'12)*, pages 1253–1260, 2012.