

Devising Adaptive Migration Policies for Cooperative Distributed Genetic Algorithms

Edgar Noda^{*}, André L.V. Coelho^{*}, Ivan L.M. Ricarte^{*}, Akebo Yamakami^{*} and Alex. A. Freitas[#]

^{*} School of Electrical and Computer Engineering (FEEC)
State University of Campinas (Unicamp)
{edgar, akebo}@dt.fee.unicamp.br
{coelho, ricarte}@dca.fee.unicamp.br

[#]Postgraduate Program in Computer Science
Pontifical Catholic University of Paraná (PUC-PR)
alex@ppgia.pucpr.br

Abstract-- Distributed Genetic Algorithms (DGAs) constitute an interesting approach to undertake the premature convergence problem in evolutionary optimization. This is done by spatial partitioning a huge panmictic population into several semi-isolated groups, called demes, each evolving in parallel by its own pace, and possibly exploring different regions of the search space. At the center of such approach lies the migratory process that simulates the swapping of individuals belonging to different demes, in such a way to ensure the sharing of good genetic material. In this paper, we model the migration step in DGAs as an explicit means to promote cooperation among genetic agents, autonomous entities encapsulating GA instances for possibly tackling different sub-problems of a complicated task. The focus is on the characterization of adaptive migration policies in which the choice of what individuals to migrate and/or replace is not defined *a priori* but according to a more knowledge-oriented rule. Comparative results obtained for a data-mining task were conducted, in order to assess the performance of adaptive migration according to efficiency/effectiveness criteria.

Index terms-- distributed genetic algorithms, adaptive migration policies, multiagent systems.

I. INTRODUCTION

The execution of a genetic algorithm (GA) can be regarded as a parallel search engine acting upon a population of feasible problem solutions codified as chromosome strings [10]. The GA success is very dependent on a proper balance between the explorative/exploitative activities performed throughout its run. When such balance is disproportionate, a premature convergence side-effect may appear, fruit of the lack of genotypical diversity. Moreover, the higher and more complex the search space, the more demanding will be the computational power required to realize the optimization process. One approach devised to tackle such problems is the distributed genetic algorithm (DGA) model [3][9], which is based on the spatial separation philosophy applied over the strings population.

The basic idea of DGAs (also known as coarse-grained parallel GAs) lies in the partition of the population into several small semi-isolated subpopulations (*demes*), each one being associated to an independent GA and possibly exploring different promising regions of the search space. These demes may, occasionally, *interact* with their

neighborhoods through the exchange of few individuals, simulating a seasonal migratory process (*migration* or *swapping*) by means of which new genotypical material can be injected and precipitated convergence can be avoided. This hopefully ensures that good genetic material be shared from time to time throughout the genetic system. The exchange typically takes the form of copying individuals between the demes [3] inasmuch as the same codification semantics is preserved through all the basic GAs. Otherwise, some kind of translation mechanism is necessary.

In this paper, we model the migration step in DGAs as an explicit means to promote cooperation among *genetic agents*, autonomous entities encapsulating GA instances for possibly tackling different sub-problems of a complicated task. The focus is on the characterization of *adaptive migration policies* in which the choice of what individuals to migrate and/or replace is not defined *a priori* but according to a more knowledge-oriented perspective. This is in contrast to more conventional static heuristic rules, like to emigrate a random or the current best individual, which sometimes show inefficiency aspects such as replication of genetic material.

In what follows, we briefly describe the basic steps underlying a typical DGA; comment on some related work; concentrate on the aspects behind MAS-DGA (our framework of distributed genetic agents); introduce and characterize some adaptive migration policies; present some simulation results for assessing the pros and cons of using adaptive migration policies; and finally address some remarks on future work.

II. IMPORTANT ISSUES ON DISTRIBUTED GENETIC ALGORITHMS

The application of DGAs to complicated optimization tasks is usually justified in the light of two advantages shown: (i) the maintenance of diversity owing to the semi-isolation of the sub-populations; and (ii) the easy implementation of DGAs in parallel architectures. The former is prominent for avoiding premature convergence and for multimodal optimization [1] as it prevents that all individuals be clones of the current best one. The latter is interesting for achieving improvements on computational performance.

Although there are a lot of different types of DGAs, all of them may be considered as variants of the "canonical DGA", which encompasses the following steps [9]:

- 1 Generate at random a population P of chromosomes;
- 2 Divide P into N_d demes;
- 3 Define the topological structure of the system;
- 4 Execute, in parallel for each deme, the next sub-steps:
 - 4.1 Apply, during f_m generations, the genetic and selection operators (customized to each GA instance);
 - 4.2 Send r_m individuals to neighboring demes;
 - 4.3 Receive new chromosomes and replace old ones;
- 5 Verify if the stopping criteria are satisfied.

In such framework, some issues deserve distinguished consideration. For instance, the neighborhood structure, or *topology*, indicates who communicates directly with whom, and, by this means, may be decisive for the whole DGA efficiency. Concerning this parameter, DGAs may be categorized as following either the *island model* (fully-connected demes) or the *stepping-stone model* (interaction restricted by customized logical or physical neighborhood). The choice of one arrangement against the other surely should observe the tradeoff "time spent in computation versus time spent in communication" [3].

As well, the deme connections may be either *static* (fixed *a priori*) or *dynamic* (reconfigured along the run). Still in this context, there are also some interesting approaches promulgating the creation of hierarchical DGAs (HDGAs) [9], DGAs whose nodes are other simple DGAs being connected one to the other. In such setting, two types of migrations are produced, local and global, according to the level of granularity we are looking at.

Another important issue in DGAs refers to the *number* of subpopulations (N_d) as well as the *size* (S_d) of each of them. Usually, the former is set in advance in consonance with the complexity of the problem at hand. The second should be kept small enough to make it possible to drift into a set of gene frequencies that correspond to a higher peak, after the introduction of good foreign genetic material. Yet, there are some approaches, like the one proposed by Bessaou and others [1], which dynamically stipulate the number and size of subpopulations according to the number of species (or clusters) current available in the whole system—speciation occurs via a distance operator that judges the similarity between strings.

Moreover, regarding the homogeneity of the demes, DGAs may be either known as *homogeneous* or *heterogeneous*, whether the configuration parameters, genetic operators or codification schemes are kept the same or not in all basic GAs. In this regard, Herrera *et al.* [9] have also examined the behavior of heterogeneous HDGAs (high-level DGAs making use of different basic homogenous DGAs) in well-known hard optimization problems.

Albeit most of the work in parallel evolutionary computation has focused, directly or indirectly, on topological aspects [3], there is by now some research focusing specifically on issues of the migratory process [4][5]. Basically, migration consists of an exchange of individuals between demes in such a manner to "perturb" their premature convergence and to foster the generation of new building blocks representing still unexplored regions.

The *migration rate* (r_m), which controls how many individuals to emigrate, the *migration interval* (f_m), which stipulates how frequent is the swapping process, and the *migration policy*, comprehending both which chromosomes should emigrate (*selection strategy*) as well as how to embody the incoming individuals (*replacement strategy*), are important parameters whose influence upon the DGA effectiveness should also be more investigated. Here, we also consider as part of the migration policy issue the choice of whether or not to replicate migrating individuals, i.e., to decide whether to send a clone or the string itself to the new deme. According to Herrera *et al.* [7], "if one does not copy individuals, it is possible that a subpopulation could be set back several generations in evolutionary terms by mass emigration of its best performers. On the other hand, simply copying individuals across could lead to highly fit individuals dominating several populations".

III. A FRAMEWORK OF DISTRIBUTED GENETIC AGENTS

In this work, we follow a novel perspective of modeling and implementing DGAs as a framework of multiple genetic search agents. Such framework, known as MAS-DGA, should serve as a testbed for studying aspects such as coordination, cooperation, communication, and problem decomposition, which are basic in DAI research [12].

In a MAS-DGA instance, each basic GA is encapsulated into an agent, an autonomous entity that must keep knowledge of the search, learning, or optimization problem it should operate on. Agents should be coordinated through a set of rules stipulating the topological and communication (migration) aspects, and these rules may be fixed *a priori* or set in run-time via a coordination entity (meta-agent). In this work, the topological issues are made fixed while the migration process is provided to be more adaptive (see next section), despite the fact that we do not apply any coordination entity.

Agents in MAS-DGA may also be classified as *homogeneous* or *heterogeneous* in accordance with the tasks they tackle, the settings of the GAs they represent, or the embedded knowledge they are endowed with (Fig. 1). By this means, it is possible to decompose huge, complicated tasks into smaller, simple subtasks to be each assigned to a different (heterogeneous) genetic agent. To make it possible that two heterogeneous agents communicate properly (that is, that their sent individuals be well-interpreted), maybe some kind of translation mechanism (indicated in the figure by an interface) could be necessary.

Distinct levels of optimization can also be envisaged, such as in a hierarchical perspective [9], wherein agents in the same level tackle identical subproblems (thus, being homogenous in a task accomplishment sense) and agents in lower levels provide solutions to subtasks of problems of higher-level agents (both considered as heterogeneous one to the other). In this work, however, we only employ one level of resolution in our simulation experiments. Furthermore, the application of MAS-DGA seems to be suited for dealing with multimodal [1], multiobjective [6], as well as dynamic, non-stationary [2] optimization problems, as one makes proper use of the homogenous/heterogeneous facility.

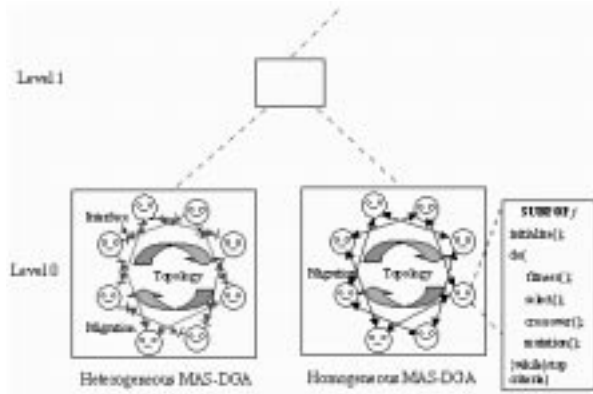


Fig. 1: MAS-DGA framework in a hierarchical perspective.

IV. ADAPTIVE MIGRATION POLICIES FOR COOPERATIVE DGAS

Although there is a bunch of work on DGAs or related material [3], most of them focus on the influence that topological aspects and/or the migration rate and migration interval exert on the final performance results. In this paper, we concentrate mainly on the migration policy issues [4] by contemplating them also as "first-class" parameters for improving the DGA performance.

In this context, Cantú-Paz [4][5] has provided a thorough investigation of how the policy used to select emigrants and the individuals they replace affects the selection pressure in multi-populated parallel EAs. Such research has focused primarily on how to quantify accurately the additional selection pressure caused by migration in order to predict precisely the number of generations until convergence. According to the author, "understanding the effect of the migration policies on the selection pressure is important because excessively slow or fast convergence rates may cause the search to fail. If selection is too weak the population may drift aimlessly for a long time, and the quality of the solutions found is not likely to be good. On the other hand, rapid convergence is desirable, but an excessively fast convergence may cause the EA to converge prematurely to a suboptimal solution" [4].

In the work of Cantú-Paz, there are two alternatives to select individuals for emigration: at random or by selecting the current best samples. As well, the replacement of existent individuals in the receiving deme with the incoming migrants may be done either randomly or by dropping out the current worst of them. The paper examines each of the four combinations of migrant selection and replacement within two perspectives: (i) by first computing the takeover time (which is a measure of how fast a good solution dominates a population once it is found), thus indicating the effect of the selection growth of good solutions; and (ii) by then calculating the increase in the selection intensity (used to compare the selection pressure of common selection methods with the one caused solely by migration). The achieved results showed that the migration policy that has caused the greatest reduction in workload is to choose both the emigrants and replacements according to their fitness (i.e., choose the best, replace the worst).

Conversely, in this work, we perceive the migration process through a different viewpoint, basically as a means

of promoting *cooperation* between MAS-DGA agents, and not only as a merely exchange of genotypical material. So, the migration policies to be adopted by the genetic agents should take such cooperation aspects into account, in order to promote the high performance of the team as a whole. Our interest is to embed more *quality* in the decision of whom to emigrate or to replace.

In this sense, we distinguish between *static* and *dynamic* (adaptive) migration policies, the former being fixed *a priori* and not changed throughout the DGA execution, and the latter being more context-oriented. So, the four migration policies followed in the study of Cantú-Paz are considered to be static (arbitrary simple rules of thumb). Another option would be to choose individuals with fitness values near to the average population fitness, as they would keep information about the (possibly promising) region of exploration by the deme and yet being less prone to entrapment in local valleys.

We advocate that with adaptive migration policies DGA agents turn to be more pro-active than passive in the search for the best solution(s). Moreover, by this means, it is possible to make more efficient use of the heterogeneous property of MAS-DGA. Static migration policies do work well in homogeneous DGAs, since the simple GAs work on the same optimization task and there is no requirement demanding each GA to maintain information on the task of its peers. Even in multimodal optimization [1], we may say that static policies have their effectiveness assured. However, this is not the same if some genetic agents are working on different, but also interdependent subtasks.

Adaptive (or dynamic) migration policies can be envisaged in consonance with two perspectives. The first encompasses the cases where each agent starts with a given static migration policy, which, according to the following circumstances of the search, may be allowed to vary. Such kind of adaptation may be achieved through a coordination entity or via a (multiagent) learning process. To our best knowledge, the impact of varying static migration policies along the DGA execution has not been investigated yet. However, in the preliminary results shown in Section 4, we do not make use of this kind of policy adaptation.

The second perspective on migration policy adaptation is more *knowledge-oriented*. In this case, there is no switch between static policies. Instead, we make use of some *migration policies mechanisms* that provide each agent with capability for reasoning about what and how many elements to emigrate (or to replace). In this paper, we propose and assess three types of mechanisms for emigration, namely:

1. *social awareness* (SA), where each agent knows the tasks (fitness functions) of (part or all) of its peers and then can send more appropriate (fitter) individuals to them;
2. *environment feedback* (EF), where each agent decision is guided by the (positive and negative) acknowledgments it receives from its peers about the quality of the individuals it sent before; and
3. *memory-oriented* (MO), where the current selection takes into account (some of) the past individuals already sent.

"Social awareness", actually, delimits a continuum between two ends, an agent only knowing the task of one of the peers to knowing the tasks of all of them. Moreover, it also encompasses the possibility that each agent being dynamically learning the tasks of the others, something we may call as "learning social awareness". This could be interesting to be exploited, for instance, in non-stationary multiobjective optimization problems [2][6], where the agent's tasks may be continuously updated.

Specifically for the EF emigration mechanism, we have conceived an associated replacement mechanism, named as *diversity-based*, which besides being fitness-oriented it also takes into account which of the new individuals provide higher levels of diversity in the deme. For the others, we have decided to use only the fitness-based static policy.

V. SIMULATION EXPERIMENTS

In this paper, we assess the effectiveness of applying adaptive migration policies for cooperative DGAs through a series of data mining experiments. The simulations aim at *dependence modeling* (DM) which is a kind of generalization of classification tasks where there are several goal attributes to be predicted instead of just one. We follow the approach suggested in [11] in which the user specifies a small set of potential goal attributes (each with more than one value) that she/he is interested in predicting. Although allowing more than one goal attribute, each prediction rule should only have a single goal attribute as its consequent. This does not impose, however, that different rules need to have the same goal attributes as their consequent.

In this work, the DM task is decomposed into N subtasks, where N is the number of potential goal attributes. Each subtask is then assigned to a MAS-DGA genetic agent, which is in charge of seeking for the best predicting rules of its associated goal attribute. The overall objective of this MAS-DGA instance is, thus, to concurrently discover good solutions for the whole DM task by assembling the sub-solutions found for each of its subtasks. This surely demands some kind of cooperative behavior among the simple GAs.

Our customized GA implementation (employed in each genetic agent) for tackling DM is called *GA Nuggets* [11], whose main configuration aspects are given briefly as follows. Each individual represents a candidate prediction rule of the form: IF *Ant* THEN *Cons*, where *Ant* is the rule antecedent (formed by a conjunction of conditions) and *Cons* is the rule consequent. An individual is encoded as a fixed-length string containing n genes (with discrete values), where n is the number of attributes, even though the genotype-phenotype mapping allows the creation of differently sized rules.

The prediction rules discovered by a data mining algorithm should satisfy three properties, namely, predictive accuracy, comprehensibility and interestingness. This is taken into account in the formulation of the fitness function [11] used in GA Nuggets. Regarding the genetic operators, we have employed in all experiments simple mutation (with rate equal to 5%), uniform crossover (with probability equal to 70%), and tournament-based selection (against 2 other players). By employing a kind of elitism, it is always possible to find at least one rule for each value of the given

goal attribute. Two other operators are necessary: (i) *insertion-removal*, for the random introduction or erasure of antecedent genetic material; and (ii) *consequent assembling*, for creating meaningful consequent parts [8].

We have conducted experiments in four distinct data sets obtained from the UCI machine learning repository (<http://www.ics.uci.edu/AI/Machine-Learning.html>). The data sets used are named as (Z)*oo*, (C)*ar Evaluation*, (A)*uto Imports* and (N)*ursery*, and were chosen for they seem to contain more than one potential interesting goal attribute.

According to the problem at hand, the number of genetic agents (demes) in MAS-DGA for DM is set. So, N_d equals 3, 2, 3, and 3 for data sets (Z), (C), (A), and (N), respectively. The size of each deme (S_d) is 25, so the whole panmictic population equals to $25 \times N_d$. The topological structure employed is fully-connected whereas the maximum number of generations for each deme is 150.

As indicated in the prior section, we have devised three emigration mechanisms for adaptive migration policies. The migration interval (f_m) is 5 for the all of them, despite the fact that, sometimes, the genetic agents are allowed to send no individuals for its peers (for EF and MO). This is why we refer to them as adaptive. The migration rate (r_m) is always one for SA and may vary within the range [0-5] for the others. For MO, the individuals to be sent are chosen following a diversity perspective (in relation to those sent before). For this purpose, a threshold of difference (30%, in the experiments) delimits those individuals that are allowed to emigrate. Conversely, for EF, the individuals to be transmitted should exhibit at least 30% of different genes in relation to those sent before but labeled as "rejected" by the destination demes.

Tables 1-4 show comparative results (obtained with cross-validation factor 10) between MAS-DGA instances implementing the three different adaptive migration policies regarding the predictive accuracy of the best-achieved rules for the four data sets. For reference purposes, we also present the results obtained for a simple GA Nuggets instance (whose population sizes $25 \times N_d$). From Table 5, which brings a more qualitative summary of the results, we can notice that:

- In average, MAS-DGA/SA has produced prediction rules that outperformed the simple GA rules eight times against one. For MAS-DGA/EF and MAS-DGA/MO, the results are even better, which certify the improvement achieved by employing MAS-DGA.
- Despite the fact that SA has produced rules with prediction accuracy lower than the ones achieved with the other migration mechanisms, its convergence is made faster as the selective pressure for each deme is always kept elevated along the generations. Since the task for agent j in SA is known by the others, and as they send to j individuals more fitter for its purposes (even though these individuals are evolved according to their own objectives), the selective activity for j is enhanced through a partial task-sharing.
- Another remarkable result refers to the number of swapping operations realized by each of the three emigration mechanisms along its MAS-DGA execution. Agents in SA have performed all 30 possible migrations as previously established. Conversely, agents in EF and MO, for the

Table 1: Predictive Accuracy (%) in the Zoo data set

Goal Attrib.	Attrib. Value	Single Population G.A.	Static social Awareness	Memory Diversity	Feedback Fitness – Diversity
Predator	False	50.5 ± 5.9	60.0 ± 1.9 (+)	63.3 ± 1.0 (+)	66.7 ± 0.9 (+)
	True	75.0 ± 11.2	85.0 ± 10.1	93.3 ± 6.0 (+)	88.0 ± 8.0
Domestic	False	97.1 ± 5.2	90.5 ± 4.4	93.1 ± 3.5	96.4 ± 1.8
	True	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Type	1	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0
	2	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0
	3	0.0 ± 0.0	89.0 ± 6.9 (+)	78.6 ± 6.9 (+)	90.0 ± 4.7 (+)
	4	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0
	5	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0
	6	90.0 ± 10.0	90.0 ± 10.0	90.0 ± 10.0	100.0 ± 0.0
	7	83.3 ± 7.2	85.0 ± 7.1	90.0 ± 6.7	96.7 ± 3.0 (+)

Table 2: Predictive Accuracy (%) in the Car Evaluation data set

Goal Attrib.	Attrib. Value	Single Population G.A.	Static social Awareness	Memory Diversity	Feedback Fitness – Diversity
Buying	V-high	60.0 ± 16.3	62.3 ± 14.7	75.0 ± 12.9	88.9 ± 9.9 (+)
	High	4.5 ± 3.0	8.5 ± 3.8	12.5 ± 3.2 (+)	8.3 ± 3.7
	Med	2.5 ± 2.5	6.0 ± 3.3	10.0 ± 2.9 (+)	8.3 ± 3.0 (+)
	Low	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0
Accept.	Unacc	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0
	Acc	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	Good	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	V-good	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0

Table 3: Predictive Accuracy (%) in the Auto Imports data set

Goal Attrib.	Attrib. Value	Single Population G.A.	Static social Awareness	Memory Diversity	Feedback Fitness – Diversity
Simb.	-3	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	-2	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	-1	55.0 ± 13.8	91.7 ± 4.3 (+)	90.5 ± 4.2 (+)	79.2 ± 9.9 (+)
	0	96.0 ± 2.7	98.0 ± 2.0	100.0 ± 0.0 (+)	100.0 ± 0.0 (+)
	1	70.0 ± 15.3	90.0 ± 10.0	88.9 ± 4.1	93.3 ± 3.1(+)
	2	63.3 ± 14.4	70.0 ± 10.2	87.5 ± 9.9	100.0 ± 0.0 (+)
	3	70.0 ± 15.3	70.0 ± 12.6	87.5 ± 4.8 (+)	90.5 ± 4.2 (+)
Body	Hardtop	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	Wagon	0.0 ± 0.0	13.3 ± 5.4 (+)	22.0 ± 4.2 (+)	19.0 ± 4.6 (+)
	Sedan	60.0 ± 12.3	85.5 ± 9.9 (+)	90.0 ± 4.1(+)	87.5 ± 4.2 (+)
	Hatch	76.7 ± 6.7	71.7 ± 5.4	74.1 ± 7.9	77.1 ± 4.1
	Convert.	40.0 ± 16.3	35.7 ± 6.3	43.7 ± 4.9	41.7 ± 4.8
Price	Low	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0
	Average	90.0 ± 4.1	81.7 ± 9.7	90.7 ± 3.4	92.7 ± 3.1
	High	72.5 ± 12.6	100.0 ± 0.0 (+)	100.0 ± 0.0 (+)	100.0 ± 0.0 (+)

knowledge they communicate (EF) or learn (MO), behave more intelligently in the decision of when to migrate. By these means, we have a trade-off of computation versus communication, which is typical in parallel GAs: SA implies quite no cost on additional computational calculus or resources (such as buffers), although it is more expensive in the communication viewpoint. The inverse is true for the other emigration mechanisms.

- Regarding the embodiment of incoming individuals, SA agents typically have shown lower (decreasing through time)

acceptance rates when compared with those for EF and MO. This owes to its static fitness-oriented replacement rule. Since MO and EF works with diversity-based replacement, such acceptance rate is kept higher.

VI. FINAL REMARKS

In this paper, we have presented a novel framework for implementing DGAs, called MAS-DGA. In this framework, each basic GA is encapsulated into an autonomous entity, which may be assigned to a different subtask. For promoting

Table 4: Predictive Accuracy (%) in the Nursery data set

Goal Attrib.	Attrib. Value	Single Population G.A.	Static social Awareness	Memory Diversity	Feedback Fitness – Diversity
Finance	Conv.	80.0 ± 13.3	100.0 ± 0.0(+)	100.0 ± 0.0 (+)	100.0 ± 0.0 (+)
	Inconv.	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0
Social	Non-prob	1.1 ± 1.1	0.0 ± 0.0 (-)	11.1 ± 0.8 (+)	7.4 ± 0.2 (+)
	Slightly prob	6.4 ± 4.3	9.1 ± 4.0	31.9 ± 2.2 (+)	19.38 ± 3.7 (+)
	Problem.	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0
Health	Recomm.	0.0 ± 0.0	0.0 ± 0.0	1.1 ± 1.1 (+)	0.0 ± 0.0
	Priority	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	Not recomm.	12.8 ± 9.8	41.8 ± 14.4(+)	59.7 ± 11.2	52.3 ± 12.6
	Spec priority	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0
	Very recomm.	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0

Table 5: Summary of Results

Average	Static social Awareness	Memory Diversity	Feedback Fitness – Diversity
Accuracy (vs. G.A)	8 – 1	15 – 0	16 – 0
Convergence (generation)	37.3 ± 7.5	62.1 ± 8.4	57.8 ± 6.9
Migrations	30 ± 0.0	13.3 ± 3.3	16.3 ± 5.5
Acceptance %	77.5 ± 2.6	87.0 ± 3.1	89.5 ± 2.7

cooperation among those genetic agents, we have devised new migration policies that are shown to be more adaptive when compared with those already available in literature [4]. For assessing the improvements achieved with these migration policies, we have performed several simulations of different MAS-DGA instances for data modeling tasks on four distinct data sets. The results ratify our expectations of improvement in efficiency (higher convergence and lower communication) as well as effectiveness (when compared with a single GA [11]).

As future work, we plan to devise novel adaptive migration policies with different emigrant/replacement mechanisms. As well, we shall also investigate the application of adaptive migration rates for MAS-DGA as (decreasing) functions of each deme current generation. Finally, the employment of MAS-DGA with adaptive migration policies in multiobjective dynamic problems [2][6] is subject to further research.

VII. REFERENCES

- [1] M. Bessau et al. A Multipopulation Genetic Algorithm Aimed at Multimodal Optimization, *Advances in Engineering Software*, 33(4), pp. 207-213, April 2002.
- [2] J. Branke. *Evolutionary Optimization in Dynamic Environments*, Kluwer, 2001.
- [3] E. Cantú-Paz. *Efficient and Accurate Parallel Genetic Algorithms*, Kluwer Academic Publisher, 2000.
- [4] E. Cantú-Paz. Migration Policies, Selection Pressure, and Parallel Evolutionary Algorithms. *Journal of Heuristics*. 7(4), 311-334. 2001.
- [5] E. Cantú-Paz. Migration Policies and Takeover Times in Parallel Genetic Algorithms, *Procs. of the Genetic and*

- Evolutionary Computation Conference (GECCO)*, p. 775, Morgan Kaufmann, 1999.
- [6] K. Deb. *Multi-objective Optimization using Evolutionary Algorithms*, John Wiley, 2001.
- [7] A.E. Eiben et al. Parameter Control in Evolutionary Algorithms, *IEEE Transactions on Evolutionary Computation*, 3(2), pp. 124-141, 1999.
- [8] A.A. Freitas. A Genetic Algorithm for Generalized Rule Induction, *Procs. 3rd On-Line World Conference on Soft Computing*, pp. 340-353, Springer-Verlag, 1999.
- [9] F. Herrera et al. Hierarchical distributed genetic algorithms, *International Journal of Intelligent Systems*, 14(11), pp. 1099-1121, November 1999.
- [10] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, 1996.
- [11] E. Noda et al. Discovering Interesting Prediction Rules with a Genetic Algorithm. *Procs. Congress on Evolutionary Computation (CEC-99)*, pp. 1322-1329, USA, July 1999.
- [12] G. Weiss. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, MIT Press, 1999.