A Social Behaviour Evolution Approach for Evolutionary Optimisation

Mikdam Turkey[*] and Riccardo Poli[†]

School of Computer Science and Electronic Engineering
University of Essex
Colchester CO4 3SQ
United Kingdom
[*]mturkey@essex.ac.uk, [†]rpoli@essex.ac.uk

ABSTRACT

Evolutionary algorithms were originally designed to locate basins of optimum solutions in a stationary environment. Therefore, additional techniques and modifications have been introduced to deal with further requirements such as handling dynamic fitness functions or finding multiple optima. In this paper, we present a new approach for building evolutionary algorithms that is based on concepts borrowed from social behaviour evolution. Algorithms built with the proposed paradigm operate on a population of individuals that move in the search space as they interact and form groups. The interaction follows a set of social behaviours evolved by each group to enhance its adaptation to the environment (and other groups) and to achieve different desirable goals such as finding multiple optima, maintaining diversity, or tracking a moving peak in a changing environment. Each group has two sets of behaviours: one for intra-group interactions and one for inter-group interactions. These behaviours are evolved using mathematical models from the field of evolutionary game theory.

This paper describes the proposed paradigm and starts studying its characteristics by building a new evolutionary algorithm and studying its behaviour. The algorithm has been tested using a benchmark problem generator with promising initial results, which are also reported.

Categories and Subject Descriptors

I.2.8 [Artificial Intelligence]: Problem Solving, Control Methods and Search—*Heuristic methods*; G.1.6 [Numerical Analysis]: Optimization—Global optimization

General Terms

Algorithms, Theory.

Keywords

Evolutionary Optimisation, Social Behaviour Evolution, Evolutionary Game Theory, Evolutionary Algorithms, Dynamic Optimisation Problems, Social Adaptive Groups.

1. INTRODUCTION

Nature has provided computer science with many sources of inspiration to develop a variety of optimisation approaches, of which natural selection or the Darwinian principle of "the survival of the fittest" has a lion's share [9]. While many types of Evolutionary Algorithms (EAs) have been developed based on Darwin's theory and our modern knowledge of genetics, rarely if ever EAs, in their original form, have naturally shown the full range of properties exhibited by natural evolution. In particular, a variety of extensions and modifications have been necessary in order to obtain EAs that could deal with multi-modal optimisation, multi-objective optimisation and Dynamic Optimisation Problems (DOPs) [28]. Under the pressure of selection, individuals with higher fitness survive for longer and/or reproduce more often. It stands to reason that, with most genetic operators and representations, this leads the population to converge into an area in the vicinity of an optimum in the fitness landscape, thereby losing diversity, and with it the ability to, for example, identify more than one optimum or to track a moving optimum. It is clear that when the natural selection process is based merely on an individual's fitness, losing diversity is an anticipated result and countermeasures have to be used. Consequently, to enable EAs to tackle these and other sorts of problems various techniques have been proposed which prevent convergence (i.e., maintain diversity) or re-diversify the population when necessary (e.g. [3, 7, 21, 22, 25, 29]).

In this paper we propose a different approach to building EAs which can potentially deal with the problems mentioned above and where populations show a natural tendency to maintain diversity and form groups. We take inspiration from the evolution of social behaviour. The approach uses a notion of *fitness of groups* which takes different measures related to a group's survival and performance into account. Each group has a set of social behaviours (operators) that individuals use in interacting with other individuals from the same or different groups. The exact nature of such behaviours is determined by a probability distribution which is tuned by an evolutionary process so as to maximise group fitness. Each behaviour serves a specific purpose and contributes to a group's survival or to the group's interaction with other groups. The behaviour probability distributions of each group are updated dynamically during the optimisation process using a dynamic mathematical model from evolutionary game theory [10, 11, 15].

Game theory was first introduced into evolutionary theory by Maynard Smith and Price who used it to model natural selection [19]. Subsequently many researchers have proposed models to deal with social behaviour evolution and population dynamics. In this paper, we use a simple dynamic mathematical model presented in [23] to evolve the social behaviours of groups. A distinguishing feature of our proposed approach is that the whole system is built based on notion of social behaviour evolution and evolutionary game theory. However there is some previous

GECCO'11, July 12–16, 2011, Dublin, Ireland.

relevant work which was inspired by similar ideas. For example an approach that incorporates ideas from game theory and social interaction into standard genetic algorithm to modify fitness values of individuals to slow down convergence and avoid local optima was proposed in [16]. This approach, which uses models from game theory to represent social interaction, improved the capability of problem solving of the standard genetic cycle. This approach is somehow related to co-evolutionary approaches [14, 20] in the dependency of an individual's fitness on its relationship with other individuals. It is worth mentioning that evolutionary game theory is different in many respects from classic game theory [24], especially in evolving the strategy (behaviour) distributions which represents the corner stone in our approach. In the general population structure and organisation, our approach has also some similarity to multi-population approaches in EA and Particle Swarm Optimisation (PSO), and niching techniques (e.g. [2, 4, 6]).

The rest of this paper is organised as follows. Section 2 introduces the proposed paradigm, its formulation and structure, and some relevant theoretical background. Section 3 presents an evolutionary system built based on our proposed paradigm. Some experimental results are also reported in that section. In Section 4 we discuss our findings and indicate some possible avenues for future work.

2. SOCIAL ADAPTIVE GROUPS

2.1. Background

In traditional types of EA, evolution is implemented by applying some form of selection pressure to distinguish and evolve fitter individuals. Normally, an individual's fitness value is evaluated according to the genetic material the individual carries. This is also known as *direct fitness* [1, 5]. However, in the real world, social individuals who live within groups and interact with other individuals have additional benefits which can be seen as an addendum to their fitness, known as *indirect fitness*. Together the two fitnesses form what is called *inclusive fitness* [8, 12, 13, 18]. The indirect fitness is the result of the influence of social behaviours on an individual's survival. The social behaviour of an individual (or a group of individuals), whether antagonistic or collaborative, is typically determined by the genetic similarity or dissimilarity with whomever the individual is interacting with. Usually, genetic similarity implies social collaboration, while genetic dissimilarity involves social competition [27]. This extension of natural selection, known as *kin* or *group selection*, has helped to interpret behaviours such as altruism, which are problematic for Darwin's theory of evolution [5].

Social interaction behaviours can be classified into four categories according to the change (increase or decrease) they cause to the fitness values of the initiator and the recipient. These four categories are: *altruism*, *spite*, *selfishness* and *cooperation* [12, 13, 26]. The pay-off of some behaviours is not immediate or direct to an individual's fitness. Instead, it may increase the *relative fitness* of the group or the specie in general, which in turn enhances the individual's fitness indirectly.

Based on the concepts above, the environment that the individual needs to be adapted to includes not only the actual environment (the fitness landscape) but also the other individuals from the same or different groups that interact with the individual, and have influence on the individual's fitness.

In applying these ideas to build a practical EA, we need to take several points into account. Firstly, social behaviour is a trait of a group of individuals that describes the way the individuals of the group interact with each other and with individuals from other groups. Secondly, like any traits contributing to an individual's survival, behaviours will be subject to an evolution process where good behaviour (i.e., one which enhances the chance of the group surviving) should be adopted and bad behaviour should go extinct [8]. Thirdly, the genotypic representation of an individual doesn't contain information about its social behaviours, simply because these behaviours are not part of the desired solution, despite the fact that these behaviours contribute to evolving that solution. So, a proper representation for social behaviours has to be introduced at the group's level.

2.2. Formulation and Structure

The proposed evolutionary system can be described as a tuple

$$E = \langle X, G, V, B_{intra}, B_{inter} \rangle$$

where $X = \{x_1, \dots, x_n \,|\, x_i \in R^{dim}\}$ represents a population of n real-valued individuals of length *dim*; G is the set of all possible groups, where $G \supset G_t = \{g_1, \dots, g_{N_t}\}$ represents the set of groups formed by individuals at time t; V is the group *behaviour probability distribution update function*; and, finally, B_{intra} and B_{inter} are two sets of transformations (operators) which represent the *intra-group and inter-group behaviours* used in pairwise interactions between individuals, respectively. The transformations are defined as follows:

$$\begin{aligned} (x_i', x_j') &= b(x_i, x_j), \quad \text{where } b \in B_{intra} \\ (x_i', x_j') &= b'(x_i, x_j), \quad \text{where } b' \in B_{inter} \end{aligned} \tag{1}$$

where b and b' are functions that transform two individuals into two new individuals, as x_i interact with x_j. The behaviours cause to change the position of individuals in the search space. So, individuals move as they interact.

A group $g \in G_t$ is defined as $g = \langle M_t, \alpha_t, \beta_t \rangle$ where

$$\begin{aligned} &M_t = \{x_i, x_j \in X \,|\, S(x_i, x_j) \le \tau\}, \quad \text{and} \\ &\alpha_t \in R_+^{|B_{intra}|} \text{and} \sum_{b \in B_{intra}} \alpha_t(b) = 1 \\ &\beta_t \in R_+^{|B_{inter}|} \text{and} \sum_{b' \in B_{inter}} \beta_t(b') = 1 \end{aligned} \tag{2}$$

The function S in (2) measures the similarity between a pair of individuals and τ is a threshold. This definition means that individuals can form groups based on their similarity. The formation of groups is a dynamic process as individuals move freely around the search space as a result of interactions. α_t and β_t are probability distributions over B_{intra} and B_{inter} at time t, respectively, and $\alpha_t(b)$ denotes the probability of using behaviour b by group g at time t.

A function V *is* used to evolve behaviours. This is done by updating their probability distribution. For a group g, V is defined as follows:

$$\theta_{t+1} = V(\theta_t, F(g), e_{\theta_t}) \tag{3}$$

where θ can be either α or β, $F(g)$ denotes a fitness function for groups, and $e_{\theta_t} \in R^{|B|}$ is a vector of effect rates of behaviours, where B could be either B_{intra} or B_{inter} and, for $b \in B, e_{\theta_t}(b)$ represents the effect rate of behaviour b. The factors that should be included in calculating the fitness of a group must reflect different aspects of the group well-being and must not be based merely on the individual direct fitness values. The effect value of behaviour measures the rate at which that specific behaviour contributes to the group fitness. Conveniently this will be used to calculate the behaviour *pay-off* value $u(b) = F(g)e_{\theta_t}(b)$ which is used in updating the behaviour probability.

Figure 1 shows the pseudo-code of the social adaptive groups evolutionary system. All random numbers are generated uniformly. The relative frequency of inter- and intra-group interactions is an important parameter of the algorithm that needs to be correctly set.

t=0
Generate an initial random population *X*
Evaluate population individual fitnesses
Form groups set G_t
Initialise behaviour distributions α_t and β_t for all $g \in G_t$
Repeat
  Repeat *//interaction round*
    Randomly **select** between *Inter*- or *Intra- group interaction*
    If *intra-group interaction* then
      Randomly **select** *x* and *y* from a group $g \in G_t$
      Randomly **select** *b* from B_{intra} according to α_t of g
    Else *//inter-group interaction*
      Randomly **select** $x \in g_1$ and $y \in g_2$ where $g_1 \neq g_2$ and $g_1, g_2 \in G_t$
      Randomly **select** *b* from B_{inter} according to β_t of g_1
    End if
    Compute $(x', y') = b(x, y)$
    Replace *x* and *y* with *x'* and *y'*, respectively
  Until *maximum number of interactions per iteration*
  t=t+1
  Evaluate population individual fitnesses
  Form groups set G_t
  Update behaviour distributions α_t and β_t for all $g \in G_t$ using eq. (3)
Until *t* reaches *maximum number of iterations*

Figure 1: Pseudo-code for proposed evolutionary system

2.3. Discussions

The proposed paradigm introduces a way of building EAs which embodies ideas from the theory of group selection. Instead of using a common mechanism to evolve optimum solutions, the proposed method evolves means of finding optimum solutions which are specific to groups and, thus, are specialised for different areas of the search space. This is achieved by evolving the behaviours for each group, which enhances a group's adaptation to the region of the fitness landscape where the group is located. This allows groups to adapt to dynamic environments, as behaviours change in response to group fitness, and this is affected by fitness landscape changes (as well as other features of the environment, including the composition of a group and its relationship with other groups).

A number of aspects need to specified in order to derive a concrete algorithm based on the proposed paradigm. First we have to define two sets of behaviours (B_{intra} and B_{inter}). Each behaviour is intended to operate on two randomly selected individuals and modify them in a way that leads to enhance groups fitness (relative fitness), but not necessarily an individual's fitness. Such behaviours must be designed to facilitate accomplishing any requirements the group has to meet. Also, the effect of each behaviour has to be measureable, so we can assess the pay-offs it produces and then identify which behaviour to blame for group fitness decreases, or which behaviour to credit for group fitness increases. This is the role of the effect rate of behaviours function, $e_{\theta_t}(b)$, which needs to be defined. Next, one has to specify the group fitness function, by which we tell the algorithm which features of a group are desirable. Also, we have to define the similarity function, *S*, which is essential for group formation. The last, and actually the most important, thing needed is a mechanism for updating the behaviour probability distributions (α_t and β_t) of each group. It is essential that any model for updating behaviour probabilities favour good behaviours that contributed to group "vitality" and to reduce the frequency of those having a negative impact. The model basically implements a natural selection process which evolves behaviour probability distributions suited for the group environment and provides a suitable balance between activities such as exploration and exploitation.

In the next section we address the issues raised above and provide concrete ideas to implement the proposed paradigm.

3. AN EA BASED ON SOCIAL ADAPTIVE GROUPS

3.1 Principles of the Algorithm

As shown by Figure 1, the algorithm starts by generating a random population, then after evaluating the individuals' fitness, groups are formed. For that purpose here we use the Euclidean distance as a similarity function (i.e., $S(x, y) = \sqrt{\sum_i (x_i - y_i)^2}$) in equation (2).

In order for a group to decide how to move individuals according to the sets of behaviours, a group uses information on the local area of the fitness landscape perceived by group members. This information is synthesised in a quantity we call *group centre*. For group *g*, the centre is defined as follows:

$$C = Centre(g) = \frac{\left(\sum_{i=1}^{N_{top}} top(g,i)\right)}{N_{top}}$$

where *top(g,i)* is a function that returns the i^{th} ranked member of the group *g* according to individual fitness and $N_{top} = 0.4 * |g|$ represents forty percent of the group size. As we will see later the motion of individuals caused by social behaviour interactions uses the group centre as a reference.

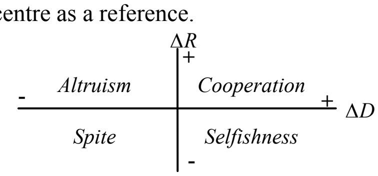


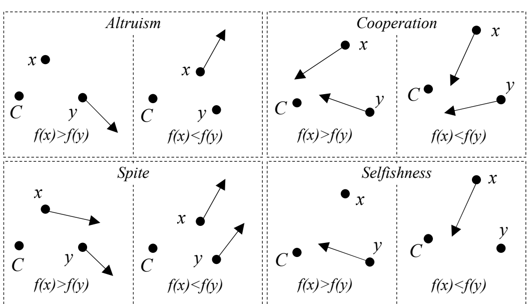
Figure 2: Categories of social behaviours according to change in direct fitness

Taking inspiration from Hamilton's categorisation [12, 13] of social behaviour according to the change in direct fitness of donor (ΔD) and recipient (ΔR) of each behaviour (Figure 2), here we adopt the following set of behaviours,

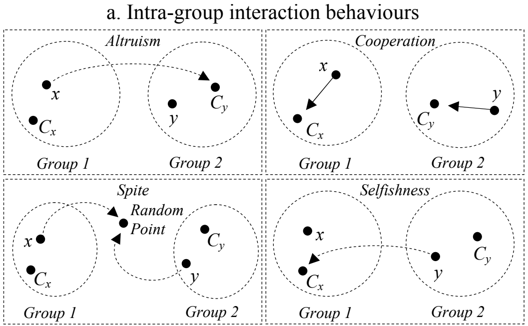
B= {*Cooperative, Selfish, Spiteful, Altruistic*} where B could be either B_{intra} or B_{inter}. As shown in equation (1) the behaviours update individual positions and, consequently their fitnesses.

Intra-group behaviours deal with moving individuals within the area where the group resides, whereas inter-group behaviours move individuals across areas. The intra-group behaviour directs individuals to the promising locations in the area occupied by a group, while, at the same time, exploring the surrounding areas and maintaining a good spread in the distribution of individuals. The inter-group behaviour, instead, moves individual between groups and also move individuals randomly to new spots in the fitness landscape to investigate the possibility of forming new groups there, in case the new area has enough resources to sustain a group. The change in an individual's position takes the form $x'=x+\Delta x$ where Δx is a displacement vector. Δx has to be computed in such a way to bring an individual closer to some target point and/or to push it away from some other point. Figure 3 depicts the two sets of social behaviours the proposed algorithm uses in inter- and intra- group interactions.

In intra-group interaction behaviours (Figure 3(a)), the direction of movement of an individual is decided on the basis of the fitness value of the individual with which the individual interacts, and also the position of that individual and the centre of the group. For example if we want to move x closer to both y and the centre of the group C, then we need to compute $d_1= x\text{-}y$ and $d_2=x\text{-}C$, where $\Delta x = -r_1*bias_y*d_1-r_2*bias_C*d_2$, where $bias_y$ and $bias_C \in [0,1]$ are suitable constants and r_1 and $r_2 \in [0,1]$ are two random numbers. If, instead, we want to move x away from the centre and closer to y then the change in its position can be computed as $\Delta x = -r_1*bias_y*d_1+r_2*bias_C*d_2$. And so on.



a. Intra-group interaction behaviours



b. Inter-group interaction behaviours

Figure 3: The proposed interaction behaviours

Computing Δ_x in inter-group interaction behaviours (Figure 3(b)) requires something different. If we are moving x closer to Z, where Z can be the centre of another group, a random point in the search space or the centre of x's group itself, then the displacement is computed as $\Delta_x = -d * r$, where $d = x - Z$ and $r \in [0.95,1.05]$ is a random number.

The group fitness function is a linear combination of three values which represent three different aspects of group quality: the ranking, the size, and the volume of the space occupied by the group. Formally the group fitness is defined as follows:

$$F(g) = \frac{|G_t| - Rank(g)}{|G_t|} + SizeFitness\left(Size_g, \frac{|X|}{|G_t|}\right) + \frac{\frac{Volume_t}{Volume_{pop}}}{|G_t|} \tag{4}$$

where g is a group and $Rank(g)$ is a function that gives the ranking of g among other groups. For the purpose of ranking, groups are sorted in descending order. The sorting is based on the value of the expression $\sigma*BestFitness+(1\text{-}\sigma)*AverageFitness$, where σ is a constant which allows weighing differently the fitness of the best individual in the group and the average of individual fitnesses. The top group's rank will be 0. The *SizeFitness* is as follows:

$$SizeFitness(S, Max_S) = \begin{cases} \frac{S}{Max_S} & if\ S < Max_S \\ 1 - \frac{S - Max_S}{Max_S} & otherwise \end{cases} \tag{5}$$

The output of this function increases as the value of S increases until it is greater than the value Max_S, beyond which the output starts to decrease. This function rewards groups with the "right size", bigger or smaller sizes leading to less group fitness. The volume of the group at time t, $Volume_t$ is the volume of a *dim*-dimensional sphere, the radius of which is computed as one half the diameter of the group (i.e., the distance between the two individuals further apart in the group). $Volume_{pop}$ is the volume of the search space (typically a multi-dimensional box).

The process of evolving behaviours tries to find the right combination of intra- and inter-group behaviours to put the groups in some state of dynamic equilibrium. The evolution process updates the behaviours to provide a group with the required operators to cope with different environmental changes, including changes that are caused by other groups as they compete or cooperate. After a round of interactions, the procedure that evolves behaviours works out how each behaviour has influenced the relative fitness (group fitness), so we can apportion blame and credit. For intra-group behaviours of group g, the effect rate is computed as follows:

$$e_{\alpha_t}(b) = \frac{\omega_t(b)}{\Omega_t(g)}\left(w_1(b)\frac{Size_t - Size_{t-1}}{Size_{t-1}} + w_2(b)\frac{A_t - A_{t-1}}{A_{t-1}} + w_3(b)\frac{Volume_t - Volume_{t-1}}{Volume_{t-1}}\right) \tag{6}$$

where $b \in B_{intra}$ and $\omega_t(b)$ is the number of occurrences of behaviour b and $\Omega_t(g)$ is the total number of behaviours that caused changes to the group, by interaction behaviours initiated by group members or by members of other groups. A_t is the average fitness of group members at time t. The values of weighting parameters $w_i(b)$ are shown in Table 1

Table 1: The values of weighting parameters $w_i(b)$

Behaviours (b)	$w_1(b)$	$w_2(b)$	$w_3(b)$
Cooperative	0	0.5	-0.5
Selfish	0	0.5	-0.5
Spiteful	-0.33	-0.33	0.33
Altruistic	-0.33	-0.33	0.33

The effect rates of inter-group behaviours B_{inter} are given by:

$$
\begin{aligned}
e_{\beta_t}(Cooperative) &= 1 - \left|\frac{|G_t| - |G_{t-1}|}{|G_{t-1}|}\right| + 1 - \left|\frac{Size_t - Size_{t-1}}{Size_{t-1}}\right| \\
e_{\beta_t}(Selfish) &= 1 - \left|\frac{|G_t| - |G_{t-1}|}{|G_{t-1}|}\right| + \frac{Size_t - Size_{t-1}}{Size_{t-1}} \\
e_{\beta_t}(Spiteful) &= \frac{|G_t| - |G_{t-1}|}{|G_{t-1}|} - \frac{Size_t - Size_{t-1}}{Size_{t-1}} \\
e_{\beta_t}(Altruistic) &= 1 - \left|\frac{|G_t| - |G_{t-1}|}{|G_{t-1}|}\right| - \frac{Size_t - Size_{t-1}}{Size_{t-1}}
\end{aligned}
\tag{7}
$$

After computing the effect rates and the group's fitness, we can update the behaviour distributions of the group and prepare for the next round of interactions. Equation (3), introduced above, needs to be instantiated to compute new behaviour distributions. Here, we realise such an equation as a dynamical system operating on behaviour distributions. First, we find the behaviours *pay-off* $u(b) = F(g)e_{\theta_t}(b)$ where $b \in B$, and B could be either B_{intra} or B_{inter}. The average of the pay-off of the two (intra- and inter-group) mixed behaviours is as follows:

$$U(\alpha_t) = \sum_{b \in B_{intra}} u(b)\alpha_t(b),$$

$$U(\beta_t) = \sum_{b \in B_{inter}} u(b)\beta_t(b)$$

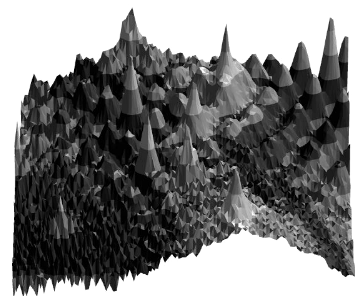
Then we use the *replicator equation* [23] to find the new distributions of group behaviours. Namely,

$$
\begin{aligned}
&\dot{\alpha}_t(b) = \alpha_t(b)\big(u(b) - U(\alpha_t)\big) \\
&\alpha_{t+1}(b) = \dot{\alpha}_t(b) + \alpha_t(b) \;\; for\ b \in B_{intra} \\
&\dot{\beta}_t(b) = \beta_t(b)\big(u(b) - U(\beta_t)\big) \\
&\beta_{t+1}(b) = \dot{\beta}_t(b) + \beta_t(b) \; for\ b \in B_{inter}
\end{aligned}
$$

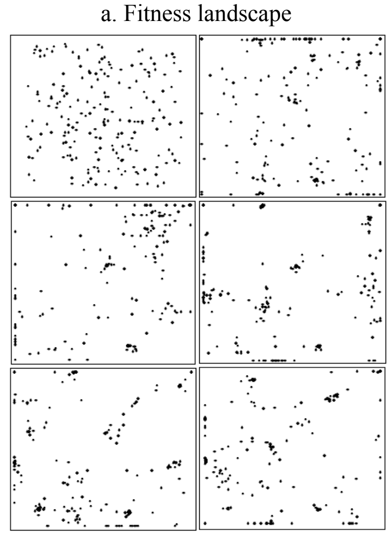
3.2 Experimental Results

In our implementation, we used two more conditions to enhance the process of group formation. The first one is that we set a limit to the maximum number of individuals in a group, i.e. $|M_t| \leq MaxSize$. *MaxSize* is chosen according to the volume of the landscape and to the size of the population. In the experiments reported below, where we use two-dimensional fitness landscapes and a population of size 200 individuals, we set *MaxSize* to 20. The second condition is that if the centres of two groups become too close, the group with the highest ranking pushes the centre of the other away. That doesn't mean that two groups of individuals cannot share the same space or overlap for some time. However it encourages the worse group to move away and explore different areas. Apart from these two conditions, there are no restrictions either on the groups or the motion of the individuals. So, groups may merge or split dynamically.

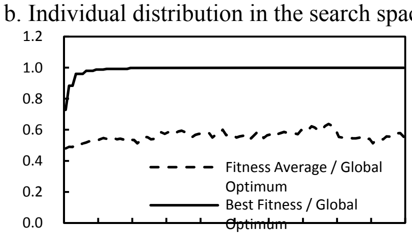
In order to better study the performance of the proposed algorithm and the general behaviour and progress of groups as they move in (and, thus, explore) the fitness landscape, we conducted experiments in two-dimensional search spaces. The fitness landscapes were created using the benchmark problem generator described in [17]. Figure 4 illustrates one such landscape and the results of a typical run. More specifically, Figure 4(a) shows the fitness landscape which is a composition of 10 different benchmark functions (i.e. Sphere, Rastrigin, and Griewank) after randomly changing the optimum location and rotating each function. In Figure 4(b), the distribution of individuals and the process of group formation are illustrated taking snapshots of the population at 20-iteration intervals (with the top-left panel showing the initial random population). Figure 4(c) describes the general behaviour of the algorithm from the point of view of the average of population fitness, the best-fitness-so-far in the run and the number of groups in the population. Note that fitness is relative to the fitness of the global optimum. So, every time the best fitness reaches 1, the global optimum has been discovered.



a. Fitness landscape



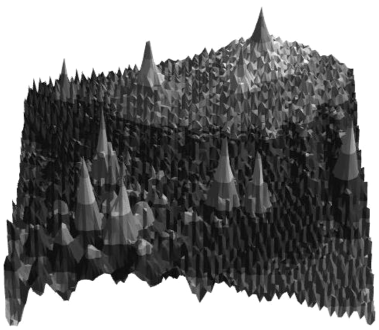
b. Individual distribution in the search space



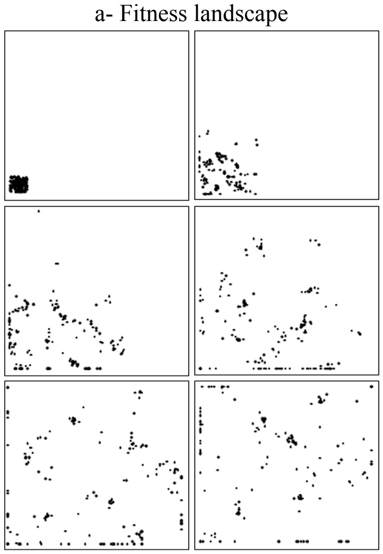
c. Performance of the proposed algorithm

Figure 4: The proposed algorithm in a 2-dimensional environment

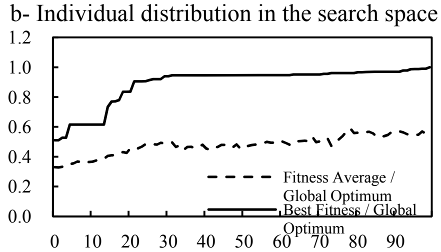
Figure 5 shows, for a different test landscape, an important feature of the propose algorithm: its ability to escape from locally optimal areas and expand the search over a global scale. As shown in Figure 5(b), in this test the initial population was artificially confined to a very small area. Yet, within a few iterations individuals spread and form groups well outside the initial area where they have been initially generated. In this particular experiment the optimum peak is located on the opposite side of the search space with respect to the area where individuals were initially located. The individuals were able to move through different peaks and cross the valley surrounding the global optimum area, to finally position a group right on the global optimum.



a- Fitness landscape



b- Individual distribution in the search space



c- Performance of the proposed algorithm

Figure 5: Another 2-dimentional experiment

To start exploring the scalability properties of the algorithm we also ran it on higher-dimensional landscapes generated using the same benchmark problem generator as above. Figure 6 illustrates the typical behaviour of the algorithm in runs with 5-dimensional problems, while Figure 7 shows the case of a 10-D problem. In both cases the best fitness clearly increases while the number of groups and average population fitness stabilise after a transient period. Notice that to cope with the larger dimensionality of spaces we used an increasing number of iterations: 100 for 2-D spaces, 750 for 5-D spaces and 1,500 for 10-D spaces.

Also we have tested the performance of the proposed algorithm in a dynamically changing environment. Figure 8 shows a typical response of the algorithm to dynamic changes in a two-dimensional fitness landscape. The change occurs every 50 rounds of interactions and it involves randomly changing the heights of the peaks (optima) and rotating their positions using random angles. We used large change steps, as suggested in [17], for both rotation angles and heights. Despite such rapid and dramatic changes, the algorithm rapidly adapted the population to the new environment, invariably re-identifying high fitness areas of the landscape. It is important to notice that the proposed approach doesn't need any change-detection mechanism or a change handling technique. The algorithm is able to detect a change automatically through the change in group fitnesses, which subsequently lead to changes in group behaviours resulting in adaptation to the new environment. The algorithm only requires an extra call to the fitness function per iteration to re-evaluate the best individual found so far.

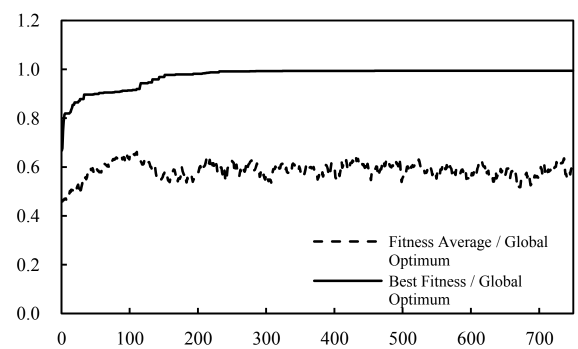


Figure 6: Performance in a 5-dimensional environment

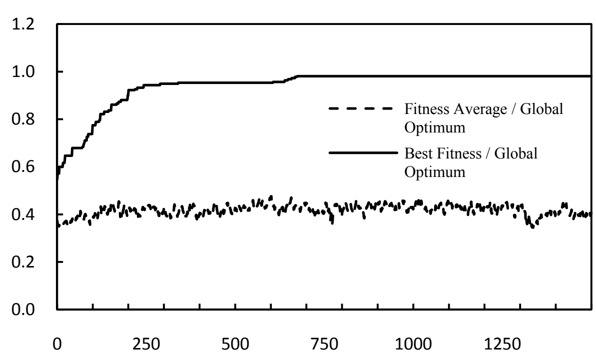


Figure 7: Performance in a 10-dimensional environment

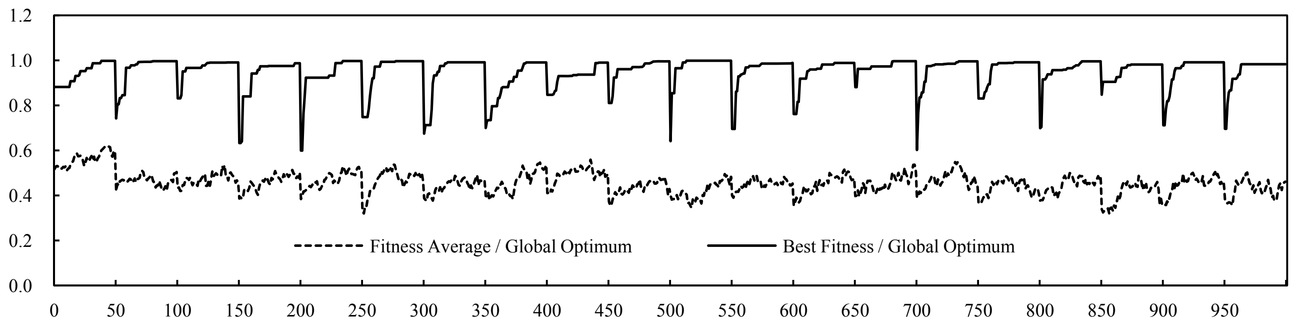


Figure 8: Performance in dynamic environment

4. DISCUSSION AND FUTURE WORK

This paper proposes some alternative sources of inspiration for building EAs: group selection and game theory. As a result of our analysis, we introduce a class of algorithms which work by forming groups of individuals, evolving a set of operators (behaviours) associated with each group and using them to control the movement of individuals in the search space. The movement of individuals depends on the individuals interacting through an operator and on the centres of the groups of the individuals involved in the interaction.

In our approach, individuals act as sensors and sources of information. Information is gathered and processed within the groups and is then used to direct the movement of individuals during interactions. Through inter-group interactions groups can bring in more members. This increases the "perception power" of a group and enhances its exploration. However, groups can also donate individuals to other groups or send them to random areas, making them effectively act as explorers in an expedition to discover different, and possibly better, regions of the search space where they can act as seeds for the formation of new groups. The real exploration, however, is done via intra-group interactions, where the individuals in a group are directed to explore promising areas within the region occupied by the group. In the preliminary experimentation we have performed so far, the approach has shown significant promise.

In our implementation of the proposed evolutionary system, we chose a group fitness function (equation 4) which is a combination of three different values, namely, Size, Volume and Ranking. These specify what we are asking the group to achieve. As mention above, the size indicates the perception power of the group, and by using the *SizeFitness* function (equation 5), we are trying to distribute the perception ability among the groups to ensure that each area of the fitness landscape is being explored reasonably well (of course, within the constraints imposed by the computational resources available). Using the volume in group assessment ensures that individuals are well-distributed in the group local area. That enhances the ability of the group to track a moving peak in dynamic environments and, most importantly to investigate the surrounding area of the group. The third component of the group fitness, the ranking, is of course a crucial one, as it is used to motivate the group to find better solutions. It is important to point out here that we have used the group ranking instead of using average of fitness because the ranking is less sensitive to small changes, but can reflect fitness landscape changes in dynamic problems, which acts as a mechanism for detecting change and as a trigger for changing the behaviours accordingly.

In light of the components of the group fitness function, we can see that each behaviour has a role in contributing and enhancing a specific feature of group fitness. For instance, the intra-group spiteful behaviour moves the individuals to potential low fitness areas increasing the volume of the group, which may cause to reduce the fitness of the affected individuals, but helps the group to spread around and monitor for a possible changes. Based on the action that the behaviour performs, the algorithm measures the effect rate (equations 6 and 7), then determines whether that effect has done good or bad for the group fitness and finally adjusts the behaviour accordingly using a mathematical evolution model.

From an evolutionary game theory point of view, in this work the groups play two games: one within the group between group members (symmetric game); the other between the group and other groups in the search space (asymmetric game). The nature of such games, in particular, whether it is cooperative or competitive, depends on the behaviours evolved by groups over time, which in turn depend on the environment and the interactions with other groups. Such behaviours are the means for a group to achieve what is require from it: to increase the fitness of the group (relative fitness). The frequency of using each behaviour by a certain group may vary over time depending on the nature of the fitness landscape and the interactions with other groups. That is why the process of updating behaviours is adaptive. The process that evolves the behaviour distribution requires finding the payoff of each behaviour and then, using a mathematical model, computing new distributions. In this cyclic process two important issues need to be considered carefully. First the way we measure the effect of behaviours and the way we measure the fitness of the group should be carefully chosen. The effects of behaviour, of course, depend on the kind of action the behaviour performs, while the group fitness depends on the goals the evolutionary system designer sets for the groups. Naturally, the designer needs to use a set of behaviours that is consistent with what is required from the group. In these conditions, evolution can discover and then favour the behaviours that contribute to the group fitness growth. The second issue is the model used to evolve the behaviours. Here we used a dynamic system to update the probability distribution for behaviours. The system uses feedback obtained after applying behaviours in a round of interactions to update their distributions. In particular here we used the replicator dynamic model [23]. However, in evolutionary dynamics there are a number of other models that are used to model the evolution and learning processes [10, 11, 15] and which might reasonably find application within our framework.

We plan to explore the two broad issues indicated above in future work. In the future we also need to start testing the proposed evolutionary system against a broader variety of different types of optimisation problems to see how it compares with state of the art algorithms. Also, further investigation is required to enhance the algorithm and to understand its emergent properties.

REFERENCES

[1] Alexander, R.D. and Tinkle, D.W. (eds.). *Natural selection and social behavior : recent research and new theory*. Chiron Press, Distributed outside North America by Blackwell Scientific Publications New York 1981.

[2] Blackwell, T. and Branke, J. Multi-swarm optimization in dynamic environments. *Applications of Evolutionary Computing*, *3005*. 489-500.

[3] Branke, J., Kaussler, T., Schmidt, C. and Schmeck, H. A multi-population approach to dynamic optimization problems. *Evolutionary Design and Manufacture* (2000). 299-307.

[4] Branke, J., Kaussler, T., Smidt, C. and Schmeck, H. A multi-population approach to dynamic optimization problems. *Evolutionary Design and Manufacture*. 299-307.

[5] Cartwright, J. *Evolution and human behaviour : Darwinian perspectives on human nature*. Palgrave Macmillan, Basingstoke 2008.

[6] Cedeno, W. and Vemuri, V.R. On the use of niching for dynamic landscapes. *Proceedings of 1997 Ieee International Conference on Evolutionary Computation (Icec '97)*. 361-366.

[7] Cobb, H. An investigation into the use of hypermutation as an adaptive operator in genetic algorithms having continuous, time-dependent nonstationary environments Naval Res. Lab., Washington, DC, Tech. Rep, 1990.

[8] Dawkins, R. *The selfish gene* Oxford University Press, Oxford, 2006.

[9] De Jong, K.A. *Evolutionary computation : a unified approach*. MIT Press, Cambridge, 2006.

[10] Durlauf, S.N. and Blume, L.E. (eds.). *Game theory*. Palgrave Macmillan, Basingstoke 2010.

[11] Fudenberg, D. and Levine, D. *The theory of learning in games* MIT Press, Cambridge, Mass. , 1998.

[12] Hamilton, W.D. The genetical evolution of social behaviour I & II. *Journal of Theoretical Biology*, *7* (1964). 1-52.

[13] Hamilton, W.D. The genetical evolution of social behaviour II *Journal of Theoretical Biology*, *7* (1964). 17-52.

[14] Hillis, W.D. Co-evolving Parasites Improve Simulated Evolution as an Optimization Procedure. *Artificial Life II*, *10*. 313-324.

[15] Hofbauer, J. and Sigmund, K. *Evolutionary games and population dynamics*. Cambridge University Press, Cambridge 1998.

[16] Lahoz-Beltra, R., Ochoa, G. and Aickelin, U., Cheating for problem solving: a genetic algorithm with social interactions. in *Genetic and Evolutionary Computation Conference (GECCO 2009)*, (Montreal, Canada, 2009), ACM, 811-818.

[17] Li, C., Yang, S., Nguyen, T., Yu, E., Yao, X., Jin, Y., Beyer, H. and Suganthan, P. Benchmark generator for CEC'2009 competition on dynamic optimization, University of Leicester, University of Birmingham, Nanyang Technological University, Tech. Rep, 2008.

[18] Maynard Smith, J. Group selection and kin selection. *Nature*, *201* (1964). 1145-1147.

[19] Maynard Smith, J. and Price, G. The logic of animal conflict. *Nature*, *246* (1973). 15-18.

[20] Potter, M.A. and De Jong, K.A. Cooperative coevolution: An architecture for evolving coadapted subcomponents. *Evolutionary Computation*, *8* (1). 1-29.

[21] Ramsey, C.L. and Grefenstette, J.J. Case-based initialization of genetic algorithms *Proceedings of the Fifth International Conference on Genetic Algorithms*, 1993, 84-91.

[22] Richter, H. and Yang, S.X. Memory based on abstraction for dynamic fitness functions *Proceedings Applications of Evolutionary Computing*, 2008, 596-605.

[23] Taylor, P. and Jonker, L. Evolutionary stable strategies and game dynamics. *Mathematical Biosciences*, *40* (1978). 145-156.

[24] Thomas L. Vincent and Brown, J.S. *Evolutionary game theory, natural selection, and Darwinian dynamics* Cambridge University Press, Cambridge :, 2005.

[25] Weicker, K. Performance measures for dynamic environments. *Parallel Problem Solving from Nature, LNCS 2439, Berlin,Germany: Springer-Verlag* (2002). 64-73.

[26] West-Eberhard, M. The evolution of social behavior by kin selection. *Quarterly Review of Biology*, *50* (1975). 1-33.

[27] West-Eberhard, M. Intragroup selection and the evolution of insect societies. *Natural selection and social behavior* (1981). 3–17.

[28] Yang, S., Ong, Y.-S. and Jin, Y. (eds.). *Evolutionary computation in dynamic and uncertain environments* Springer, Berlin 2007.

[29] Yang, S. and Yao, X. Population-based incremental learning with associative memory for dynamic environments. *Evolutionary Computation, IEEE Transactions on*, *12* (2008). 542-561.