Enhancing Distributed EAs using Proactivity

[Extended Abstract]

Carolina Salto
Facultad de Intenieriía
Universidad Nacional de La Pampa - CONICET
La Pampa, Argentine
saltoc@ing.unlpam.edu.ar

Francisco Luna
Departamento de Informática
Universidad Carlos III de Madrid
Leganés, Madrid, Spain
f una@inf.uc3m.es

Enrique Alba
Departamento de Lenguajes y Ciencias de la Computación
Universidad de Málaga
Málaga, Spain
eat@lcc.uma.es

ABSTRACT

In this abstract we describe a proactive strategy followed by a distributed evolutionary algorithm to adapt its migration policy. The proactive decision is made locally within each subpopulation, ant it is based on the entropy of that subpopulation. In that way, each subpopulation can ask for more/less frequent migrations from its neighbors in order to maintain the genetic diversity at a desired level, thus avoiding the subpopulations to get trapped into local minima. We conduct computational experiments on a set of different problems and it is shown that our proactive approach outperforms classical dEA settings by reaching accurate solutions in a lower number of generations.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search

General Terms

Theory, Algorithms

Keywords

distributed evolutionary algorithms, heterogeneity, proactive algorithms

1. INTRODUCTION

The working principles of a distributed EA (dEAs) include a communication phase, which is governed by a migration policy. The migration policy determines how communication is carried out by the islands of the dGA, and its is defined by the migration period, migration rate, selection/replacement of migrants, and topology. According to the results presented by Tanese [4], performance degrades if migration happens too frequently or too infrequently, such that the frequency of migration is a critical parameter for dEAs. It also occurs, as stated by [1], that the best parameter setting of an algorithm is different depending on the stage of the evolutionary process. These two factors leads to engineering algorithms whose parametrization is automatically modified according to an *intelligent* adaptive strategy.

GECCO'13 Companion, July 6–10, 2013, Amsterdam, The Netherlands.
ACM 978-1-4503-1964-5/13/07.

Our goal here is to adaptively control the migration policy of a dEA using the novel idea of the algorithm being proactive based on entropy information [3] of each subpopulation. In order to isolate their effect and better analyze the results, only one single migration parameter is under study: the migration period (mig_period) at which the individuals are exchanged among subpopulations. Whenever the entropy falls below a given threshold, these subpopulations proactively look for new genetic material by reducing the migration period of the subpopulation (more frequent migrations) that provides this island with more migrants.

In the following the structure of the new proactive dEA is presented and brief description of the results reached is further explained.

2. PROACT: A PROACTIVE DEA

The working of our proactive strategy, called PROACT, is as follows. Let $H(g)_i$ be the entropy value of subpopulation p_i at generation g. When the entropy value is close to 1.0 in a given island, we assume that the island has a good and diverse genetic material, so therefore the search has to be further intensified (increase exploitation with less frequent incoming individuals). On the other hand, the strategy tries to promote the exploration (more frequent migrations) when $H(g)_i$ is close to zero. In consequence, when p_i detects a decreasing diversity (low value of $H(g)_i$), it asks p_{i-1} to send individuals with higher frequency by updating the mig_period at p_{i-1}. That is, p_i receives new genetic material proactively by taking into account its actual needs. In this way, p_i acts in advance of losing diversity (proactive behavior), in other words, it anticipates the loss of diversity and changes its incoming flow of migrants.

Our proactive scheme uses an upper and a lower bound of $H(g)_i$, $\overline{H}$ and $\underline{H}$ respectively, in order to modify the migration frequency. Therefore, if $H(g)_i > \overline{H}$, PROACT decreases the migration period value in a value equal to the population size (μ value); analogously, if $H(g)_i < \underline{H}$, PROACT increases the period in μ units. Finally, if $\overline{H} \leq H(g)_i \leq \overline{H}$, we assume the search has a controlled entropy and, consequently, the migration period remains without modification. Algorithm 1 sketches the proactive strategy followed by each proactive subpopulation dEA_i. Of course, the migration periods are assumed to be discrete values in the range $[mig\_period_{min}, mig\_period_{max}]$. Thus, PROACT can directly measure and control the migration period.

PROACT uses an unidirectional ring topology so that each subpopulation p_i only receives/sends individuals from/to

Procedure 1 Proactive strategy

if $(H(g)_i > \overline{H})$ **then**
  $mig\_period = mig\_period - \mu$;
else
  if $(H(g)_i < \underline{H})$ **then**
    $mig\_period = mig\_period + \mu$;
  end if
end if

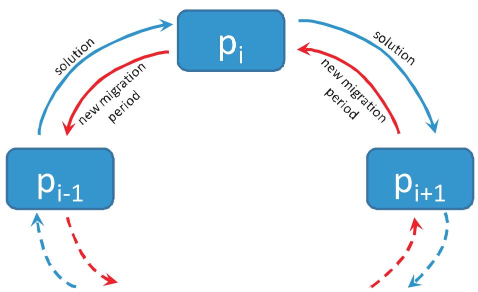

Figure 1: Outline of the ring topology of Proact.

subpopulation p_{i-1}/p_{i+1}. The selection and replacement strategies are, respectively, sending the best and always replacing the worst. The migration rate (or the number of individuals involved in each operation) is one.

Summarizing, each subpopulation p_i of the PROACT sends: *i*) its migration period to the left neighboring subpopulation in the ring (island p_{i-1}) in a proactive way, and *ii*) the best solution found so far to the right neighboring subpopulation in the ring at a frequency indicated by the p_{i+1} island. Consequently, there is a double sense of information flow between the islands. Figure 1 outlines this scheme.

3. RESULTS

The goal of this section is to evaluate PROACT in terms of its effectiveness with respect to its homogeneous counterparts, in which the migration period is fixed and preprogrammed for the entire execution. They have been named Hom$<mig_period>$, with $mig\_period \in \{32, 64, 128, 256, 512\}$, that is, from a strong coupling among islands to fairly isolated search. As a testbed, we have used two different problems: a large instance of the the Massively Multimodal Deceptive Problem (MMDP) with $k = 40$ deceptive subproblems, and a Knapsack Problem instance with 1000 items (K100-1000).

The common settings for all the algorithms is as follows. The whole population is composed of 512 individuals, divided in 8 islands (each island is physically run on a separate processor). The maximum number of generations is fixed at 5000. The tentative solutions for the problems are encoded as binary strings. The genetic operators used within the evolutionary loop are binary tournament selection, two point crossover, and bit flip mutation. The crossover rate is set to 0.65, meanwhile the mutation rates is set to $1/L$, where L is the length of the solutions. Proportional selection is used to build up the next population. The values for the lower and upper bound that triggers the proactive actions of PROACT, i.e., $\underline{H}$ and $\overline{H}$, has been set to 0.3 and 0.6, respectively.

Both PROACT and the variants of homogeneous dGAs are able to find the optimal values for the problems considered,

Table 1: Experimental results of Proact and homogeneous dEAs.

	MMDP	K100-1000
PROACT	254.03	633.67
Hom32	265.03	1040.10
Hom64	321.03	929.67
Hom128	356.13	846.03
Hom256	348.77	1079.27
Hom512	326.03	1222.37

so we are enabled to analyze the numerical effort of the different approaches. That is, Table 1 shows the average number of generations (over 30 independent runs) that the algorithms considered in this study need to reach the optimal solution for each instance.

The results show that PROACT is the algorithm that required the lower number of generations to reach the optimal solution for each instance of the MMDP and Knapsack problems. That is, the controlled in the population diversity of PROACT has allowed the algorithm to maintain enough genetic material to avoid premature convergence towards local minima. This is of special interest in many optimization scenarios in which a reduced number of function evaluations is a must (e.g., simulation optimization). The results obtained are coherent with previous findings [2].

4. CONCLUSIONS AND FUTURE WORK

In this work, we have introduced PROACT, a distributed EA which proactively controls the migration period of the neighboring subpopulations in order to maintain a good genetic diversity within each island. Decisions are made based on the Shannon entropy. We plan to extend PROACT for adapting other configuration parameters (migration policy or genetic operators) that allow the algorithm to have a better control of the search.

Acknowledgments

We acknowledge the Universidad Nacional de La Pampa, the ANPCYT, PICTO2001-0278 and CONICET in Argentina from which the first author receive continuous support. The work of Francisco Luna and Enrique Alba has been partially funded by the Spanish Ministry of Economy and Competitiveness and FEDER under contract TIN2011-28194. Francisco Luna also acknowledges support from TIN2011-28336.

5. REFERENCES

[1] A. Eiben, R. Hinterding, and Z. Michalewicz. Parameter control in evolutionary algorithms. *IEEE Trans. on Evolutionary Compututation*, 3(2):124–141, 1999.

[2] C. Salto, F. Luna, and E. Alba. Heterogeneity through proactivity: Enhancing distributed eas. *Seventh Int. Conf. on Parallel, Grid, Cloud and Internet Computing (3PGCIC)*, pages 279–284, 2012.

[3] C.E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423 and 623–656, 1948.

[4] R. Tanese. Distributed genetic algorithms. *Proc. of the Third Int. Conf. on Genetic Algorithms*, pages 434–439, 1989.