

Techniques for Maintaining Diversity in Classical and Agent-Based Evolutionary Computation

Rafał Dreżewski
Marek Kisiel-Dorohinicki

Department of Computer Science
University of Mining and Metallurgy, Kraków, Poland
{drezew, doroh}@agh.edu.pl

Abstract. Niching techniques for evolutionary algorithms are aimed at forming and maintaining stable subpopulations (*species*) in multimodal domains. This paper reviews some of the existing niching techniques for classical evolutionary algorithms. Also the idea of a *niching coevolutionary multi-agent system* (NCoEMAS) is presented. In such a system the niche formation phenomena occur within one of the preexisting species as a result of coevolutionary interactions.

1 Introduction

Terms *Evolutionary Computation* and *Evolutionary Algorithms* cover a wide range of global search and optimization techniques based on analogies to natural evolutionary processes. However, both experiments and analysis show that even for multi-modal problem landscapes a simple evolutionary algorithm will inevitably locate a single solution — even when there are multiple solutions of equal fitness [8, 15]. This tendency is caused by the *genetic drift* [6]. In order to overcome this limitation some mechanism that creates and maintains different subpopulations (*species*) in a multimodal domain must be used.

In nature, there exist different subspaces within the environment (*niches*), which support different types of life (*species*). The number of organisms within each niche depends on niche fertility (its *carrying capacity*) and the efficiency at each organism exploits the niche resources. If there are too many organism in a given niche, the least efficient die because of resource shortages. And conversely, if there are too few organisms in a very fertile niche, they reproduce quickly so as resources of a given niche are fully exploited.

Traditionally, there are two basic approaches in evolutionary biology to understanding *speciation process* [7]. The first one called *allopatric speciation* occurs when subpopulations of a given species become geographically isolated. After isolation they follow different paths of evolution, which eventually lead to forming of different species. Resulting species are reproductively isolated even after secondary contact. The second kind of speciation is called *sympatric speciation*. Such speciation results from niche separation due to resource competition, predator-prey coevolution, sexual preferences, etc. [7]. In this case speciation process takes place within single population and geographical separation of subpopulations is not needed.

This paper starts with the overview of the existing niching techniques for classical evolutionary algorithms. Then the idea of a *niching coevolutionary multi-agent system* (NCoEMAS) is introduced, which opens new possibilities of modelling biological speciation mechanisms based on coevolutionary interactions, competition for limited resources, and geographical isolation. In such a system the niching (species formation) phenomena occurs as a result of coevolution of two different species: solutions and niches. Coevolution forces the process of speciation within the former one.

2 Niching Techniques for Evolutionary Algorithms

In case of a multimodal landscapes every peak can be treated as a niche (everywhere in this paper we assume maximization problems). The number of individuals that live within a niche should be in direct proportion to its carrying capacity. Carrying capacity in this case means peak's fitness relative to other peaks present in multimodal domain. This is called *niche proportionate population*.

Various mechanisms have been proposed to stably maintain species throughout the search process, thereby allowing to identify all peaks of a multimodal function. Most of these techniques allow niche formation through the implementation of *crowding*, *fitness sharing* or some modifications of these mechanisms. Crowding and sharing based techniques are inspired by speciation resulting from niche separation due to resource competition.

Every niching technique can also be classified as *parallel* or *sequential* [14]. Parallel niching methods form and maintain species simultaneously within a single population (regardless of the number of processors used). Sequential niching methods locate multiple peaks temporally, one after another.

Parallel evolutionary algorithms represent quite different approach to species formation, which is based on *allopatric speciation* [2].

2.1 Crowding Based Techniques

One of the first attempts to introduce niching into a genetic algorithm was Cavicchio's *preselection* scheme [3]. In preselection scheme offspring replaces less fit of two parents only when it has higher fitness than parent.

In the *crowding* technique [6] in each generation, a part of the population G (*generation gap*) is selected (via fitness proportionate selection) for reproduction. For each offspring, a certain number — CF (*crowding factor*) — of individuals are selected at random. The most similar individual, according to a similarity metric, is then replaced by the offspring. As a similarity metric De Jong used Hamming distance in genotypic space. Crowding does not promote the formation of stable species, but rather aims at maintaining the diversity of initial population.

Mahfoud developed niching mechanism called *deterministic crowding* [13]. He showed that similarity metrics based upon phenotypes should be preferred to genotype based ones. It was also demonstrated that there was very high probability that the offspring most similar individual should be searched among its parents. The new offspring is directly compared to their parents. In deterministic crowding parent is replaced

only if the child has higher fitness. To determine which of the two possible parent-child pairings should be used in the process of comparing parents to their offspring the total similarities were determined for each possible combination. The pairing that had the highest total similarity (according to some similarity metric) was used.

Probabilistic crowding developed by Mengshoel and Goldberg [16] is based upon Mahfoud's deterministic crowding. The main difference to deterministic crowding is the use of a probabilistic rather than a deterministic acceptance function. This means that stronger individuals win proportionally according to their fitness. The probability of winning the tournament by individual x is

$$P_x = P(x) = \frac{f(x)}{f(x) + f(y)} \quad (1)$$

where f is a fitness function.

2.2 Sharing Based Techniques

Fitness sharing was first introduced by Holland [12] and further developed by Goldberg and Richardson [10]. This technique models the ecological phenomenon of competition for limited resources between individuals that occupy the same niche. Fitness sharing technique reduces the fitness of individuals that have highly similar members within the population. Such a mechanism rewards unique individuals and punishes redundant individuals within the population. The reduced fitness of an individual i is given by

$$f'_i = \frac{f_i}{m_i} \quad (2)$$

where f_i is its raw fitness and m_i is the niche count. The niche count is given by

$$m_i = \sum_{j=1}^n sh(d_{ij}) \quad (3)$$

where d_{ij} is the distance between individual i and individual j , determined by a similarity metric. The similarity metric used can be based on either phenotype or genotype similarity. A genotype similarity metric is often domain independent (for example Hamming distance between the genotype bit strings). A phenotype similarity metric utilizes some knowledge about the domain, so it is more meaningful. The similarity metric should always return a distance d_{ij} , which increases as similarity decreases (and vice versa). If the distance between two individuals i and j is less than some fixed radius σ_{sh} the sharing function will return a value from $[0, 1]$, which increases with greater similarity between two individuals. In sharing technique each individual is considered to be the center of a niche with radius σ_{sh} . Fitness of each individual is reduced for every other individual that lives in its niche in a proportion to their similarity. The sharing function is given by

$$sh(d) = \begin{cases} 1 - (\frac{d}{\sigma_{sh}})^\alpha & \text{if } d < \sigma_{sh}, \\ 0 & \text{otherwise,} \end{cases} \quad (4)$$

where α is a constant that regulates the shape of the sharing function. It is commonly set to 1. Goldberg and Deb [9] developed formulas for determining the appropriate value for σ_{sh} given the expected number of peaks and assuming that maxima are regularly located in the search space.

Miller and Shaw [17] developed the niching technique called *dynamic niche sharing*. Their approach utilizes two assumptions. The first assumption is that the number of niche peaks, q can be estimated. The second assumption is that the peaks are all a minimum distance $2\sigma_{sh}$ from each other. Dynamic niche sharing technique attempts to identify the q peaks of forming niches and uses these dynamically identified peaks to classify all individuals as either belonging to one of these dynamic niches or belonging to the “non-peak” category. The shared fitness value for an individual that belongs to one of the dynamic niches is its raw fitness value divided by the dynamic niche population size. If the individual does not belong to a dynamic niche it belongs to the “non-peak” category and its niche count is calculated using the standard niche count equation (4). The shared fitness value for dynamic niche sharing is given by

$$f'_{dsh,i} = \frac{f_i}{m_{dsh,i}} \quad (5)$$

The dynamic niche count is

$$m_{dsh,i} = \begin{cases} n_j & \text{if ind. is within dyn. niche } j \\ m_i & \text{otherwise (non-peak individual),} \end{cases} \quad (6)$$

where n_j is the niche population size of the j th dynamic niche and m_i is the standard niche count. An individual i is considered to be within a dynamic niche j if its distance $d_{i,j}$ from peak j in the dynamic peak set is less than σ_{sh} . A greedy approach is used to identify the dynamic peaks for each generation. The population is sorted in decreasing raw fitness order. First the population member with the highest fitness is inserted into *Dynamic Peak Set* (DPS). Then for every individual in population array (in decreasing fitness value order) its distance from every peak in DPS is calculated. If all these distances are greater than σ_{sh} the individual is inserted into DPS. The whole process is continued until there are q peaks in DPS.

Coevolutionary sharing technique (CSN) was developed by Goldberg and Wang [11]. Their technique is loosely inspired by the economic model of *monopolistic competition*. The customer population is the usual population of candidate solutions. The businessmen will locate themselves in solution space in order to obtain largest payoff. Customer population C consists of n_c l -bit binary strings. Businessman population B consists of n_b l -bit binary strings. The distance between a customer c and a businessman b is the Hamming distance of string c and b ($|c-b|$). Customer c belongs to businessman b if b is the nearest businessman to c that is $\min_{b \in B} |c-b|$. The set of customers that belong to businessman b at generation t is denoted $C_{b,t}$. The number of customers a businessman b serves at generation t is $m_{b,t} = |C_{b,t}|$. The modified customer fitness is

$$f'(c) = \frac{f(c)}{m_{b,t}} \Big|_{c \in C_b} \quad (7)$$

The modified businessman fitness is

$$\phi(b) = \sum_{c \in C_{b,t}} f(c) \quad (8)$$

Simple CSN uses selectorecombinative GA for the customer population and selectomutational GA for the businessman population. Each customer is assigned to the closest businessman. Proportionate selection and genetic operators are used to compute a new customer generation. Each businessman is chosen in turn and a single mutation site is selected randomly. The resulting individual replaces its parent if it is at least d_{min} from other businessmen and it is an improvement over its parent. Otherwise another mutation site is selected (max. n_{limit} times). In the case of a massively multimodal function it was necessary to apply *imprint* mechanism. In CSN with imprint a candidate businessman is chosen from among the best individuals of the customer population.

2.3 Sequential Niching

One of the example of sequential niching methods is *sequential niche technique* (SN) developed by Beasley, Bull and Martin [1]. Their technique works by running multiple times a simple GA. The best solution of each run is maintained off-line. To avoid converging to the same niche, whenever a peak is located SN depresses the fitness landscape within some radius of that solution. The *niche radius* used in SN plays a role similar to that of σ_{sh} in sharing. In fact the authors suggest that SN is a sequentialization of fitness sharing.

The authors claim that there are three potential advantages of sequential niching: simplicity, ability to work with smaller populations, and speed (partially a byproduct of smaller populations). Mahfoud [14] showed that two latter potential advantages never materialize and what is more there are many disadvantages:

- Loss, through deration, of optimal solutions and their building blocks;
- Repeated search of depressed regions of the search space;
- Repeated convergence to the same solutions;
- Loss of cooperative population properties, including cooperative problem solving, and niche maintenance;
- Slower runtime, even on serial machines.

2.4 Parallel Evolutionary Algorithms

Parallel EAs (PEAs) represent quite different approach to species formation, which is based on *allopatric speciation* [2].

In the *island model* PEA [2] the population is divided into several subpopulations. Each subpopulation is assigned to a different processor (*island*). Individuals in each subpopulation are relatively isolated from individuals on another subpopulations. To exchange genetic material between islands, individuals with high fitness migrate occasionally from one subpopulation to another. All these techniques help maintaining genetic diversity and allow each subpopulation to search different part of multimodal domain.

Also *fine-grained* PEA [2] can be treated as a technique that makes possible the process of species formation. In this technique usually one individual is assigned to each processor. The individuals are allowed to mate only within a neighborhood, called a *deme*. Since neighborhoods overlap, the best individuals will propagate through the whole population.

There are two basic problems with the application of PEA to multimodal optimization. First, the number of niches (peaks of a multimodal function) is not known a priori, so there exist uncertainty about how many subpopulations should there exist. Second, there is no guarantee that different subpopulations will explore different areas of a search space.

3 Niching Coevolutionary Multi-Agent Systems

Most of niching techniques presented in the above sections *indirectly* model resource sharing within the niches (through crowding or fitness sharing techniques). On the other hand, PAEs model speciation caused by geographical isolation. However in nature there exist much more speciation mechanisms such as predator-prey coevolution, sexual preferences, etc. [7].

The idea of a *niching coevolutionary multi-agent system (NCoEMAS)* offers us new possibilities for modeling species (niche) formation mechanisms. These include explicit modelling of niches and resource sharing mechanism (*energy sharing*), predator-prey interactions, and sexual preferences.

3.1 Evolutionary Multi-Agent Systems

The main idea of *evolutionary multi-agent system (EMAS)* is the modelling of evolution process in multi-agent system (MAS) [5].

In opposition to classical EAs, in EMAS there is no centralized algorithm which manipulates the whole population. All individuals (*agents*) are independent and make their own decisions concerning reproduction, changing location etc. Agents live in an environment with defined structure. Agents can move within environment, changing their location. Every agent has some amount of resource called *life energy*. Every activity costs some energy (activities include: reproduction, migration, etc.) Energy can be gained only from the environment, and the rule is that better fit agents are given more energy than less fit ones.

It is very natural to implement the model of *allopatric speciation* in such a system because there can be many environments coexisting in parallel, each of them with different conditions. Agents can migrate between these environments, so the genetic material (information) is exchanged [4]. This model is, to some extent, similar to *island model* PEA [2] with all its advantages and disadvantages.

The idea of *niching coevolutionary multi-agent system (NCoEMAS)* presented in the following section offers us new possibilities of modeling niching-like behavior, that are not based on allopatric speciation.

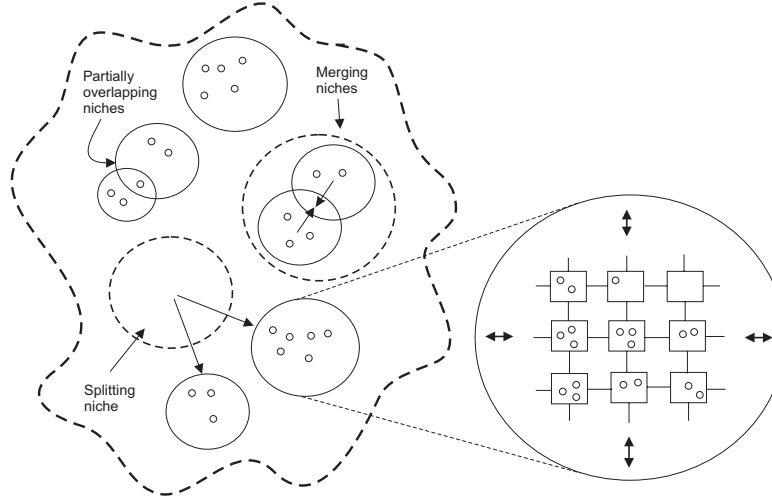


Fig. 1. A conceptual schema of a niching coevolutionary multi-agent system

3.2 The Idea of a Niching Coevolutionary Multi-Agent System

The key idea that differentiates *coevolutionary multi-agent system (CoEMAS)* from EMAS is the coevolution of several species. CoEMAS can be applied, for example, to multiobjective optimization and multimodal function optimization (NCoEMAS).

In NCoEMAS several (usually two) different species coevolve. One of them represents solutions. The goal of the second species is to cooperate (or compete) with the first one in order to force the population of solutions to proportionally populate and stably maintain niches in multimodal domain i.e. to initialize and maintain the process of speciation within the population of solutions.

It seems that NCoEMAS is especially suited for modelling sympatric speciation (niche separation due to resource competition, predator-prey coevolution, sexual preferences). Also, allopatric speciation can be modelled in NCoEMAS.

One of the possibilities is the coevolution of the population of individuals representing niches themselves and the population of individuals representing solutions (see fig. 1). In NCoEMAS we can model niches as individuals that are characterized by parameters like location, radius, etc. and evolve to best cover real niches in multimodal domain. Two additional operators can be introduced for niches: splitting and merging. Each niche can make decision on splitting into two niches based on the current distribution of its subpopulation. Also, the decision of merging can be made by two niches that are close enough and that are located on the same peak in the multimodal domain. In order to proportionally populate niches the mechanism of *explicit* resource sharing can be introduced. Agents' *life energy* can be treated as a resource for which individuals compete. This mechanism can be called *energy sharing*. Also, within each niche we can model the process of allopatric speciation. Each niche can have its inner spatial

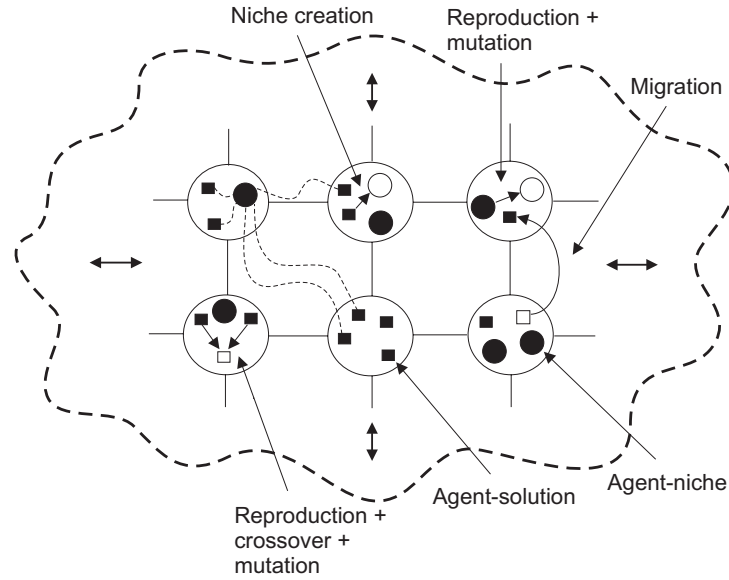


Fig. 2. NCoEMAS with agents-niches

structure in which agents live (see fig. 1). This will help in maintaining genetic diversity within each subpopulation.

3.3 NCoEMAS with Agents-Niches

In the particular case niches of NCoEMAS may be realized as agents (see fig. 2). There exist two different species: niches and solutions. All agents representing niches are located in nodes and can not change their location. Agents representing solutions are also located in nodes but they can change their location in environment migrating from node to node.

Every agent-solution has some amount of resource called *life energy*. There is closed circulation of energy in the system, which means that the total energy possessed by agents and the environment is constant during the whole simulation. Agents need energy for almost every activity: migration, reproduction etc. An individual dies when its energy is equal to 0. An agent can migrate from one node to another guided by the total energy of agents living in that node. The reproduction process can take place when agent's energy is above the given level. Agent starts reproduction, searches in its neighborhood for partner and then new agent is created. Mutation and crossover (one point crossover is used) are applied with the given probability in order to produce child's chromosome. An agent created in reproduction process obtains energy from the environment.

The behavior of the niche population is very similar to the algorithm used for businessman population in CSN [11]. Each time step a single mutation site is selected ran-

domly. The resulting individual replaces its parent if it is at least d_{min} from other niche and it is an improvement over its parent. Otherwise another mutation site is selected (max. n_{limit} times).

In the time t every agent-solution searches for the closest niche (the weighted sum of Hamming distance in genotype space and Euclidean distance in environment is used). If there is no niche, such that its distance from the agent is less than given value, then the new niche is created with the copy of agent's chromosome (imprint mechanism).

In each time step less fit agents must give some amount of their energy to better fit agents (according to fitness function). Agents are compared within niches and also outside niches in the environment space. The latter comparisons are realized within nodes. Given agent is compared with agents that stay in its node and also with agents from the neighboring nodes.

4 Concluding Remarks

In the paper a survey of niching techniques for evolutionary algorithms was presented. Most of classical niching techniques *indirectly* model resource sharing within the niches. On the other hand, parallel evolutionary algorithms model speciation caused by geographical isolation of subpopulations.

However, something more is needed if we want to model other biological speciation mechanisms. The idea is to allow the coevolution of several different species in EMAS. The resulting *niching coevolutionary multi-agent system (NCoEMAS)* allows us to model the process of *sympatric speciation* based on niche separation due to resource competition (*energy sharing*), predator-prey coevolution, sexual preferences, etc. At the same time *allopatric speciation* is modelled based on environmental structure of NCoEMAS.

References

1. D. Beasley, D. R. Bull, and R. R. Martin. A sequential niche technique for multimodal function optimization. *Evolutionary Computation*, 1(2):101–125, 1993.
2. E. Cantú-Paz. A survey of parallel genetic algorithms. Technical Report IlliGAL 97003, Illinois Genetic Algorithms Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL, USA, 1997.
3. D. J. Cavicchio. *Adaptive search using simulated evolution*. PhD thesis, University of Michigan, Ann Arbor, Michigan, USA, 1970.
4. K. Cetnarowicz and R. Drezewski. The mechanism of supervising the evolution process realized in multi-agent world. In *Proceedings of the 2nd International Conference on Formal Methods and Intelligent Techniques in Control, Decision Making, Multimedia and Robotics*, Warszawa, 2000. Polish-Japanese Institute of Information Technology.
5. K. Cetnarowicz, M. Kisiel-Dorohinicki, and E. Nawarecki. The application of evolution process in multi-agent world to the prediction system. In *Proceedings of the Second International Conference on Multi-Agent Systems — ICMAS'96*, Osaka, Japan, 1996. AAAI Press.
6. K. A. De Jong. *An analysis of the behavior of a class of genetic adaptive systems*. PhD thesis, University of Michigan, Ann Arbor, Michigan, USA, 1975.

7. M. Doebeli and U. Dieckmann. Evolutionary branching and sympatric speciation caused by different types of ecological interactions. Technical Report IR-00-040, International Institute for Applied Systems Analysis, July 2000.
8. D. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
9. D. E. Goldberg and K. Deb. An investigation of niche and species formation in genetic function optimization. In J. J. Grefenstette, editor, *Proceedings of the Third International Conference on Genetic Algorithms*, pages 42–50, San Mateo, CA, 1989. Morgan Kaufmann.
10. D. E. Goldberg and J. Richardson. Genetic algorithms with sharing for multimodal function optimization. In J. J. Grefenstette, editor, *Proceedings of the Second International Conference on Genetic Algorithms*, pages 41–49, Hillsdale, NJ, 1987. Lawrence Erlbaum Associates.
11. D. E. Goldberg and L. Wang. Adaptive niching via coevolutionary sharing. Technical Report IlliGAL 97007, Illinois Genetic Algorithms Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL, USA, 1997.
12. J. H. Holland. *Adaptation in natural and artificial systems*. The University of Michigan Press, Ann Arbor, Michigan, USA, 1975.
13. S. W. Mahfoud. Crowding and preselection revisited. In R. Manner and B. Manderick, editors, *Parallel Problem Solving From Nature, 2*, pages 27–36, Amsterdam, 1992. Elsevier Science Publishers (North Holland). IlliGAL report No. 92004.
14. S. W. Mahfoud. A comparison of parallel and sequential niching methods. In *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 136–143, 1995.
15. S. W. Mahfoud. *Niching methods for genetic algorithms*. PhD thesis, University of Illinois at Urbana-Champaign, Urbana, IL, USA, 1995.
16. O. J. Mengshoel and D. E. Goldberg. Probabilistic crowding: Deterministic crowding with probabilistic replacement. Technical Report IlliGAL 99004, Illinois Genetic Algorithms Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL, USA, 1999.
17. B. L. Miller and M. J. Shaw. Genetic algorithms with dynamic niche sharing for multimodal function optimization. Technical Report IlliGAL 95010, Illinois Genetic Algorithms Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL, USA, 1995.