Cooperative Coevolution: An Architecture for Evolving Coadapted Subcomponents

Mitchell A. Potter
Navy Center for Applied Research
in Artificial Intelligence
Naval Research Laboratory
Washington, DC 20375, USA
mpotter@aic.nrl.navy.mil

Kenneth A. De Jong
Computer Science Department
George Mason University
Fairfax, VA 22030, USA
kdejong@cs.gmu.edu

Abstract

To successfully apply evolutionary algorithms to the solution of increasingly complex problems, we must develop effective techniques for evolving solutions in the form of interacting coadapted subcomponents. One of the major difficulties is finding computational extensions to our current evolutionary paradigms that will enable such subcomponents to "emerge" rather than being hand designed. In this paper, we describe an architecture for evolving such subcomponents as a collection of cooperating species. Given a simple string-matching task, we show that evolutionary pressure to increase the overall fitness of the ecosystem can provide the needed stimulus for the emergence of an appropriate number of interdependent subcomponents that cover multiple niches, evolve to an appropriate level of generality, and adapt as the number and roles of their fellow subcomponents change over time. We then explore these issues within the context of a more complicated domain through a case study involving the evolution of artificial neural networks.

Keywords

Coevolution, genetic algorithms, evolution strategies, emergent decomposition, neural networks.

1 Introduction

The basic hypothesis underlying the work described in this paper is that to apply evolutionary algorithms (EAs) effectively to increasingly complex problems, explicit notions of modularity must be introduced to provide reasonable opportunities for solutions to evolve in the form of interacting coadapted subcomponents. A good example of such problems is behavior learning tasks, such as those one would encounter in the domain of robotics, where complex behavior can be decomposed into simpler subbehaviors. What makes finding solutions to problems of this sort especially difficult is that a natural decomposition often leads to interacting subcomponents that are highly dependent on one another. This makes coadaptation a critical requirement for their evolution.

There are two primary reasons traditional EAs are not entirely adequate for solving these types of problems. First, the population of individuals evolved by these algorithms has a strong tendency to converge in response to an increasing number of trials being allocated to observed regions of the solution space with above average fitness. This strong convergence property precludes the long-term preservation of diverse subcomponents, because any but the strongest individual will ultimately be eliminated. Second, individuals evolved by traditional EAs typically represent complete solutions and are evaluated in isolation. Since interactions between population members are not modeled, there is no evolutionary pressure for coadaptation to occur. To complicate matters further, we may neither know *a priori* how many subcomponents there should be nor what roles the subcomponents should assume. The difficulty then comes in finding computational extensions to our current evolutionary paradigms in which reasonable subcomponents "emerge" rather than being designed by hand. At issue is how to identify and represent such subcomponents, provide an environment in which they can interact and coadapt, and apportion credit to them for their contributions to the problem-solving activity such that their evolution proceeds without human involvement.

This paper is organized as follows. Section 2 begins with a discussion of some important issues related to evolving coadapted subcomponents and describes some previous approaches to extending the basic evolutionary model for this purpose. This is followed in Section 3 by a description of our architecture for evolving coadapted subcomponents as a collection of cooperating species. Section 4 investigates whether reasonable subcomponents will automatically arise through the use of such systems by attempting to show that evolutionary pressure to increase the overall fitness of the ecosystem can provide the needed stimulus for the emergence of an appropriate number of interdependent subcomponents that cover multiple niches, are evolved to an appropriate level of generality, and adapt as the number and roles of their fellow subcomponents change over time. Section 5 further explores the emergence of coadapted subcomponents through a case study in neural network evolution. Section 6 summarizes the main contributions of this paper and discusses directions for future research.

2 Evolving Coadapted Subcomponents

If we are to extend the basic computational model of evolution to provide reasonable opportunities for the emergence of coadapted subcomponents, we must address the issues of problem decomposition, interdependencies between subcomponents, credit assignment, and the maintenance of diversity.

Problem decomposition consists of determining an appropriate number of subcomponents and the role each will play. For some problems, an appropriate decomposition may be known *a priori*. Consider the problem of optimizing a function of K independent variables. It may be reasonable to decompose the problem into K subtasks, with each assigned to the optimization of a single variable. However, there are many problems for which we have little or no information pertaining to the number or roles of subcomponents that, ideally, should be in the decomposition. For example, if the task is to learn classification rules for a multimodal concept from preclassified examples, we probably will not know beforehand how many rules will be required to cover the examples or which modality of the concept each rule should respond to. Given that an appropriate decomposition is not always obvious, it is extremely important that the EA addresses the decomposition task either as an explicit component of the algorithm or as an emergent property.

The second issue concerns the evolution of interdependent subcomponents. If a problem can be decomposed into subcomponents without interdependencies, clearly each can be evolved without regard to the others. Graphically, one can imagine each subcomponent evolving to achieve a higher position on its own fitness landscape, disjoint from the fitness landscapes of the other subcomponents. Unfortunately, many problems can only be decomposed into subcomponents exhibiting complex interdependencies. The effect of changing one of these interdependent subcomponents is sometimes described as a "deforming" or "warping" of the fitness landscapes associated with each of the other interdependent subcomponents (Kauffman and Johnsen, 1991). Given that EAs are adaptive, it would seem that they would be well suited to the dynamics of these coupled landscapes. However, natural selection will only result in coadaptation in such an environment if the interaction between subcomponents is modeled.

The third issue is the determination of the contribution each subcomponent is making to the solution as a whole. This is called the *Credit Assignment Problem*, and it can be traced back to early attempts by Arthur Samuel (1959) to apply machine learning to the game of checkers. For example, if we are given a set of rules for playing a two-player game, such as checkers, we can evaluate the fitness of the rule set as a whole by letting it play actual games against alternative rule sets or human opponents while keeping track of how often it wins. However, it is far from obvious how much credit a single rule within the rule set should receive given a win, or how much blame the rule should accept given a loss.

The fourth issue is the maintenance of diversity in the ecosystem. If one is using an EA to find a single individual representing a complete solution to a problem, diversity only needs to be preserved in the population long enough to perform a reasonable exploration of the search space. Typically, once a good solution is found, the EA terminates, and all but the best individual is discarded. Contrast this to evolving a solution consisting of coadapted subcomponents. In the coevolutionary paradigm, although some subcomponents may be stronger than others with respect to their contribution to the solution as a whole, all subcomponents are required to be present in the final solution.

2.1 Previous Approaches

One of the earliest extensions to the basic evolutionary model for the support of coadapted subcomponents is the *classifier system* (Holland and Reitman, 1978; Holland, 1986). A classifier system is a rule-based system in which a population of stimulus-response rules is evolved using a genetic algorithm (GA). The individual rules in the population work together to form a complete solution to a target problem. A micro-economy model, in which rules place bids for the privilege of activation, is used to handle the interactions between population members, enabling them to be highly interdependent. Credit assignment is accomplished using an algorithm called the *bucket brigade*, which passes value back to the rules along the activation chain when the system is doing well. The complex dynamics of this micro-economy model results in emergent problem decomposition and the preservation of diversity.

A more task specific approach to evolving a population of coadapted rules was taken by Giordana et al. (1994) in their REGAL system. REGAL learns classification rules consisting of conjunctive descriptions in first order logic from preclassified training examples. Problem decomposition is handled by a selection operator called *universal suffrage*, which clusters individuals based on their coverage of a randomly chosen subset $\mathcal{E}$ of the positive examples. A complete solution is formed by selecting the best rule from each cluster, which provides the necessary interaction among subcomponents. A seeding operator maintains diversity in the ecosystem by creating appropriate individuals when none with the necessary properties to cover a particular element of $\mathcal{E}$ exist in the current population. The fitness of the individuals within each cluster is a function of their consistency with respect to the set of negative examples and their simplicity. This fitness evaluation procedure effectively solves the credit assignment problem but is highly specific to the task of concept learning from preclassified examples.

More recently, the performance of REGAL was improved through the use of semi-isolated population islands (Giordana and Neri, 1996). Each REGAL island consists of a population of conjunctive descriptions that evolve to classify a subset of the preclassified training examples. The universal suffrage operator is applied within each island population, and the REGAL seeding operator ensures that all examples assigned to an island are covered. Migration of individuals between islands occurs at the end of each generation. A supervisor process determines which examples are assigned to the islands and, occasionally, reassigns them so that each island will ultimately produce a single conjunctive description that will correctly classify all the training examples when disjunctively combined with a description from each of the other islands. As hypothesized by population geneticist Sewall Wright (1932), population islands enable the system to maintain more diversity and reach higher fitness peaks than possible with a single freely interbreeding population.

Multiple-species models, that is, those incorporating genetically isolated populations, have also been used to evolve coadapted subcomponents. Early work includes the application of a model of hosts and parasites to the evolution of sorting networks using a GA (Hillis, 1991). One species (the hosts) represents sorting networks, and the other species (the parasites) represents test cases in the form of sequences of numbers to be sorted. The interaction between the two species takes the form of complementary fitness functions. Specifically, a sorting network is evaluated on how well it sorts test cases, while the test cases are evaluated on how poorly they are sorted. Because the host and parasite species are genetically isolated and only interact through their fitness functions, they are full-fledged species in a biological sense. A two-species model has also been used to solve a number of game learning problems, including tic-tac-toe, nim, and go, by having the species represent competing players (Rosin and Belew, 1995). These competitive-species models have demonstrated that this form of interaction helps to preserve genetic diversity and results in better final solutions when compared with non-coevolutionary approaches. In addition, the credit-assignment problem is trivially solved through the use of complementary fitness functions. A limitation of these approaches, however, is their narrow range of applicability due to the requirement that the problem be hand-decomposed into two antagonistic subcomponents.

Other researchers have explored the use of cooperative-species models. Besides our approach, which will be discussed in detail throughout the rest of this paper, the coevolution of multiple cooperative species has been applied to job-shop scheduling (Husbands and Mill, 1991), and a two-species cooperative model was applied to Goldberg's three-bit deceptive function (Paredis, 1995). The decomposition used by the job-shop scheduling system was to have each species but one evolve plans for manufacturing a different component. The single remaining species evolved an arbitrator for resolving conflicts when two or more plans required the same piece of shop equipment at the same time. The two-species system for solving Goldberg's three-bit deceptive problem assigned one species to the task of evolving an effective representation in the form of a mapping between genes and subproblems and the other species to the task of evolving subproblem solutions. These two species had a symbiotic relationship in that the second species used the representations coevolved by the first species. While both of these cooperative-species models involved a hand-decomposition of the problem, our approach has emphasized emergent problem decomposition.

The application of EAs to the construction of artificial neural networks has also motivated extensions to the basic evolutionary model for the support of coadapted subcomponents. For example, in the SANE system (Moriarty, 1997; Moriarty and Miikkulainen, 1997), each individual in the population represents a single neuron by specifying which input and output units it connects to and the weights on each of its connections. A collection of neurons selected from the population constitutes a specification for constructing a complete neural network. A genetically isolated population of network *blueprints* evolves records of neurons that work well together. The neural-evaluation phase of SANE performs many cycles of selecting neurons from the population based on a blueprint, connecting the neurons into a functional network, evaluating the network, and passing the resulting fitness back to the blueprint. The fitness of a neuron is computed as the average fitness of the five best networks it participates in, which is effectively a measure of how well the neuron collaborates with other neurons in the population to solve the target problem. The relationship between the blueprint population and the neuron population is also collaborative. Rewarding individuals based on how well they collaborate results in the long term maintenance of population diversity and a form of emergent decomposition.

A more macroscopic approach has been taken by de Garis (1990) to evolve artificial neural networks for controlling simulated creatures. Rather than evolving a single large neural network to control a creature, de Garis first hand-decomposes the problem into a set of component behaviors and control inputs. A GA is then used to evolve small specialized neural networks that exhibit the appropriate subbehaviors. This is similar to the *chaining* technique pioneered by Skinner (1938) and used by the Reinforcement Learning community to train robots (Singh, 1992; Lin, 1993). Clearly, the human is very much in the loop when taking this approach.

Another biological model that has inspired the evolution of coadapted subcomponents is the vertebrate immune system (Forrest and Perelson, 1990; Smith et al., 1993; Forrest et al., 1993). Forrest et al. introduced a technique called *emergent fitness sharing* that preserves diversity in a single population and results in emergent problem decomposition by modeling some of the interactions that occur between antibodies and antigens. Emergent fitness sharing computes the fitness of each generation of antibodies by performing many iterations of a three-step process. First, a single antigen is randomly chosen from a fixed collection; second, a set of antibodies is randomly chosen from the evolving population; and third, a tournament is held to determine which of the selected antibodies matches the antigen most closely. The winner of each tournament receives a fitness increment based on the quality of the match. This model is similar in some ways to SANE, but the relationship between individuals is competitive rather than cooperative.

Finally, there has been work on extending the basic single-population evolutionary model to allow coadapted subcomponents specific to the construction of computer programs. Koza (1993), for example, has reported on the beneficial hand-decomposition of problems into a main program and a number of subroutines. Others have taken a more emergent approach through the exploration of techniques for automatically identifying blocks of useful code, generalizing them, and adapting them for use as subroutines in future generations (Rosca and Ballard, 1994, 1996).

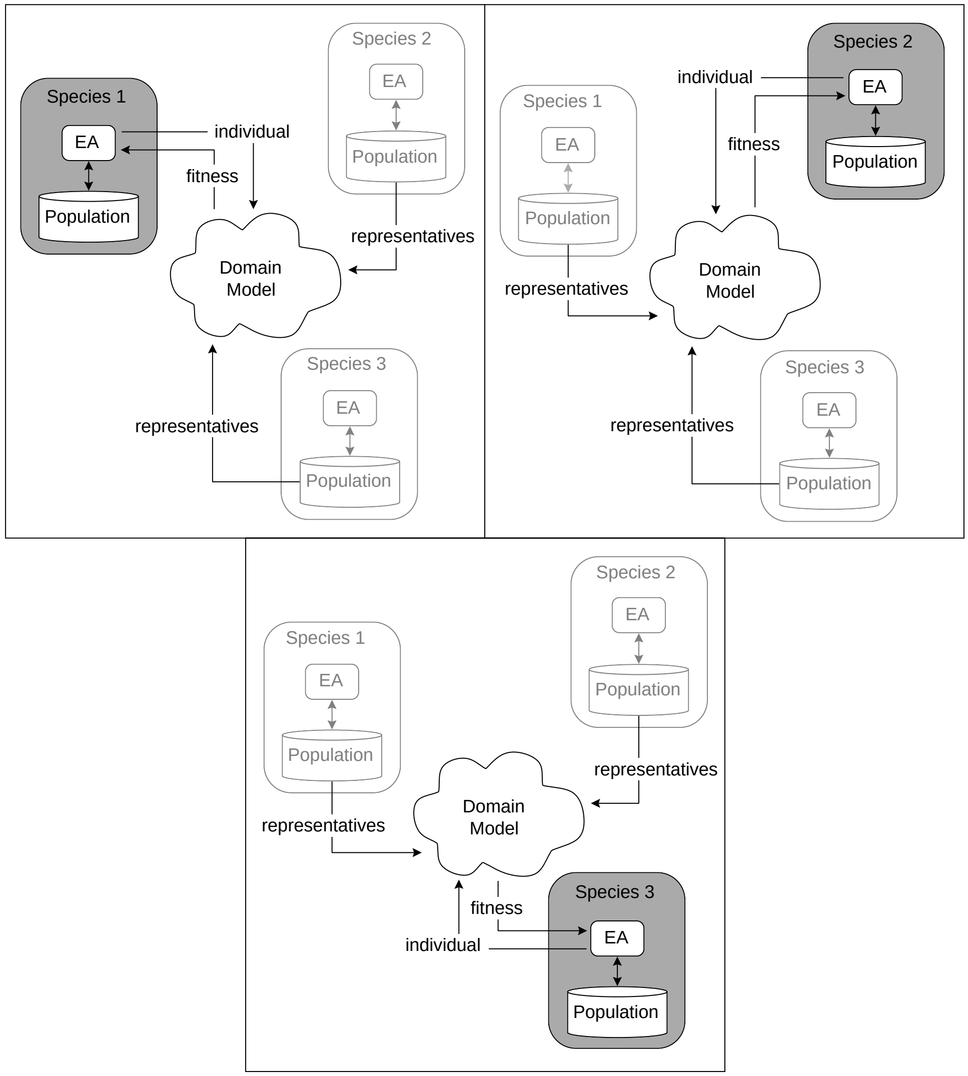

Figure 1: Coevolutionary model of three species shown from the perspective of each in turn.

3 Cooperative Coevolution Architecture

We now describe a generalized architecture for evolving interacting coadapted subcomponents. The architecture, which we call *cooperative coevolution*, models an ecosystem consisting of two or more species. As in nature, the species are genetically isolated—individuals only mate with other members of their species. Mating restrictions are enforced simply by evolving the species in separate populations. The species interact with one another within a shared domain model and have a cooperative relationship.

The basic coevolutionary model is shown in Figure 1. Although this particular illustration shows three species, the actual number in the ecosystem may be more or less. Each species is evolved in its own population and adapts to the environment through the repeated application of an EA. Although, in principle, any EA appropriate to the task at hand can be used, we only have firsthand experience constructing coevolutionary versions of genetic algorithms and evolution strategies. The figure shows the fitness evaluation phase of the EA from the perspective of each of the three species. Although most of our implementations of this model have utilized a sequential pattern of evaluation, where the complete population of each species is evaluated in turn, the species could also be evaluated in parallel. To evaluate individuals from one species, collaborations are formed with representatives from each of the other species.

There are many possible methods for choosing representatives with which to collaborate. In some cases it is appropriate to simply let the current best individual from each species be the representative. In other cases this is too greedy and alternative strategies are preferable. For example, a sample of individuals from each species could be chosen randomly, or a more ecological approach in which representatives are chosen non-deterministically based on their fitness could be used. Alternatively, a topology could be introduced and individuals who share a neighborhood allowed to collaborate.

3.1 Examples

Some simple examples may help illustrate the architecture. One of our first uses of cooperative coevolution was to maximize a function $f(\vec{x})$ of n independent variables (Potter and De Jong, 1994). The problem was hand-decomposed into n species and each assigned to one of the independent variables. Each species consisted of a population of alternative values for its assigned variable. To evaluate an individual from one of the species, we first selected the current best individual from every one of the other species and combined them, along with the individual being evaluated, into a vector of variable values. This vector was then applied to the target function. An individual was rewarded based on how well it maximized the function within the context of the variable values selected from the other species.

As a second example, cooperative coevolution was used to develop a rule-based control system for a simulated autonomous robot (Potter et al., 1995). There were two species, each consisting of a population of rule sets for a class of behaviors. We wanted one species to evolve behaviors appropriate for beating a hand-coded robot to food pellets randomly placed in the environment and the other to evolve behaviors appropriate for the time spent waiting for food to appear. The species were seeded with some initial knowledge in their assigned area of expertise to encourage evolution to proceed along the desired trajectory. To evaluate an individual from one of the species, we selected the current best rule set from the other species, merged this rule set with the one being evaluated, and used the resulting superset to control the robot. An individual was rewarded based on how well its rule set complemented the rule set selected from the other species.

3.2 Problem Decomposition

In the previous two examples, it was known how many species were required for the task at hand and what role each should play. We took advantage of this knowledge to produce human-engineered decompositions of the problems. In other situations, it is quite possible that we may have little or no prior knowledge to help us make this determination. Ideally, we would like both the number of species in the ecosystem and the roles the species assume to be an emergent property of cooperative coevolution.

One possible algorithm for achieving this is based on the premise that if evolution stagnates, it may be that there are too few species in the ecosystem from which to construct a good solution. Therefore, when stagnation is detected, a new species is added to the ecosystem. We initialize the species randomly and evaluate its individuals based only on the overall fitness of the ecosystem. Since we do not bias these new individuals, the initial evolutionary cycles of the species will be spent searching for an appropriate niche in which it can make a useful contribution—that is, the role of each new species will be emergent. Once a species finds a niche where it can make a contribution, it will tend to exploit this area. The better adapted a species becomes, the less likely it will be that some other species will evolve individuals that perform the same function because they will receive no reward for doing so. Conversely, if a species is unproductive, determined by the contribution its individuals make to the collaborations they participate in, the species will be destroyed. Stagnation can be detected by monitoring the quality of the collaborations through the application of the inequality

$$f(t) - f(t - L) < G \tag{1}$$

where $f(t)$ is the fitness of the best collaboration at time t, G is a constant specifying the fitness gain considered to be a significant improvement, and L is a constant specifying the length of an evolutionary window in which significant improvement must be made.

3.3 Other Characteristics of the Architecture

Evolving genetically isolated species in separate populations is a simple solution to the problem of maintaining sufficient diversity to support coadapted subcomponents. While there is evolutionary pressure for a species population to converge once it finds a useful niche, no such pressure exists for the various species to converge to the *same* niche. Rather, rewarding individuals based on how well they collaborate with representatives from the other species provides evolutionary pressure for them to make a unique contribution to the problem-solving effort. This ensures that the ecosystem will consist of a diverse collection of species.

Evolving species in separate populations also eliminates destructive cross-species mating. When individuals become highly specialized, mixing their genetic material through cross-species mating will usually produce non-viable offspring. As is demonstrated in nature, the genetic distance between two species is highly correlated with mating discrimination and the likelihood that if interspecies mating does occur the offspring will either not survive or be sterile (Smith, 1989).

Of course, the existence of separate breeding populations does not preclude interaction between species. The architecture handles interdependencies between subcomponents by evolving the species in parallel and evaluating them within the context of each other. This requires that individuals from different species be merged within a shared domain model to form a composite solution to the target problem.

Credit assignment occurs at two levels of abstraction. When evaluating the individuals within a species, the representatives from the other species remain fixed. Therefore, the fitness differential that is used in making reproduction decisions is strictly a function of the individual's relative contribution to the problem-solving effort within the context of the other species. The fitness is assigned only to the individual being evaluated, not shared with the representatives from the other species. This greatly simplifies the credit assignment problem because there is no need to determine which species contributed what. On the other hand, we do need to occasionally estimate the level of contribution a species makes to determine whether it should be allowed to survive. This can often be accomplished by computing the fitness differential of a solution with and without the participation of the species in question.

Each species is evolved by its own EA. Communication between species is limited to an occasional broadcast of representatives, and the only global control is that required to create new species and eliminate unproductive ones. This makes parallel implementations, in which species are assigned to separate processors, trivial.

Finally, heterogeneous representations are supported. This will become increasingly important as we apply evolutionary computation to larger problems. For example, in developing a control system for an autonomous robot, some components may best be implemented as artificial neural networks, others as collections of symbolic rules, and still others as parameter vectors. The cooperative coevolution architecture enables each of these components to be represented appropriately and evolved with a suitable class of EA.

4 Analysis of Decomposition Capability

In the following empirical analysis we explore whether the algorithm outlined in Section 3.2 is capable of producing good problem decompositions purely as a result of evolutionary pressure to increase the overall fitness of the ecosystem. We will describe four studies—each designed to answer one of the following questions concerning the ability of this algorithm to decompose problems:

- Will species locate and cover multiple environmental niches?
- Will species evolve to an appropriate level of generality?
- Will adaptation occur as the number and role of species change?
- Will an appropriate number of species emerge?

This is, of course, an inductive argument. Although positive answers to these questions may not guarantee that good problem decompositions will emerge, we would certainly expect positive answers if this capability does, in fact, exist.

4.1 Coevolving String Covers

In these studies we use binary string covering as a target problem, in part because the task provides a relatively simple environment in which the emergent decomposition properties of cooperative coevolution can be explored, and because it has been used by others in related studies (Forrest et al., 1993). The problem consists of finding a set of N binary vectors that match as strongly as possible another set of K binary vectors, where K is typically much larger than N. We call these sets the *match set* and *target set* respectively. Given that K is larger than N, the match set must contain patterns shared by multiple target strings to cover the target set optimally, that is, the match set must *generalize*. The match strength S between two binary vectors $\vec{x}$ and $\vec{y}$ of length L is determined simply by summing the number of bits in the same position with the same value as follows:

$$S(\vec{x}, \vec{y}) = \sum_{i=1}^{L} \begin{cases} 1 & \text{if } x_i = y_i \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

To compute the strength of a match set M, we apply it to the target set and average the maximum computed match strengths with respect to each target set element as follows:

$$S(M) = \frac{1}{K} \sum_{i=1}^{K} \max(S(\vec{m}_1, \vec{t}_i), \ldots, S(\vec{m}_N, \vec{t}_i)) \quad (3)$$

where $\vec{m}$ and $\vec{t}$ are elements of the match set and target set respectively.

The string covering problem can be easily mapped to the model of cooperative coevolution. Recall from Figure 1 that to evaluate an individual from one species it must collaborate with representatives from each of the other species in the ecosystem. Here, individuals represent match strings, and each species will contribute one string to the match set. Representatives are chosen as the current best string from each of the other species. In other words, a match set of size N will consist of a string being evaluated from one species and the current best string from each of the other $N - 1$ species. During the initial few generations, a species may participate in collaborations before its population has ever been evaluated. In this case a random representative is chosen. The fitness of the match set will be computed from Equation 3 and assigned to the string being evaluated.

Problem decomposition, in this context, consists of determining the size of the match set and the coverage of each match set element with respect to the target strings. In the first three studies the size of the match set is predetermined. However, given that the species are randomly initialized, the coverage of each will be an emergent property of the system in all four studies.

The particular EA used in these studies is a GA. In all cases we initialize the population of each of the species randomly, use a population size of 50, two-point crossover at a rate of 0.6, bit-flipping mutation at a rate set to the reciprocal of the chromosome length, and proportionate selection based on scaled fitness.

4.2 Experimental Results

4.2.1 Locating and Covering Multiple Environmental Niches

One of the most fundamental questions concerning emergent problem decomposition is whether the method can locate and cover multiple environmental niches. In string covering, the niches we are interested in are schemata common among the target strings. By *schemata* we are referring to string templates consisting of a fixed binary part and a variable part designated by the symbol "#". In a previous study by Forrest et al. (1993), an experiment was performed demonstrating the ability of a traditional single-population GA to detect common schemata in a large collection of target strings. To compute the match strength between two strings, they used a linear function similar to Equation 2. Their schema detection experiment is duplicated here, with cooperative coevolution substituted for their traditional single-population GA.

The experiment consists of evolving match sets for three separate target sets, each consisting of 200 64-bit strings. The strings in the first target set will be generated in equal proportion from the following two half-length schemata:

```
11111111111111111111111111111111################################
################################11111111111111111111111111111111
```

The variable half of each of the strings will consist of random patterns of ones and zeros. Similarly, the strings in the second target set will be generated in equal proportion from the following quarter-length schemata:

```
1111111111111111################################################
################1111111111111111################################
################################1111111111111111################
################################################1111111111111111
```

and the strings in the third target set will be generated in equal proportion from the following eighth-length schemata:

```
11111111########################################################
########11111111################################################
################11111111########################################
########################11111111################################
################################11111111########################
########################################11111111################
################################################11111111########
########################################################11111111
```

The niches in the target set generated from the eighth-length schemata should be significantly harder to find than those generated from the half-length or quarter-length schemata because the fixed regions that define the eighth-length niches are smaller with respect to the variable region of the strings.

We know *a priori* how many niches exist and are only interested in whether we can locate and cover them, so we simply evolve an equal number of species as niches. For example, since we know that the first target set was generated from two schemata and, therefore, will contain two niches, we evolve two species to cover these niches. Similarly, four species are evolved to cover the second target set, and eight species are evolved to cover the third target set.

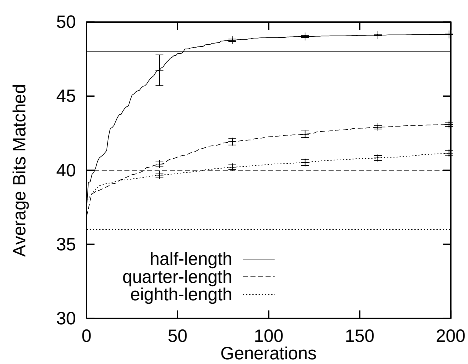

Figure 2: Finding half-length, quarter-length, and eighth-length schemata.

The average number of bits matched per target string using a match set consisting of the best individual from each species is shown in Figure 2. Each curve in the figure was computed from the average of five runs of 200 generations using the indicated target set. Overlaid on the curves at increments of 40 generations are 95-percent confidence intervals. The horizontal lines in the graph represent the expected match values produced from the best possible single-string generalist. Given the half-length, quarter-length, and eight-length schemata shown, this generalist will consist entirely of ones, and its average match scores for the three target sets will be 48, 40, and 36 respectively. The Forrest study demonstrated that a traditional single-population GA consistently evolves this best possible single-string generalist. Our study shows that when multiple species collaborate, they are able to cover the target set better than any single individual evolved with a traditional GA. Furthermore, when more species are employed, as in the eighth-length schema experiment, the amount of improvement over the traditional GA increases.

```
Half-length

  Species 1:  1111111111111111111111111111111110100110001000111111100010110111
  Species 2:  0010000001001110110100001000100011111111111111111111111111111111

Quarter-length

  Species 1:  1111111111111111000110011001010011111111111111110111111111110111
  Species 2:  1111111110111111111111101011111111111111111111111000010100010000
  Species 3:  0101010101101101111111011111110100001010001011101111111111111111
  Species 4:  1001101110011000111111111111111110110100010100011010101101101111

Eighth-length

  Species 1:  1100100011111110001011110100000011001011111111110011010011110010
  Species 2:  1011111001010001111111110010101110010000101101111110111011010010
  Species 3:  1010111100000111111011110010000111001100111101111011111100000111
  Species 4:  0000111011111111101110010111101011111111010001101100010111111101
  Species 5:  1101100100100010110000111100101111111111100101011001000111111111
  Species 6:  0001011011111011011010001111111100110011111111101111111100110000
  Species 7:  1111001011111110100000101011010100100100100100010111000111111111
  Species 8:  1111111110001101111000001111111101011010110111101100001111101010
```

Figure 3: Final species representatives from schemata experiments.

The reason for this improvement can be seen in Figure 3. This figure shows the best individual from each of the species at the end of the final generation of the first of five runs from the half-length, quarter-length, and eighth-length schemata experiments. Substrings perfectly matching the fixed regions of the schemata are highlighted. A couple of observations can be made from this figure. First, clearly, each species focuses on one or two niches and relies on the other species to cover the remaining ones. This enables the species to cover their respective target strings better than if they had to generalize over the entire target set. Stated another way, each species locates one or two niches where it can make a useful contribution to the collaborations that are formed. This is strong evidence that the species have a cooperative relationship with one another. Second, two or more species may occasionally occupy a common niche. See, for example, the fourth and fifth species from the eighth-length schemata experiment. Although our model of cooperative coevolution does not exclude this possibility, each species must make some unique contribution to be considered viable. Third, some of the species, for example, the third species from the eighth-length schemata experiment, make no obvious contribution. It may be that this species has found a pattern that repeatedly occurs in the random region of some of the target strings, and its contribution is simply not readily visible to us. Another possibility is that this particular species is genuinely making no useful contribution and should be eliminated.

4.2.2 Evolving to an Appropriate Level of Generality

To determine whether species will evolve to an appropriate level of generality, we generate a target set from three 32-bit test patterns and vary the number of species from one to four. Given three patterns and three or more species, each species should be able to specialize on one pattern. However, given fewer species than patterns, each must generalize to cover the patterns as best as possible. As with the previous study, this study was inspired by experiments done on emergent fitness sharing by Forrest et al. (1993).

Our test patterns are shortened complements of the ones used by Forrest as follows:

```
11111111111111111111111111111111
11111111110000000000000000000000
00000000000000000000001111111111
```

The optimal single-string cover of these patterns is the following string:

```
11111111110000000000001111111111
```

which produces an average match score of $(20+22+22)/3 = 21.33$. The optimal two-string cover of the three patterns is a string consisting of all ones and a string whose 12-bit middle segment is all zeros. A cover composed of these two strings will produce an average match score of 25.33. For example, the following two strings:

```
11111111111111111111111111111111
10010110110000000000001111110101
```

are scored as follows: $(32 + 20 + 24)/3 = 25.33$. The makeup of the extreme left and right 10-bit segments of the second string is unimportant. The optimal three-string cover of the patterns is obviously the patterns themselves.

To build on Section 4.2.1, we hid the three 32-bit test patterns by embedding them in the following three schemata of length 64:

```
1##1###1###11111##1##1111#1##1###1#1111##111111##1#11#1#11######
1##1###1###11111##1##1000#0##0###0#0000##000000##0#00#0#00######
0##0###0###00000##0##0000#0##0###0#0000##001111##1#11#1#11######
```

A target set composed of 30 strings was then generated in equal proportion from the schemata.

Four sets of five runs were performed. In the first set we evolved a cover for the target set using a single species, which is equivalent to using a traditional single-population GA. The remaining three sets of runs include evolving two, three, and four species respectively. The plots in Figure 4 show the number of target bits matched by the best individual from each species. They were generated from the first run of each set rather than the average of the five runs so that the instability that occurs during the early generations was not masked. However, we verified that all five runs of each set produced similar results. Although the

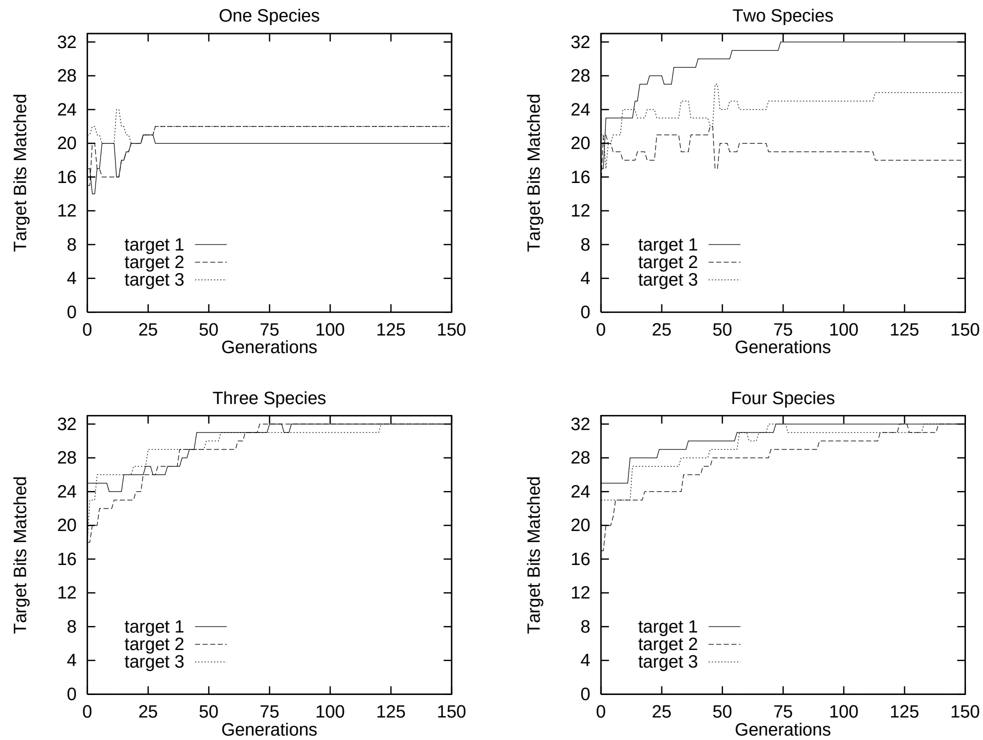

Figure 4: Covering three hidden niches with one, two, three, and four species.

figure shows just the number of bits matching the 32-bit target patterns, the fitness of individuals was based on how well all 64 bits of each target string were matched.

In the single species plot, we see that after an initial period of instability the species stabilizes at the appropriate level of generality. Specifically, 20 bits of the first pattern are matched, and 22 bits of the second and third patterns are matched. This result is consistent with the optimal single-string generalist described previously. When we increase the number of species to two, the first pattern is matched perfectly and the other two patterns are matched at the level of 26 and 18 bits respectively—consistent with the optimal two-string generalization. When three species are evolved, all three patterns are matched perfectly, indicating that each species has specialized on a different test pattern. In the four species plot, we see that when the number of species is increased beyond the number of niches, perfect matches are also achieved; however, more fitness evaluations are required to achieve this level of performance.

Figure 5 shows the best individual from each of the species at the end of the final generation. After removing all the bits corresponding to the variable regions of the target strings, the patterns that remain are the optimal one, two, and three element covers described earlier, which is conclusive evidence that the species have evolved to appropriate levels of generality. One can also see from this figure that the third and fourth species have focused on the same 32-bit target pattern. However, the bits from these two individuals corresponding to the variable regions of the target strings are quite different. What is occurring is that the two species are adapting to different repeating patterns in the variable regions of the target

One-species experiment

Species 1: 1001010100011111001001000101101010100001100111101111101011100110

Noise removed
11111111110000000000001111111111

Two-species experiment

Species 1: 1011000100011111001001111111111011011111111111101111101111110110
Species 2: 0100110000000110000010000101101010100000000100101111010011000110

Noise removed
11111111111111111111111111111111
00000110000000000000001001110011

Three-species experiment

Species 1: 1001000100011111001001111111111111111111111111111111101111111100
Species 2: 0100010000000000000000100000010011101000001001111011011010110000010
Species 3: 1001100100011111011001000101101000100000100000001010010100110010

Noise removed
11111111111111111111111111111111
00000000000000000000001111111111
11111111110000000000000000000000

Four-species experiment

Species 1: 1001000100011111001001111111111111111111111111111111101111111100
Species 2: 0100010000000000000000100000010011101000001001111011011010110000010
Species 3: 1101011101111111101001000101101000100001100000001000000100101100
Species 4: 1011000101011111011101000101000000000001000000000010000100011000

Noise removed
11111111111111111111111111111111
00000000000000000000001111111111
11111111110000000000000000000000
11111111110000000000000000000000

Figure 5: Final representatives from one through four species experiments before and after the removal bits corresponding to variable target regions.

strings. This enabled the ecosystem with four species to achieve a slightly higher match score on the full 64-bit target strings than the ecosystem with three species. To determine whether this difference is significant, an additional 95 runs were performed using the three- and four-species ecosystems to bring the total to 100 runs apiece. The arithmetic means from the two sets of runs were 51.037 and 51.258 respectively, and a t-test verified that this difference is unlikely to have occurred by chance.

4.2.3 Adaptation as the Number and Role of Species Change

If we do not know *a priori* how many species are required and must dynamically create them as evolution progresses, existing species must be able to adapt to these new species. One of the forces driving this adaptation is the freedom of older species to become more specialized as new species begin making useful contributions.

Specializing in this manner is similar to the notion of *character displacement* that occurs

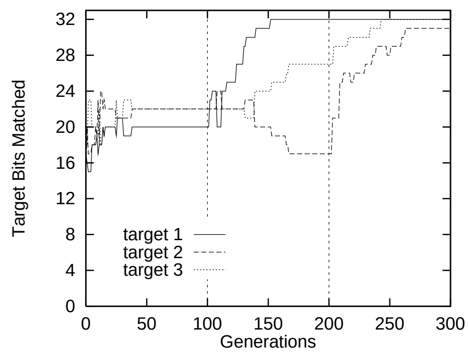

Figure 6: Shifting from generalists to specialists as new species are added to the ecosystem on a fixed schedule.

in natural ecosystems (Brown and Wilson, 1956). For example, a study of finches in the Galápagos by Lack (1947) determined that on the eight islands occupied by both *Geospiza fortis* and *Geospiza fuliginosa*, the average depth of the *G. fortis* beak was approximately 12 mm while the average depth of the *G. fuliginosa* beak was approximately 8 mm. However, on the islands Daphne, occupied only by *G. fortis*, and Crossman, occupied only by *G. fuliginosa*, the average beak depth of both species was approximately 10 mm. The interpretation of this observation is that when both finch species compete for food within the same ecosystem, their beaks evolve to become specialized to either a larger or smaller variety of seeds. However, when only one of these two species occupies an ecosystem, the species evolves a more general purpose beak suitable for consuming a wider variety of seeds.

To determine whether the species in our model of cooperative coevolution will mimic this process and become more specialized as new species are introduced into their ecosystem, we begin this experiment with a single species, add a second species at generation 100, and add a third species at generation 200. A 30-element target set was generated from the same three schemata used in Section 4.2.2.

The number of target bits from the fixed region of the schemata matched by the best individual from each species is shown in Figure 6. As before, although only the performance on the 32-bit fixed region is shown, the fitness of individuals was based on how well the entire 64 bits of each string was covered. Each dashed, vertical line marks the creation of a new species. A period of instability just after each species is introduced is evidence of a few quick role changes as the species "decide" which niche they will occupy. However, the roles of the species stabilize after they evolve for a few generations. It is clear from the figure that when the second species is introduced, one of the species specializes on the strings containing the first target pattern, while the other species generalizes to the strings containing the other two target patterns. Similarly, when the third species is introduced, all three species are able to become specialists. Furthermore, by comparing Figure 6 with the plots in Figure 4, we see that this adaptation results in the same levels of generalization and specialization that are achieved when the species are evolved together from the beginning.

4.2.4 Evolving an Appropriate Number of Species

It is important not to evolve too many species because each requires computational resources to support 1) the increasing number of fitness evaluations that need to be performed, 2) the need for applying operators such as crossover and mutation to more individuals, and 3) the conversion between genotypic and higher-level representations that may be required. On the other hand, if we evolve too few species they will be forced to be very general—resulting in mediocre covers as we saw in the previous few studies.

An algorithm for evolving an appropriate number of species was introduced in Section 3.2. To test the effectiveness of this algorithm, we use the same target set as in the previous two studies and begin by evolving a single species. At each ecosystem generation[1] we check for evolutionary stagnation, and if we are not making sufficient improvement, we add a new randomly initialized species and destroy those that are not making a significant contribution. Precisely what constitutes stagnation is application dependent, but here we determined through experimentation that if the current-best fitness as computed from Equation 3 does not improve by at least 0.5 over five generations, further significant improvement without the addition of a new species is unlikely. Regarding species destruction, we consider that a species contributes to the fitness of a collaboration when its representative matches at least one of the target strings better than any other member of the collaboration, with ties being won by the older species. We refer to the amount of contribution that a species must make to be considered viable as its *extinction threshold*. We set the extinction threshold here to 5.0 to eliminate species that gain an advantage by matching spurious shared patterns in the random regions of the target strings. If we had been interested in these less significant patterns, we could have set the extinction threshold to a smaller value—perhaps just slightly above zero—and the species matching these patterns would be preserved.

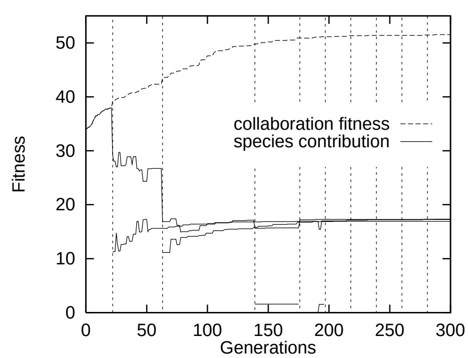

Figure 7: Changing contributions as species are dynamically created and eliminated from the ecosystem.

The fitness of the best collaboration and the amount of fitness contributed by each species in the ecosystem over 300 generations is plotted in Figure 7. The vertical, dashed lines represent stagnation events in which unproductive species are eliminated, and a new species is created. For the first 21 generations, the ecosystem consists of only one species; therefore, its contribution during this period is equal to the collaboration fitness. At generation 22 stagnation is detected, and a second species is added to the ecosystem. Although the contribution of the first species now drops from 37.9 to 29.1, the contribution of the new species causes the collaboration fitness to increase from 37.9 to 39.2. At generation 63 the evolution of these two species has stagnated, and we add a third species that

[1] By *ecosystem generation* we mean that all species are evolved for a generation.

focuses on some of the strings being matched by the first species—causing the contribution of the first species to drop to 16.8. However, the third species contributes 11.1, and the fitness of the collaboration jumps from 42.3 to 43.5. At generation 138 evolution has stagnated with three species in the ecosystem, the collaboration fitness is 49.7, and the species are contributing 17.1, 16.8, and 15.8 respectively. We know from the previous few studies that this is the optimal number of species for this particular problem; however, the system does not possess this knowledge and creates a new species. This species is only able to contribute 1.6 to the fitness of the collaboration, which is less than the extinction threshold, and, therefore, it is eliminated at generation 176. At this point another species is created, but it does not begin making a contribution until generation 192. Since the most this new species contributes is 1.5, it is eliminated at generation 197, and another new species is created. From this point until the end of the run, none of the new species ever makes a non-zero contribution, and each is eliminated in turn when stagnation is detected.

The first observation that can be made from this experiment is that an appropriate number of species emerges. Specifically, the ecosystem stabilizes in a state in which there are three species making significant contributions and a fourth exploratory species providing insurance against the contingency of a significant change in the environment. Our simple string covering application has a stationary objective function, so this insurance is not really necessary, but this would often not be the case in "real-world" problems. The second observation is that although the fourth and fifth species were eventually eliminated, they were able to make small contributions by matching patterns in the random regions of a few targets rather than generalizing over a large number of them. If we were interested in maintaining these specialists in the ecosystem we could lower the extinction threshold, and they would survive.

5 Case Study in Emergent Problem Decomposition

Moving beyond the simple studies of the previous section, we now describe a more complex case study in which cooperative coevolution is applied to the construction of an artificial neural network. Our task will be to construct a multilayered feed-forward artificial neural network that when presented with an input pattern, will produce some desired output signal. We are given the basic topology of the network, but we know neither the number of neural units required to adequately produce the desired output nor the role each unit should play—the determination of which constitutes our decomposition task. The primary focus of this case study is to directly compare and contrast the decompositions produced by cooperative coevolution to those produced by a non-evolutionary approach that is specific to the construction of artificial neural networks.

5.1 Evolving Cascade Networks

The topology evolved in this case study is a *cascade network*. In cascade networks, all input units have direct connections to all hidden units and to all output units, the hidden units are ordered, and each hidden unit sends its output to all downstream hidden units and to all output units. Cascade networks were originally used in conjunction with the cascade-correlation learning architecture (Fahlman and Lebiere, 1990).

Cascade network evolution can be easily mapped to the cooperative coevolution architecture by assigning one species to evolve the weights on the connections leading into the output units and assigning each of the other species to evolve the connection weights for one

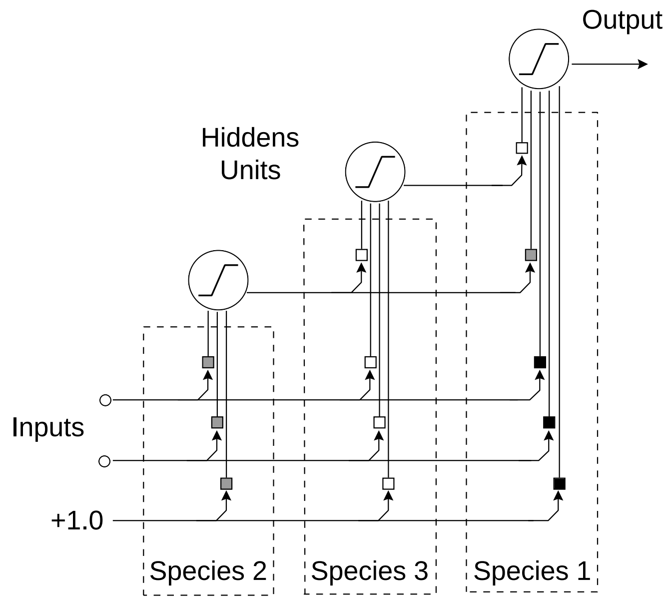

Figure 8: Mapping between cascade network and species.

of the hidden units.[2] For example, the species to network mapping for a cascade network having two inputs, a bias signal of 1.0, two hidden units, and one output is shown in Figure 8. Each of the small black, gray, and white boxes in the figure represents a connection weight being evolved. To add a new hidden unit to the network we simply create a new species for that unit and add a new randomly initialized weight to each of the individuals of the species assigned to the output unit. The fitness of an individual is determined by combining it with the current best individual from each of the other species to produce a connection weight specification for the entire network. The sum-squared error is then computed as preclassified training patterns are fed forward through the network. Individuals producing networks with lower sum-squared errors are considered more highly fit.

The network shown in Figure 8 is constructed incrementally as follows. When the evolution of the network begins, there is only one species in the ecosystem, and its individuals represent alternatives for the output connection weights denoted by the three black boxes. Later in the network's evolution, the first hidden unit is added, and a second species is created to represent the new unit's input connection weights. In addition, a new connection weight is added to each individual of the first species. All of these new weights are denoted by gray boxes in the figure. The species creation event is triggered by evolutionary stagnation as described earlier. Later still, evolution again stagnates and the second hidden unit is added, a third species is created to represent the unit's connection weights, and the individuals of the first species are further lengthened. These connection weights are denoted by white boxes. This cycle continues until a network is created that produces a sufficiently low sum-squared error. Using this scheme, a cascade network with k hidden units will be constructed from $k + 1$ species. Alternatively, we could have started with a liberal estimation of the number of hidden units required and let the destruction of species pare the network down to an

[2] Although we have not done so, other network topologies could also be easily mapped to the cooperative coevolution architecture.

appropriate size, but this option was not explored.

Due to the complexity of the neural network search space, when adding a new species, we found it helpful to temporarily focus computational resources on enabling it to find a niche in which it could make a contribution. This is accomplished simply by evolving just the new species and the species representing the weights on the connections to the output unit until progress again approaches an asymptote. At this point, we eliminate the new species if it is not making a significant contribution and create another one to replace it, or we continue with the evolution of *all* of the species in the ecosystem.

The EA used in this case study is a (μ, λ) evolution strategy (ES) as described by Schwefel (1995), with $\mu = 10$ and $\lambda = 100$. We previously applied a GA to this task (Potter and De Jong, 1995) but achieved better results with the ES. Each individual consists of two real-valued vectors: a vector of connection weights and a vector of standard deviations used by the mutation operator. We require the standard deviations to always be greater than 0.01 and they are adapted as follows:

$$\vec{\sigma}_{t+1} = \vec{\sigma}_t e^{Gauss(0,\sigma')} \tag{4}$$

where t denotes time expressed discretely in generations. The rate of convergence of the ES is sensitive to the choices of σ' and the initial setting of the standard-deviation vectors $\vec{\sigma}$. We follow Schwefel's recommendation and set σ' using the equation

$$\sigma' = \frac{C}{\sqrt{|\vec{\sigma}|}} \tag{5}$$

where C depends on μ and λ. Given the $(10, 100)$ ES used here, C is set to 1.0. Schwefel also recommends initializing $\vec{\sigma}$ using the equation

$$\sigma_k = \frac{R_k}{\sqrt{|\vec{\sigma}|}} \qquad \text{for } k = 1, 2, \ldots, |\vec{\sigma}| \tag{6}$$

where the constant R_k is the maximum uncertainty range of the corresponding variable. Given that our randomly initialized connection weights are limited to the range $(-10.0, 10.0)$, each R_k is correspondingly set to 20.0. Mutation is the only evolutionary operator used.

5.2 The Cascade-Correlation Approach to Decomposition

In the context of a cascade network, problem decomposition consists of determining the number of hidden units and their roles. We will be comparing and contrasting the decompositions produced by cooperative coevolution to those produced by a second-order gradient-descent technique for constructing cascade networks called the cascade-correlation learning architecture (Fahlman and Lebiere, 1990).

Prior to the development of the cascade-correlation learning architecture, feed-forward networks were constructed by using rules-of-thumb to determine the number of hidden layers, units, and their connectivity. The roles of the hidden units were allowed to emerge through the application of the backpropagation algorithm—a first-order gradient-descent technique (Rumelhart et al., 1986). A source of inefficiency in this process is the considerable time it takes for the emergence of roles to stabilize (Fahlman and Lebiere, 1990).

Cascade-correlation was specifically designed to eliminate this inefficiency by constructing the network one hidden unit at a time and freezing the roles of the hidden units once established. Rather than simply allowing these roles to emerge, each hidden unit is trained to respond either positively or negatively to the largest portion of remaining error signal using gradient-descent to adjust its input connection weights[3]. The gradient is with respect to the magnitude of the correlation between the output from the hidden unit and the sum-squared error as training patterns are fed forward through the network. After training, the hidden unit will only fire when the most problematic patterns from the training set are presented to the network—forcing the hidden unit to focus on a specific region of the input space. Once the input connection weights of a hidden unit are trained and frozen, all output connection weights are trained by descending the sum-squared network error gradient. The addition of a new hidden unit is triggered when the reduction of the sum-squared network error approaches an asymptote. This cycle of adding a new hidden unit, training and freezing its input connection weights, and training all output connection weights continues until a sufficiently low sum-squared error is produced.

5.3 Two-Spirals Problem

We will construct cascade networks to solve the two-spirals problem originally proposed in a post to the *connectionists* mailing list by Alexis Wieland. This problem is a classification task that consists of deciding in which of two interlocking spiral-shaped regions a given (x, y) coordinate lies. The interlocking spiral shapes were chosen for this problem because they are not linearly separable. Finding a neural network solution to the two-spirals problem has proven to be very difficult when using a traditional gradient-descent learning method such as backpropagation, and, therefore, it has been used in a number of studies to test new learning methods (Lang and Witbrock, 1988; Fahlman and Lebiere, 1990; Whitley and Karunanithi, 1991; Suewatanakul and Himmelblau, 1992; Potter, 1992; Karunanithi et al., 1992).

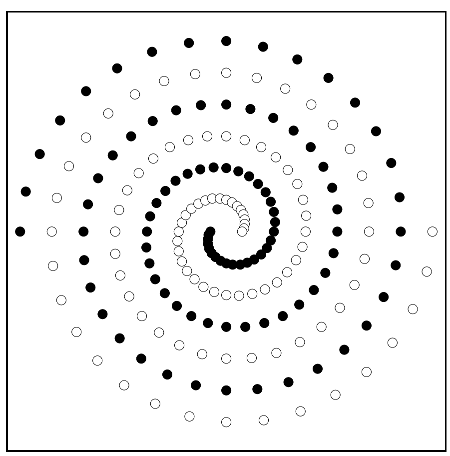

Figure 9: Training set for the two-spirals problem.

To learn to solve this task, we are given a training set consisting of 194 preclassified

[3]More accurately, a small number of *candidate units* are created and trained in parallel; however, for the sake of clarity, we ignore that detail in this description.

coordinates as shown in Figure 9. Half of the coordinates are located in one spiral-shaped region and marked with black circles, and half are in the other spiral-shaped region and marked with white circles. The coordinates of the 97 black circles are generated using the following equations:

$$r = \frac{6.5(104-i)}{104} \tag{7}$$

$$\theta = i\frac{\pi}{16} \tag{8}$$

$$x = r\sin\theta \tag{9}$$

$$y = r\cos\theta \tag{10}$$

where $i = 0, 1, \ldots, 96$. The coordinates of the white circles are generated simply by negating the coordinates of the black circles.

When performing a correct classification, the neural network takes two inputs corresponding to an (x, y) coordinate and produces a positive signal if the point falls within the black spiral and a negative signal if the point falls within the white spiral.

5.4 Experimental Results

Ten runs were performed using cascade-correlation, and ten runs were performed using cooperative coevolution. Runs were terminated when either all 194 training patterns were classified correctly, or it was subjectively determined that learning was stuck at a local optimum. Cascade-correlation generated networks capable of correctly classifying all the training patterns in all ten runs, while cooperative coevolution was successful in seven out of ten runs. We will address the three unsuccessful runs at the end of this section.

Table 1: Required number of hidden units.

Method	*Hidden units*		
	Mean	*Max*	*Min*
Cascade-correlation	16.80 ± 1.16	19	14
Cooperative coevolution	13.71 ± 2.18	18	12

Table 1 shows the average number of hidden units produced by each method, along with 95-percent confidence intervals on the mean, and the maximum and minimum number of hidden nodes generated. Cascade-correlation networks required an average of 16.8 hidden units, which is consistent with results reported by Fahlman and Lebiere (1990). In contrast, in its seven successful runs, cooperative coevolution networks required an average of only 13.7 hidden units to perform the task. This represents a statistically significant difference in the number of hidden units required to solve the problem. The p-value produced from a t-test of the means was 0.015.

We begin our characterization of the roles played by the hidden units produced by the two methods by describing the reduction in misclassification and sum-squared error attributable to each unit. Table 2 was generated by starting with the final networks produced by the first runs of the two methods and eliminating one hidden unit at a time while measuring the number of training-set misclassifications and the sum-squared error.

Table 2: Effect of adding hidden units on training set classification.

Hidden units	*Misclassifications*		*Sum-squared error*	
	CasCorr	*CoopCoev*	*CasCorr*	*CoopCoev*
0	96	99	84.96	68.26
1	94	97	83.41	64.96
2	76	84	64.61	61.34
3	74	70	64.68	67.24
4	64	80	62.21	68.36
5	64	72	61.45	54.57
6	58	70	50.65	62.53
7	54	67	37.98	54.76
8	58	44	46.24	35.38
9	52	61	35.04	46.84
10	36	27	30.27	20.78
11	34	27	25.38	17.18
12	26	0	21.52	6.63
13	22		14.49	
14	16		8.87	
15	0		1.67	

The first run was chosen for this comparison arbitrarily; however, it appears to provide a reasonably fair comparison. Overall, we find the sequences from the two methods to be quite similar. One similarity is that neither the misclassification nor the sum-squared error sequences monotonically decrease; that is, both methods have created hidden units that, when looked at in isolation, make matters worse. These units presumably play more complex roles—perhaps working in conjunction with other hidden units. Another similarity is that the misclassification sequences of both methods are more erratic than the sum-squared error sequences; however, this is no surprise because neither method used misclassification information for training. The major difference between the methods is that the cooperative coevolution sequences tend to make bigger steps with higher variance and contain fewer elements.

As in Fahlman and Lebiere (1990), we gain further insight into the roles played by the hidden units by studying a series of field-response diagrams generated from the same networks summarized in Table 2. The field-response diagrams shown in Figure 10 were produced from the cascade-correlation network, and those shown in Figure 11 were produced from the network evolved with cooperative coevolution. The diagrams were generated by feeding the elements of a 256 x 256 grid of coordinates forward through the network and measuring the output signal produced both by individual hidden units and the entire network. Positive signals are displayed as black pixels, and negative signals are displayed as white pixels. For example, in Figure 10 the top-right pair of field response diagrams is generated from a cascade-correlation network in which all but the first two hidden units have been eliminated. The left diagram of that particular pair shows the output from the second hidden unit, and the right diagram of the pair shows the corresponding output from the network.

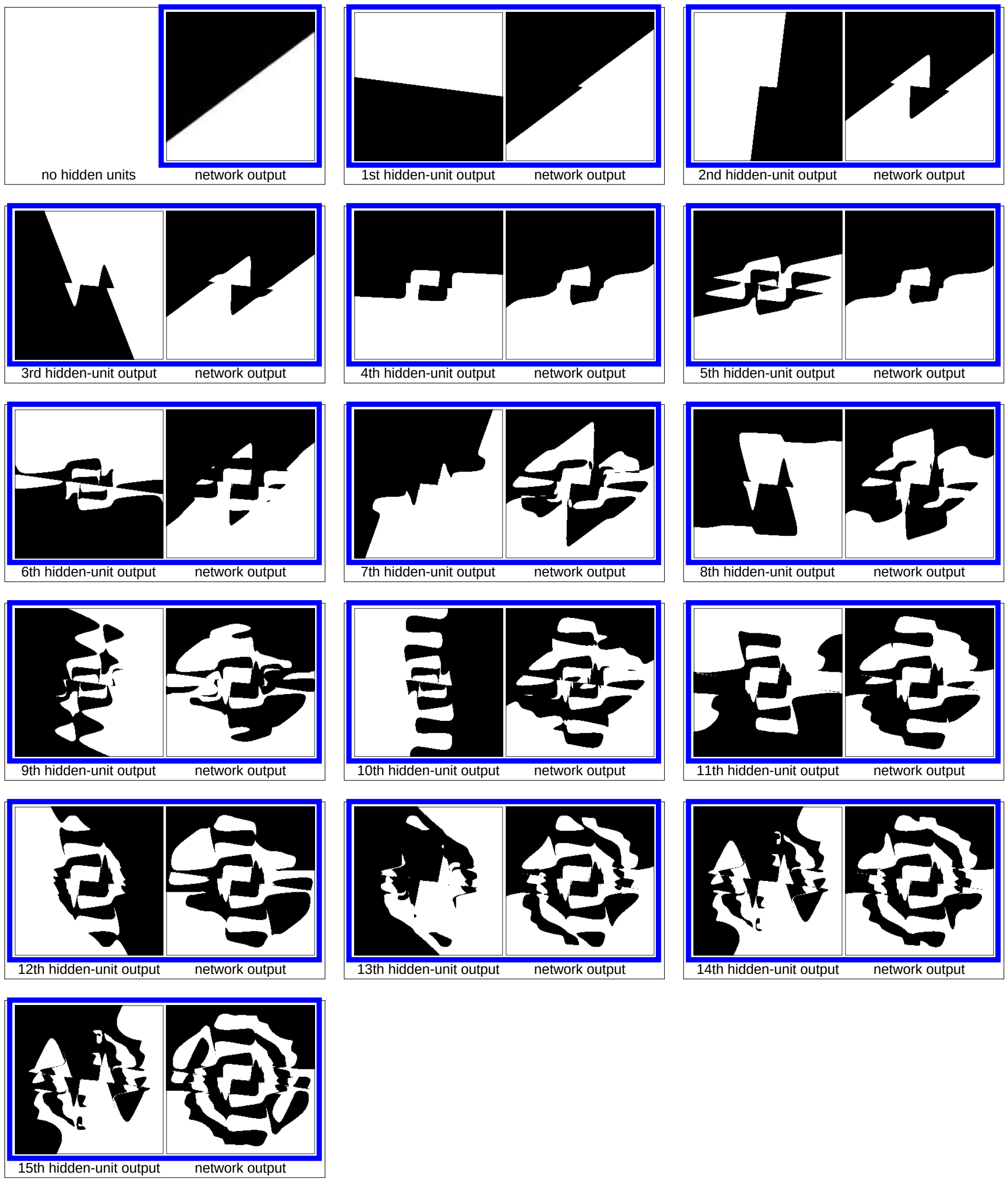

Figure 10: Field response diagrams generated from a neural network constructed using cascade correlation, showing the incremental effect of adding hidden units to the network.

We make a number of observations from a comparison of these two sets of figures. First, both the cascade-correlation decompositions and those produced by cooperative coevolution clearly exploit the symmetry inherent in the two-spirals problem. A second similarity is that the early hidden units focus on recognition in the center region of the field. This shows that both methods are exploiting the greater central concentration of training set elements, as one can see from Figure 9. A third similarity is that as hidden units are added to the network, their response patterns tend to become increasingly complex;

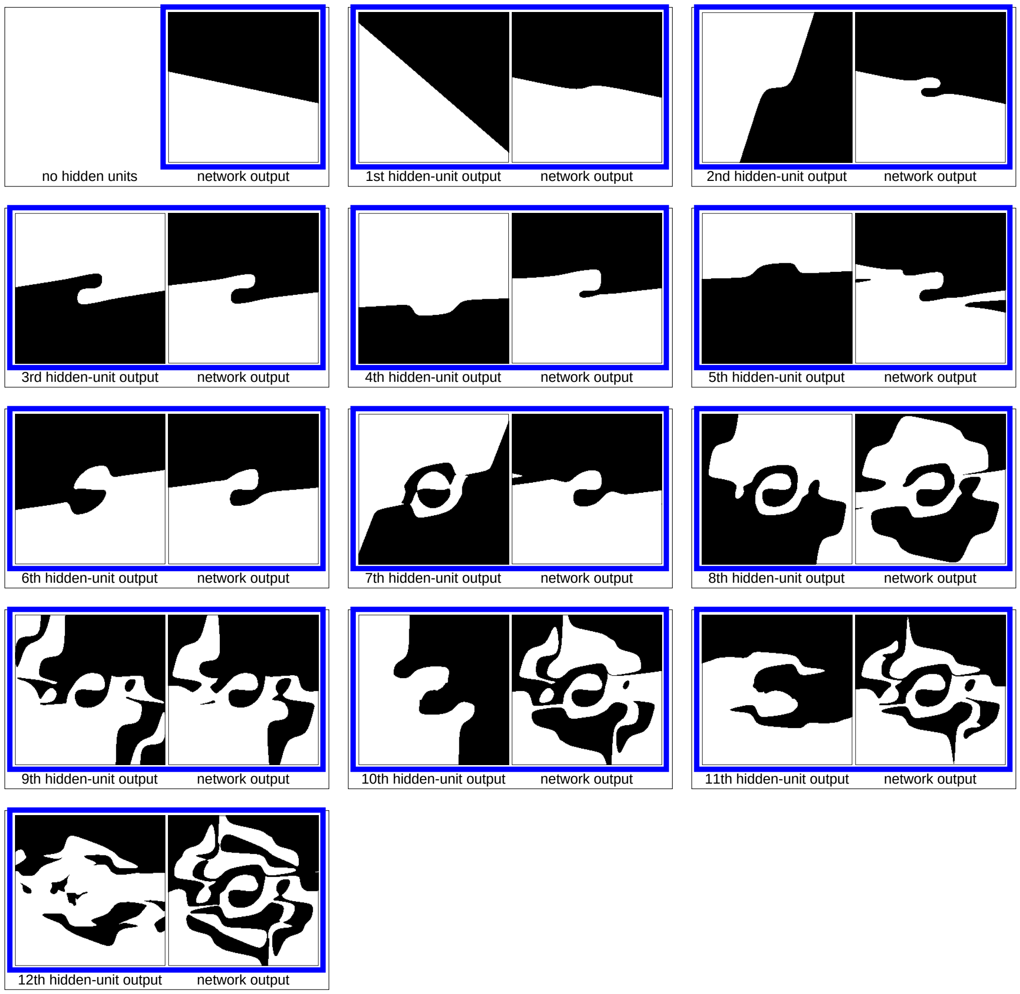

Figure 11: Field response diagrams generated from a neural network constructed using cooperative coevolution, showing the incremental effect of adding hidden units to the network.

although, this is less true with cooperative coevolution than with cascade-correlation. The increase in complexity may simply be a result of the network topology—the later hidden units have more inputs than the early hidden units as shown in Figure 8.

There are also a couple of noticeable differences between the two sets of figures. The cascade-correlation field-response diagrams tend to consist of angular-shaped regions, while the shapes in the diagrams produced by the network evolved with cooperative coevolution are more rounded. In addition, the cascade-correlation diagrams are visually more complex than the ones from cooperative coevolution. We hypothesize that these differences between the decompositions, as highlighted by the field-response diagrams, are due to the task-specific nature of the cascade-correlation decomposition technique. Recall that cascade-correlation uses the correlation between the output of a hidden unit and the network error signal to train the weights on the connections leading into the unit. This enables the hidden unit to respond precisely to the training patterns that are responsible for most of the error signal while ignoring the other training patterns. This is manifested in the field-response diagrams as complex angular regions. Since this particular instantiation of the cooperative coevolution architecture does not use task-specific statistical information as a focusing tool, it tends to paint with broader brush strokes.

Painting with broad brush strokes may be advantageous with respect to the ability to generalize, but it can be a disadvantage as well. In all three unsuccessful runs, after all but a few of the training patterns were correctly classified, the last species created was unable to find a niche in which it could contribute. Further investigation of the unsuccessful runs revealed that the sum-squared error generated by the few remaining misclassifications was being masked by the residual sum-squared error generated by all the other training patterns. Therefore, variation among individuals produced no selective advantage, and, hence, no further evolutionary progress occurred. This could probably be addressed by utilizing the correlation statistic in the fitness function of the species representing hidden units.

In summary, our approach to letting decompositions emerge purely as a result of evolutionary pressure to increase the overall fitness of the ecosystem works well in simple domains, and it even has the potential of producing good decompositions in domains as complex as the one described here. However, as our domains become increasingly complex we may need to bestow the fitness functions with a more specialized ability to steer the species towards niches in which they can make useful contributions.

6 Conclusions and Future Work

We believe that to apply evolutionary algorithms to the solution of increasingly complex problems, explicit notions of modularity must be introduced to provide reasonable opportunities for solutions to evolve in the form of interacting coadapted subcomponents. The difficulty comes in finding computational extensions to our current evolutionary paradigms in which such subcomponents "emerge" rather than being hand designed. In this paper we have described an architecture for evolving such subcomponents as a collection of cooperating species. We showed that evolutionary pressure to increase the overall fitness of the ecosystem can provide the needed stimulus for the emergence of an appropriate number of interdependent subcomponents that cover multiple niches, evolve to an appropriate level of generality, and adapt as the number and roles of their fellow subcomponents change over time. Finally, we explored the emergence of coadapted subcomponents in more detail through a case study involving neural network evolution, which suggested that, as our domains become increasingly complex, the evolution of species may need to be driven by more than the overall fitness of the ecosystem to produce good decompositions.

The results are quite positive in a number of ways. First, the architecture has demonstrated the ability to evolve useful decompositions expressed computationally as the emergence of cooperating species. Second, the cooperative coevolutionary architecture has been shown to be a general extension for any of the standard EC paradigms and not tightly coupled to a specific approach such as GAs. Finally, this architecture is allowing us to scale up to much more complex problems than possible with our standard EAs. As an example, we are currently using this approach to evolve complex behaviors for robots in multi-agent environments. In this regard, the architecture is suitable both for coevolving various modalities of behavior within a single robot, and for coevolving the behavior of a group of cooperating robots.

At the same time there are a number of improvements to the current system that we are exploring. At present, the mechanism for evaluating the collaborative potential of coevolving components has been deliberately kept as simple as possible. There are several ways in which such evaluations could be made to be more effective. Similarly, the mechanism for the creation and destruction of species is currently quite simplistic and provides opportunities for significant improvements. Finally, our current coevolutionary architecture is implemented serially even though parallel implementations are quite natural. Of particular interest as we scale up to more complex domains are implementations of the architecture in which species are assigned to separate processors and coevolved asynchronously.

Acknowledgments

This work was supported by the Office of Naval Research.

References

Brown, Jr., W. L. and Wilson, E. O. (1956). Character displacement. *Systematic Zoology*, 5(2):49–64.

de Garis, H. (1990). Building artificial nervous systems using genetically programmed neural network modules. In Porter, B. and Mooney, R., editors, *Proceedings of the Seventh International Conference on Machine Learning*, pages 132–139, Morgan Kaufmann, Palo Alto, California.

Fahlman, S. E. and Lebiere, C. (1990). The cascade-correlation learning architecture. Technical Report CMU-CS-90-100, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania.

Forrest, S. and Perelson, A. S. (1990). Genetic algorithms and the immune system. In Schwefel, H.-P. and Männer, R., editors, *Parallel Problem Solving from Nature*, pages 320–325, Springer-Verlag Berlin, Germany.

Forrest, S., Javornik, B., Smith, R. E. and Perelson, A. S. (1993). Using genetic algorithms to explore pattern recognition in the immune system. *Evolutionary Computation*, 1(3):191–211.

Giordana, A. and Neri, F. (1996). Search-intensive concept induction. *Evolutionary Computation*, 3(4):375–416.

Giordana, A., Saitta, L. and Zini, F. (1994). Learning disjunctive concepts by means of genetic algorithms. In Cohen, W. and Hirsh, H., editors, *Proceedings of the Eleventh International Conference on Machine Learning*, pages 96–104, Morgan Kaufmann, San Francisco, California.

Hillis, D. W. (1991). Co-evolving parasites improve simulated evolution as an optimization procedure. In Langton, C. G., Taylor, C., Farmer, J. D. and Rasmussen, S., editors, *Artificial Life II, SFI Studies in the Sciences of Complexity*, Volume 10, pages 313–324. Addison-Wesley, Redwood City, California.

Holland, J. H. (1986). Escaping brittleness: The possibilities of general purpose learning algorithms applied to parallel rule-based systems. In Michalski, R. S., Carbonell, J. G. and Mitchell, T. M., editors, *Machine Learning*, Volume 2, pages 593–623. Morgan Kaufman, Los Altos, California.

Holland, J. H. and Reitman, J. S. (1978). Cognitive systems based on adaptive algorithms. In Waterman, D. A. and Hayes-Roth, F., editors, *Pattern-Directed Inference Systems*. Academic Press, New York, New York.

Husbands, P. and Mill, F. (1991). Simulated co-evolution as the mechanism for emergent planning and scheduling. In Belew, R. K. and Booker, L. B., editors, *Proceedings of the Fourth International Conference on Genetic Algorithms*, pages 264–270, Morgan Kaufmann, San Mateo, California.

Karunanithi, N., Das, R. and Whitley, D. (1992). Genetic cascade learning for neural networks. In Whitley, L. D. and Schaffer, J. D., editors, *COGANN-92 International Workshop on Combinations of Genetic Algorithms and Neural Networks*, pages 134–145, IEEE Computer Society Press, Piscataway, New Jersey.

Kauffman, S. A. and Johnsen, S. (1991). Co-evolution to the edge of chaos: Coupled fitness landscapes, poised states, and co-evolutionary avalanches. In Langton, C. G., Taylor, C., Farmer, J. D. and Rasmussen, S., editors, *Artificial Life II, SFI Studies in the Sciences of Complexity*, Volume 10, pages 325–369. Addison-Wesley, Redwood City, California.

Koza, J. R. (1993). Hierarchical automatic function definition in genetic programming. In Whitley, L. D., editor, *Foundations of Genetic Algorithms 2*, pages 297–318. Morgan Kaufmann, San Mateo, California.

Lack, D. L. (1947). *Darwin's Finches*. Cambridge University Press, Cambridge, England.

Lang, K. J. and Witbrock, M. J. (1988). Learning to tell two spirals apart. In Touretzky, D., Hinton, G. and Sejnowski, T., editors, *Proceedings of the 1988 Connectionist Models Summer School*, pages 52–59, Morgan Kaufmann, San Mateo, California.

Lin, L.-J. (1993). Hierarchical learning of robot skills by reinforcement. In *Proceedings of the 1993 International Joint Conference on Neural Networks*, pages 181–186, IEEE Computer Society Press, Piscataway, New Jersey.

Moriarty, D. E. (1997). *Symbiotic Evolution of Neural Networks in Sequential Decision Tasks*. Ph. D. thesis, Department of Computer Science, University of Texas, Austin, Texas.

Moriarty, D. E. and Miikkulainen, R. (1997). Forming neural networks through efficient and adaptive coevolution. *Evolutionary Computation*, 5(4):373–399.

Paredis, J. (1995). The symbiotic evolution of solutions and their representations. In Eshelman, L., editor, *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 359–365, Morgan Kaufmann, San Francisco, California.

Potter, M. A. (1992). A genetic cascade-correlation learning algorithm. In Whitley, L. D. and Schaffer, J. D., editors, *COGANN-92 International Workshop on Combinations of Genetic Algorithms and Neural Networks*, pages 123–133, IEEE Computer Society Press, Piscataway, New Jersey.

Potter, M. A. and De Jong, K. A. (1994). A cooperative coevolutionary approach to function optimization. In Davidor, Y. and Schwefel, H.-P., editors, *Proceedings of the Third Conference on Parallel Problem Solving from Nature*, pages 249–257, Springer-Verlag, Berlin, Germany.

Potter, M. A. and De Jong, K. A. (1995). Evolving neural networks with collaborative species. In Ören, T. I. and Birta, L. G., editors, *Proceedings of the 1995 Summer Computer Simulation Conference*, pages 340–345, The Society for Computer Simulation, San Diego, California.

Potter, M. A., De Jong, K. A. and Grefenstette, J. J. (1995). A coevolutionary approach to learning sequential decision rules. In Eshelman, L., editor, *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 366–372, Morgan Kaufmann, San Francisco, California.

Rosca, J. P. and Ballard, D. H. (1994). Hierarchical self-organization in genetic programming. In Cohen, W. and Hirsh, H., editors, *Proceedings of the Eleventh International Conference on Machine Learning*, pages 251–258, Morgan Kaufmann, San Francisco, California.

Rosca, J. P. and Ballard, D. H. (1996). Discovery of subroutines in genetic programming. In Angeline, P. and Kinnear, K. E., editors, *Advances in Genetic Programming 2*, Chapter 9. The MIT Press, Cambridge, Massachusetts.

Rosin, C. D. and Belew, R. K. (1995). Methods for competitive co-evolution: Finding opponents worth beating. In Eshelman, L., editor, *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 373–380, Morgan Kaufmann, San Francisco, California.

Rumelhart, D. E., Hinton, G. E. and Williams, R. J. (1986). Learning internal representations by error propagation. In Rumelhart, D. E. and McClelland, J. L., editors, *Parallel Distributed Processing: Explorations in the Microstructures of Cognition*, Volume 1, pages 318–362. The MIT Press, Cambridge, Massachusetts.

Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, 3(3):210–229.

Schwefel, H.-P. (1995). *Evolution and Optimum Seeking*. John Wiley and Sons, New York, New York.

Singh, S. P. (1992). Transfer of learning by composing solutions of elemental sequential tasks. *Machine Learning*, 8:323–339.

Skinner, B. F. (1938). *The Behavior of Organisms: An Experimental Analysis.* Appleton-Century, New York, New York.

Smith, J. M. (1989). *Evolutionary Genetics*. Oxford University Press, New York, New York.

Smith, R. E., Forrest, S. and Perelson, A. S. (1993). Searching for diverse, cooperative populations with genetic algorithms. *Evolutionary Computation*, 1(2):127–149.

Suewatanakul, W. and Himmelblau, D. M. (1992). Comparison of artificial neural networks and traditional classifiers via the two-spiral problem. In Padgett, M. L., editor, *Proceedings of the Third Workshop on Neural Networks: Academic/Industrial/NASA/Defense*, pages 275–282, Society for Computer Simulation, San Diego, California.

Whitley, D. and Karunanithi, N. (1991). Generalization in feed forward neural networks. In *Proceedings of the International Joint Conference on Neural Networks – Seattle*, Volume 2, pages 77–82, IEEE, Piscataway, New Jersey.

Wright, S. (1932). The roles of mutation, inbreeding, crossbreeding and selection in evolution. In Jones, D. F., editor, *Proceedings of the Sixth International Conference of Genetics*, pages 356–366, Brooklyn Botanic Garden, Brooklyn, New York.

This article has been cited by:

1. Sukanta Dey, Sukumar Nandi, Gaurav Trivedi. PGRDP: Reliability, Delay, and Power-Aware Area Minimization of Large-Scale VLSI Power Grid Network Using Cooperative Coevolution 69-84. [Crossref]
2. Javier Del Ser, Eneko Osaba, Daniel Molina, Xin-She Yang, Sancho Salcedo-Sanz, David Camacho, Swagatam Das, Ponnuthurai N. Suganthan, Carlos A. Coello Coello, Francisco Herrera. 2019. Bio-inspired computation: Where we stand and what's next. *Swarm and Evolutionary Computation* **48**, 220-250. [Crossref]
3. Jorge Gálvez, Erik Cuevas, Salvador Hinojosa, Omar Avalos, Marco Pérez-Cisneros. 2019. A reactive model based on neighborhood consensus for continuous optimization. *Expert Systems with Applications* **121**, 115-141. [Crossref]
4. Yuxin Liu, Yi Mei, Mengjie Zhang, Zili Zhang. A Predictive-Reactive Approach with Genetic Programming and Cooperative Co-evolution for Uncertain Capacitated Arc Routing Problem. *Evolutionary Computation* **0**:ja, 1-25. [Abstract] [PDF]
5. Tomasz Praczyk. 2019. Neural collision avoidance system for biomimetic autonomous underwater vehicle. *Soft Computing* **7**. . [Crossref]
6. Emily L. Dolson, Anya E. Vostinar, Michael J. Wiser, Charles Ofria. 2019. The MODES Toolbox: Measurements of Open-Ended Dynamics in Evolving Systems. *Artificial Life* **25**:1, 50-73. [Abstract] [Full Text] [PDF] [PDF Plus]
7. Wu Lin, Qiuzhen Lin, Zexuan Zhu, Jianqiang Li, Jianyong Chen, Zhong Ming. 2019. Evolutionary Search with Multiple Utopian Reference Points in Decomposition-Based Multiobjective Optimization. *Complexity* **2019**, 1-22. [Crossref]
8. Borhan Kazimipour, Mohammad Nabi Omidvar, A.K. Qin, Xiaodong Li, Xin Yao. 2019. Bandit-based cooperative coevolution for tackling contribution imbalance in large-scale optimization problems. *Applied Soft Computing* **76**, 265-281. [Crossref]
9. Jorge Gomes, Pedro Mariano, Anders Lyhne Christensen. 2019. Challenges in cooperative coevolution of physically heterogeneous robot teams. *Natural Computing* **18**:1, 29-46. [Crossref]
10. Zhigang Ren, Bei Pang, Muyi Wang, Zuren Feng, Yongsheng Liang, An Chen, Yipeng Zhang. 2019. Surrogate model assisted cooperative coevolution for large scale optimization. *Applied Intelligence* **49**:2, 513-531. [Crossref]
11. Julien Blanchard, Charlotte Beauthier, Timoteo Carletti. A Surrogate-Assisted Cooperative Co-evolutionary Algorithm for Solving High Dimensional, Expensive and Black Box Optimization Problems 41-52. [Crossref]
12. Nikola K. Kasabov. Evolutionary- and Quantum-Inspired Computation. Applications for SNN Optimisation 245-287. [Crossref]
13. Wei Sun, Yuting Wu, Qi Lou, Yang Yu. 2019. A Cooperative Coevolution Algorithm for the Seru Production with Minimizing Makespan. *IEEE Access* 1-1. [Crossref]
14. Hongwei Ge, Liang Sun, Kai Zhang, Chunguo Wu. 2019. Cooperative differential evolution framework with utility-based adaptive grouping for large-scale optimization. *Advances in Mechanical Engineering* **11**:3, 168781401983416. [Crossref]
15. Moshe Sipper, Jason H. Moore, Ryan J. Urbanowicz. Solution and Fitness Evolution (SAFE): Coevolving Solutions and Their Objective Functions 146-161. [Crossref]
16. James McDermott. Why Is Auto-Encoding Difficult for Genetic Programming? 131-145. [Crossref]
17. Rohitash Chandra, Sally Cripps. 2018. Coevolutionary multi-task learning for feature-based modular pattern classification. *Neurocomputing* **319**, 164-175. [Crossref]
18. Maria Amélia Lopes Silva, Sérgio Ricardo de Souza, Marcone Jamilson Freitas Souza, Moacir Felizardo de França Filho. 2018. Hybrid metaheuristics and multi-agent systems for solving optimization problems: A review of frameworks and a comparative analysis. *Applied Soft Computing* **71**, 433-459. [Crossref]
19. Haiyan Liu, Yuping Wang, Liwen Liu, Xiaodong Li. 2018. A two phase hybrid algorithm with a new decomposition method for large scale optimization. *Integrated Computer-Aided Engineering* **25**:4, 349-367. [Crossref]
20. Rohitash Chandra, Yew-Soon Ong, Chi-Keong Goh. 2018. Co-evolutionary multi-task learning for dynamic time series prediction. *Applied Soft Computing* **70**, 576-589. [Crossref]
21. Michaela Drahosova, Lukas Sekanina, Michal Wiglasz. Adaptive Fitness Predictors in Coevolutionary Cartesian Genetic Programming. *Evolutionary Computation*, ahead of print1-27. [Abstract] [PDF] [PDF Plus]
22. Meryem Saidi, Mohammed El Amine Bechar, Nesma Settouti, Mohamed Amine Chikh. 2018. Instances selection algorithm by ensemble margin. *Journal of Experimental & Theoretical Artificial Intelligence* **30**:3, 457-478. [Crossref]
23. Marco Castellani. 2018. Competitive co-evolution of multi-layer perceptron classifiers. *Soft Computing* **22**:10, 3417-3432. [Crossref]
24. Vinícius Veloso de Melo, Wolfgang Banzhaf. 2018. Automatic feature engineering for regression models with machine learning: An evolutionary computation and statistics hybrid. *Information Sciences* **430-431**, 287-313. [Crossref]

25. Zijian Cao, Lei Wang, Xinhong Hei. 2018. A global-best guided phase based optimization algorithm for scalable optimization problems and its application. *Journal of Computational Science* **25**, 38-49. [Crossref]
26. John Park, Yi Mei, Su Nguyen, Gang Chen, Mengjie Zhang. 2018. An investigation of ensemble combination schemes for genetic programming based hyper-heuristic approaches to dynamic job shop scheduling. *Applied Soft Computing* **63**, 72-86. [Crossref]
27. Konstantinos E. Parsopoulos. Particle Swarm Methods 639-685. [Crossref]
28. Daniel Yska, Yi Mei, Mengjie Zhang. Genetic Programming Hyper-Heuristic with Cooperative Coevolution for Dynamic Flexible Job Shop Scheduling 306-321. [Crossref]
29. Evgenii Sopov. Adaptive Variable-Size Random Grouping for Evolutionary Large-Scale Global Optimization 583-592. [Crossref]
30. Shuai Wu, Zhitao Zou, Wei Fang. A Dynamic Global Differential Grouping for Large-Scale Black-Box Optimization 593-603. [Crossref]
31. Mohammad Zohaib. 2018. Dynamic Difficulty Adjustment (DDA) in Computer Games: A Review. *Advances in Human-Computer Interaction* **2018**, 1. [Crossref]
32. Damien O'Neill, Bing Xue, Mengjie Zhang. Co-evolution of Novel Tree-Like ANNs and Activation Functions: An Observational Study 616-629. [Crossref]
33. Weiping Ding, Chin-Teng Lin, Zehong Cao. 2018. Deep Neuro-Cognitive Co-Evolution for Fuzzy Attribute Reduction by Quantum Leaping PSO With Nearest-Neighbor Memeplexes. *IEEE Transactions on Cybernetics* 1-14. [Crossref]
34. David Corne, Michael A. Lones. Evolutionary Algorithms 1-22. [Crossref]
35. Emile Glorieux, Bo Svensson, Fredrik Danielsson, Bengt Lennartson. 2017. Constructive cooperative coevolution for large-scale global optimisation. *Journal of Heuristics* **23**:6, 449-469. [Crossref]
36. David Greiner, Jacques Periaux, Jose M. Emperador, Blas Galván, Gabriel Winter. 2017. Game Theory Based Evolutionary Algorithms: A Review with Nash Applications in Structural Engineering Optimization Problems. *Archives of Computational Methods in Engineering* **24**:4, 703-750. [Crossref]
37. Helan Liang, Yanhua Du. 2017. Dynamic service selection with QoS constraints and inter-service correlations using cooperative coevolution. *Future Generation Computer Systems* **76**, 119-135. [Crossref]
38. Jun-ichi Matsuoka, Yuki Nakashima, Satoshi Ono. A preliminary study on designing a benchmark problem for analysis of sparsely-synchronized heterogeneous coevolution 1-8. [Crossref]
39. Marco Scirea, Peter Eklund, Julian Togelius, Sebastian Risi. Primal-improv: Towards co-evolutionary musical improvisation 172-177. [Crossref]
40. Rafał Dreżewski, Krzysztof Doroz. 2017. An Agent-Based Co-Evolutionary Multi-Objective Algorithm for Portfolio Optimization. *Symmetry* **9**:9, 168. [Crossref]
41. Khalid M. Salama, Ashraf M. Abdelbar, Ayah M. Helal, Alex A. Freitas. 2017. Instance-based classification with Ant Colony Optimization. *Intelligent Data Analysis* **21**:4, 913-944. [Crossref]
42. Bin Cao, Jianwei Zhao, Zhihan Lv, Xin Liu. 2017. A Distributed Parallel Cooperative Coevolutionary Multiobjective Evolutionary Algorithm for Large-Scale Optimization. *IEEE Transactions on Industrial Informatics* **13**:4, 2030-2038. [Crossref]
43. Jorge Gomes, Pedro Mariano, Anders Lyhne Christensen. 2017. Novelty-Driven Cooperative Coevolution. *Evolutionary Computation* **25**:2, 275-307. [Abstract] [Full Text] [PDF] [PDF Plus]
44. Rohitash Chandra, Yew-Soon Ong, Chi-Keong Goh. 2017. Co-evolutionary multi-task learning with predictive recurrence for multi-step chaotic time series prediction. *Neurocomputing* **243**, 21-34. [Crossref]
45. Satish Kumar Jaiswal, Hitoshi Iba. Coevolution of mapping functions for linear SVM 2225-2232. [Crossref]
46. Alejandro Rosales-Perez, Andres E. Gutierrez-Rodriguez, Jose C. Ortiz-Bayliss, Hugo Terashima-Marin, Carlos A. Coello Coello. Evolutionary multilabel hyper-heuristic design 2622-2629. [Crossref]
47. Wei Fang, Lingzhi Zhang, Jianhong Zhou, Xiaojun Wu, Jun Sun. A novel quantum-behaved particle swarm optimization with random selection for large scale optimization 2746-2751. [Crossref]
48. Michael Cook, Simon Colton, Jeremy Gow. 2017. The ANGELINA Videogame Design System?Part I. *IEEE Transactions on Computational Intelligence and AI in Games* **9**:2, 192-203. [Crossref]
49. Bing Wang, Kathryn E. Merrick, Hussein A. Abbass. 2017. Co-Operative Coevolutionary Neural Networks for Mining Functional Association Rules. *IEEE Transactions on Neural Networks and Learning Systems* **28**:6, 1331-1344. [Crossref]
50. Edward T. Norris, Xin Liu. 2017. An evolutionary algorithm for spatial discretization optimization. *Progress in Nuclear Energy* **97**, 220-230. [Crossref]

51. Shane Strasser, John Sheppard, Nathan Fortier, Rollie Goodman. 2017. Factored Evolutionary Algorithms. *IEEE Transactions on Evolutionary Computation* **21**:2, 281-293. [Crossref]

52. Nasser R. Sabar, Jemal Abawajy, John Yearwood. 2017. Heterogeneous Cooperative Co-Evolution Memetic Differential Evolution Algorithm for Big Data Optimization Problems. *IEEE Transactions on Evolutionary Computation* **21**:2, 315-327. [Crossref]

53. Su Nguyen, Yi Mei, Mengjie Zhang. 2017. Genetic programming for production scheduling: a survey with a unified framework. *Complex & Intelligent Systems* **3**:1, 41-66. [Crossref]

54. Varun Kumar Ojha, Ajith Abraham, Václav Snášel. 2017. Ensemble of heterogeneous flexible neural trees using multiobjective genetic programming. *Applied Soft Computing* **52**, 909-924. [Crossref]

55. Xiao-Min Hu, Fei-Long He, Wei-Neng Chen, Jun Zhang. 2017. Cooperation coevolution with fast interdependency identification for large scale optimization. *Information Sciences* **381**, 142-160. [Crossref]

56. Min Shi, Shang Gao. 2017. Reference sharing: a new collaboration model for cooperative coevolution. *Journal of Heuristics* **23**:1, 1-30. [Crossref]

57. Kathrin Klamroth, Sanaz Mostaghim, Boris Naujoks, Silvia Poles, Robin Purshouse, Günter Rudolph, Stefan Ruzika, Serpil Sayın, Margaret M. Wiecek, Xin Yao. 2017. Multiobjective optimization for interwoven systems. *Journal of Multi-Criteria Decision Analysis* **24**:1-2, 71-81. [Crossref]

58. Fabio Caraffini, Ferrante Neri, Giovanni Iacca. Large Scale Problems in Practice: The Effect of Dimensionality on the Interaction Among Variables 636-652. [Crossref]

59. Chen Yating. Cooperation Coevolution Differential Evolution with Gradient Descent Strategy for Large Scale 429-439. [Crossref]

60. Rohitash Chandra. Co-evolutionary Multi-task Learning for Modular Pattern Classification 692-701. [Crossref]

61. Rohitash Chandra. Dynamic Cyclone Wind-Intensity Prediction Using Co-Evolutionary Multi-task Learning 618-627. [Crossref]

62. Peng Yang, Ke Tang, Xin Yao. 2017. Turning High-dimensional Optimization into Computationally Expensive Optimization. *IEEE Transactions on Evolutionary Computation* 1-1. [Crossref]

63. Cheng-Hung Chen, Chong-Bin Liu. 2017. Reinforcement Learning-Based Differential Evolution With Cooperative Coevolution for a Compensatory Neuro-Fuzzy Controller. *IEEE Transactions on Neural Networks and Learning Systems* 1-11. [Crossref]

64. Piotr Szymak. 2017. Comparison of Fuzzy System with Neural Aggregation FSNA with Classical TSK Fuzzy System in Anti-Collision Problem of USV. *Polish Maritime Research* **24**:3. . [Crossref]

65. Jing Tian, Xinchang Hao, Tomohiro Murata. 2017. Robust Optimization Method based on Hybridization of GA and MMEDA for Resource Constraint Project Scheduling with Uncertainty. *IEEJ Transactions on Electronics, Information and Systems* **137**:7, 957-966. [Crossref]

66. Emma Hart, Kevin Sim. 2016. A Hyper-Heuristic Ensemble Method for Static Job-Shop Scheduling. *Evolutionary Computation* **24**:4, 609-635. [Abstract] [Full Text] [PDF] [PDF Plus]

67. Rohitash Chandra, Shelvin Chand. 2016. Evaluation of co-evolutionary neural network architectures for time series prediction with mobile application in finance. *Applied Soft Computing* **49**, 462-473. [Crossref]

68. Giuseppe A. Trunfio, Paweł Topa, Jarosław Wąs. 2016. A new algorithm for adapting the configuration of subcomponents in large-scale optimization with cooperative coevolution. *Information Sciences* **372**, 773-795. [Crossref]

69. Haiyan Liu, Yuping Wang, Liwen Liu, Xiaodong Li, Xiaozhi Gao. A Two Phase Approach Based on Dynamic Variable Grouping and Self-Adaptive Group Search for Large Scale Optimization 170-174. [Crossref]

70. Xingguang Peng, Zhe Shi. Finding informative collaborators for cooperative co-evolutionary algorithms using a dynamic multi-population framework 1-6. [Crossref]

71. . Bibliography 127-148. [Crossref]

72. Thomas Weise, Yuezhong Wu, Raymond Chiong, Ke Tang, Jörg Lässig. 2016. Global versus local search: the impact of population sizes on evolutionary algorithm performance. *Journal of Global Optimization* **66**:3, 511-534. [Crossref]

73. Xingguang Peng, Kun Liu, Yaochu Jin. 2016. A dynamic optimization approach to the design of cooperative co-evolutionary algorithms. *Knowledge-Based Systems* **109**, 174-186. [Crossref]

74. Ivanoe De Falco, Eryk Laskowski, Richard Olejnik, Umberto Scafuri, Ernesto Tarantino, Marek Tudruj. 2016. Parallel extremal optimization in processor load balancing for distributed applications. *Applied Soft Computing* **46**, 187-203. [Crossref]

75. Noor Shaker. Intrinsically motivated reinforcement learning: A promising framework for procedural content generation 1-8. [Crossref]

76. Zahara Ali, Mohammad Waseem, Rahul Kumar, Priyanka Pandey, Ghulam Mohammad, M.A. Qadar Pasha. 2016. Unveiling the interactions among BMPR-2, ALK-1 and 5-HTT genes in the pathophysiology of HAPE. *Gene* **588**:2, 163-172. [Crossref]

77. Hongwei Ge, Liang Sun, Xin Yang. Adaptive hybrid differential evolution with circular sliding window for large scale optimization 87-94. [Crossref]

78. Luis Miguel Antonio, Carlos A. Coello Coello. Indicator-based cooperative coevolution for multi-objective optimization 991-998. [Crossref]

79. Kavitesh Bali, Rohitash Chandra, Mohammad N. Omidvar. Contribution based multi-island competitive cooperative coevolution 1823-1830. [Crossref]

80. Shamina Hussein, Rohitash Chandra, Anuraganand Sharma. Multi-step-ahead chaotic time series prediction using coevolutionary recurrent neural networks 3084-3091. [Crossref]

81. Guangming Dai, Xiaoyu Chen, Liang Chen, Maocai Wang, Lei Peng. Cooperative coevolution with dependency identification grouping for large scale global optimization 5201-5208. [Crossref]

82. Jing Tian, Tomohiro Murata. Robust Scheduling for Resource Constraint Scheduling Problem by Two-Stage GA and MMEDA 1042-1047. [Crossref]

83. Haiyan Liu, Yuping Wang, Xuyan Liu, Shiwei Guan. Empirical study of effect of grouping strategies for large scale optimization 3433-3439. [Crossref]

84. Wei Zhong, Ruiyi Su, Liangjin Gui, Zijie Fan. 2016. Topology and sizing optimization of discrete structures using a cooperative coevolutionary genetic algorithm with independent ground structures. *Engineering Optimization* **48**:6, 911-932. [Crossref]

85. Jian Xiong, Roel Leus, Zhenyu Yang, Hussein A. Abbass. 2016. Evolutionary multi-objective resource allocation and scheduling in the Chinese navigation satellite system project. *European Journal of Operational Research* **251**:2, 662-675. [Crossref]

86. Xueyuan Li, Xuejun Zhang, Huaxian Liu, Xiangmin Guan. Formation reconfiguration based on distributed cooperative coevolutionary for multi-UAV 2308-2311. [Crossref]

87. Weiping Ding, Jiehua Wang, Jiandong Wang. 2016. A hierarchical-coevolutionary-MapReduce-based knowledge reduction algorithm with robust ensemble Pareto equilibrium. *Information Sciences* **342**, 153-175. [Crossref]

88. Tomasz Praczyk. 2016. Cooperative co–evolutionary neural networks. *Journal of Intelligent & Fuzzy Systems* **30**:5, 2843-2858. [Crossref]

89. Ronghua Shang, Yang Li, Licheng Jiao. 2016. Co-evolution-based immune clonal algorithm for clustering. *Soft Computing* **20**:4, 1503-1519. [Crossref]

90. Ronghua Shang, Kaiyun Dai, Licheng Jiao, Rustam Stolkin. 2016. Improved Memetic Algorithm Based on Route Distance Grouping for Multiobjective Large Scale Capacitated Arc Routing Problems. *IEEE Transactions on Cybernetics* **46**:4, 1000-1013. [Crossref]

91. Ning Ding, Rasool Erfani, Hamid Mokhtar, Tohid Erfani. 2016. Agent Based Modelling for Water Resource Allocation in the Transboundary Nile River. *Water* **8**:4, 139. [Crossref]

92. Lianbo Ma, Yunlong Zhu, Dingyi Zhang, Ben Niu. 2016. A hybrid approach to artificial bee colony algorithm. *Neural Computing and Applications* **27**:2, 387-409. [Crossref]

93. Giuseppe A. Trunfio. Metaheuristics for Continuous Optimization of High-Dimensional Problems: State of the Art and Perspectives 437-460. [Crossref]

94. Evgenii Sopov. Large-Scale Global Optimization Using a Binary Genetic Algorithm with EDA-Based Decomposition 619-626. [Crossref]

95. Ke-Lin Du, M. N. S. Swamy. Genetic Algorithms 37-69. [Crossref]

96. Ke-Lin Du, M. N. S. Swamy. Topics in Evolutinary Algorithms 121-152. [Crossref]

97. Ilhem Boussaïd. Some Other Metaheuristics 229-262. [Crossref]

98. Luis Miguel Antonio, Carlos A. Coello Coello. Decomposition-Based Approach for Solving Large Scale Multi-objective Problems 525-534. [Crossref]

99. Jorge Gomes, Miguel Duarte, Pedro Mariano, Anders Lyhne Christensen. Cooperative Coevolution of Control for a Real Multirobot System 591-601. [Crossref]

100. Gary Wong, Rohitash Chandra, Anuraganand Sharma. Memetic Cooperative Neuro-Evolution for Chaotic Time Series Prediction 299-308. [Crossref]

101. Jin Tian, Minqiang Li, Fuzan Chen, Nan Feng. 2016. Learning Subspace-Based RBFNN Using Coevolutionary Algorithm for Complex Classification Tasks. *IEEE Transactions on Neural Networks and Learning Systems* **27**:1, 47-61. [Crossref]

102. Yang Liu, Junfei Liu, Liwei Tian, Lianbo Ma. 2016. Hybrid Artificial Root Foraging Optimizer Based Multilevel Threshold for Image Segmentation. *Computational Intelligence and Neuroscience* **2016**, 1-16. [Crossref]

103. Tohid Erfani, Rasool Erfani. 2015. An evolutionary approach to solve a system of multiple interrelated agent problems. *Applied Soft Computing* **37**, 40-47. [Crossref]

104. Haiyan Liu, Shiwei Guan, Fangjie Liu, Yuping Wang. Cooperative Co-evolution with Formula Based Grouping and CMA for Large Scale Optimization 282-285. [Crossref]

105. Hongwei Ge, Liang Sun, Xin Yang, Shinichi Yoshida, Yanchun Liang. 2015. Cooperative differential evolution with fast variable interdependence learning and cross-cluster mutation. *Applied Soft Computing* **36**, 300-314. [Crossref]

106. Juan José Palacios, Inés González-Rodríguez, Camino R. Vela, Jorge Puente. 2015. Coevolutionary makespan optimisation through different ranking methods for the fuzzy flexible job shop. *Fuzzy Sets and Systems* **278**, 81-97. [Crossref]

107. Zhouzhou Su, Wei Yan. 2015. A fast genetic algorithm for solving architectural design optimization problems. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **29**:4, 457-469. [Crossref]

108. Tomasz Praczyk. 2015. Using evolutionary neural networks to predict spatial orientation of a ship. *Neurocomputing* **166**, 229-243. [Crossref]

109. Zhao-Hua Liu, Xiao-Hua Li, Liang-Hong Wu, Shao-Wu Zhou, Kan Liu. 2015. GPU-Accelerated Parallel Coevolutionary Algorithm for Parameters Identification and Temperature Monitoring in Permanent Magnet Synchronous Machines. *IEEE Transactions on Industrial Informatics* **11**:5, 1220-1230. [Crossref]

110. Yue-Jiao Gong, Wei-Neng Chen, Zhi-Hui Zhan, Jun Zhang, Yun Li, Qingfu Zhang, Jing-Jing Li. 2015. Distributed evolutionary algorithms and their models: A survey of the state-of-the-art. *Applied Soft Computing* **34**, 286-300. [Crossref]

111. Mostafa Z. Ali, Noor H. Awad, Ponnuthurai N. Suganthan. 2015. Multi-population differential evolution with balanced ensemble of mutation strategies for large-scale global optimization. *Applied Soft Computing* **33**, 304-327. [Crossref]

112. Yu-Jun Zheng, Hai-Feng Ling, Sheng-Yong Chen, Jin-Yun Xue. 2015. A Hybrid Neuro-Fuzzy Network Based on Differential Biogeography-Based Optimization for Online Population Classification in Earthquakes. *IEEE Transactions on Fuzzy Systems* **23**:4, 1070-1083. [Crossref]

113. JorgeGomes, PedroMariano, Anders LyhneChristensen. 2015. Cooperative Coevolution of Morphologically Heterogeneous Robots. *The 2018 Conference on Artificial Life: A Hybrid of the European Conference on Artificial Life (ECAL) and the International Conference on the Synthesis and Simulation of Living Systems (ALIFE)* 312-319. [Citation] [PDF] [PDF Plus]

114. Xiang-wei Zheng, Dian-jie Lu, Xiao-guang Wang, Hong Liu. 2015. A cooperative coevolutionary biogeography-based optimizer. *Applied Intelligence* **43**:1, 95-111. [Crossref]

115. Pablo Mesejo, Oscar Ibáñez, Enrique Fernández-Blanco, Francisco Cedrón, Alejandro Pazos, Ana B. Porto-Pazos. 2015. Artificial Neuron–Glia Networks Learning Approach Based on Cooperative Coevolution. *International Journal of Neural Systems* **25**:04, 1550012. [Crossref]

116. Yu-Jun Zheng, Xiao-Bei Wu. Tuning maturity model of ecogeography-based optimization on CEC 2015 single-objective optimization test problems 1018-1024. [Crossref]

117. Rohitash Chandra, Kavina Dayal. Cooperative neuro-evolution of Elman recurrent networks for tropical cyclone wind-intensity prediction in the South Pacific region 1784-1791. [Crossref]

118. Zijian Cao, Lei Wang, Yuhui Shi, Xinhong Hei, Xiaofeng Rong, Qiaoyong Jiang, Hongye Li. An effective cooperative coevolution framework integrating global and local search for large scale optimization problems 1986-1993. [Crossref]

119. James Watson, Geoff Nitschke. Deriving minimal sensory configurations for evolved cooperative robot teams 3065-3071. [Crossref]

120. Francois-Michel De Rainville, Jean-Philippe Mercier, Christian Gagne, Philippe Giguere, Denis Laurendeau. Multisensor placement in 3D environments via visibility estimation and derivative-free optimization 3327-3334. [Crossref]

121. Aleksander Byrski, Rafał Dreżewski, Leszek Siwik, Marek Kisiel-Dorohinicki. 2015. Evolutionary multi-agent systems. *The Knowledge Engineering Review* **30**:2, 171-186. [Crossref]

122. Sedigheh Mahdavi, Mohammad Ebrahim Shiri, Shahryar Rahnamayan. 2015. Metaheuristics in large-scale global continues optimization: A survey. *Information Sciences* **295**, 407-428. [Crossref]

123. Tomasz Praczyk. 2015. Neural anti-collision system for Autonomous Surface Vehicle. *Neurocomputing* **149**, 559-572. [Crossref]

124. Rahul Kumar, Samantha Kohli, Aastha Mishra, Ritu Garg, Perwez Alam, Tsering Stobdan, Azim Nejatizadeh, Mohit Gupta, Sanjay Tyagi, M. A. Qadar Pasha. 2015. Interactions Between the Genes of Vasodilatation Pathways Influence Blood Pressure and Nitric Oxide Level in Hypertension. *American Journal of Hypertension* **28**:2, 239-247. [Crossref]

125. Jonathan E. Rowe. Genetic Algorithms 825-844. [Crossref]

126. Malcolm I. Heywood, Krzysztof Krawiec. Solving Complex Problems with Coevolutionary Algorithms 547-573. [Crossref]

127. Jorge Gomes, Pedro Mariano, Anders Lyhne Christensen. Hyb-CCEA 1251-1252. [Crossref]
128. Tohid Erfani, Rasool Erfani. Fair Resource Allocation Using Multi-population Evolutionary Algorithm 214-224. [Crossref]
129. Konstantinos E. Parsopoulos. Particle Swarm Methods 1-47. [Crossref]
130. Tomasz Praczyk. 2014. Using augmenting modular neural networks to evolve neuro-controllers for a team of underwater vehicles. *Soft Computing* **18**:12, 2445-2460. [Crossref]
131. Gaoji Sun, Ruiqing Zhao. 2014. Dynamic partition search algorithm for global numerical optimization. *Applied Intelligence* **41**:4, 1108-1126. [Crossref]
132. Chun-Kit Au, Ho-Fung Leung. 2014. Cooperative coevolutionary algorithms for dynamic optimization: an experimental study. *Evolutionary Intelligence* **7**:4, 201-218. [Crossref]
133. Chong Zhi Song, Zhao Youqun, Wang Lu. 2014. Tri-objective co-evolutionary algorithm and application of suspension parameter design based on lizard behavior bionics. *Journal of Mechanical Science and Technology* **28**:12, 4857-4867. [Crossref]
134. M.D. Pérez-Godoy, Antonio J. Rivera, C.J. Carmona, M.J. del Jesus. 2014. Training algorithms for Radial Basis Function Networks to tackle learning processes with imbalanced data-sets. *Applied Soft Computing* **25**, 26-39. [Crossref]
135. Catalin Stoean, Ruxandra Stoean. 2014. Post-evolution of variable-length class prototypes to unlock decision making within support vector machines. *Applied Soft Computing* **25**, 159-173. [Crossref]
136. Zhao-Hua Liu, Jing-Xing Zhao, Jing-Xing Zhao, Wen Tan. Multi-core based parallelized cooperative PSO with immunity for large scale optimization problem 96-100. [Crossref]
137. Michaela Sikulova, Gergely Komjathy, Lukas Sekanina. Towards compositional coevolution in evolutionary circuit design 157-164. [Crossref]
138. Helon Vicente Hultmann Ayala, Luciano Ferreira da Cruz, Roberto Zanetti Freire, Leandro dos Santos Coelho. Cascaded evolutionary multiobjective identification based on correlation function statistical tests for improving velocity analyzes in swimming 185-193. [Crossref]
139. Lian-bo Ma, Kun-yuan Hu, Yun-long Zhu, Han-ning Chen. 2014. Improved multi-objective artificial bee colony algorithm for optimal power flow problem. *Journal of Central South University* **21**:11, 4220-4227. [Crossref]
140. Ruochen Liu, Yangyang Chen, Wenping Ma, Caihong Mu, Licheng Jiao. 2014. A novel cooperative coevolutionary dynamic multi-objective optimization algorithm using a new predictive model. *Soft Computing* **18**:10, 1913-1929. [Crossref]
141. Xin-Chang Hao, Jei-Zheng Wu, Chen-Fu Chien, Mitsuo Gen. 2014. The cooperative estimation of distribution algorithm: a novel approach for semiconductor final test scheduling problems. *Journal of Intelligent Manufacturing* **25**:5, 867-879. [Crossref]
142. Wenjing Zhao, Sameer Alam, Hussein A. Abbass. 2014. MOCCA-II: A multi-objective co-operative co-evolutionary algorithm. *Applied Soft Computing* **23**, 407-416. [Crossref]
143. Rob Mills, Thomas Jansen, Richard A. Watson. 2014. Transforming Evolutionary Search into Higher-Level Evolutionary Search by Capturing Problem Structure. *IEEE Transactions on Evolutionary Computation* **18**:5, 628-642. [Crossref]
144. Ronghua Shang, Yuying Wang, Jia Wang, Licheng Jiao, Shuo Wang, Liping Qi. 2014. A multi-population cooperative coevolutionary algorithm for multi-objective capacitated arc routing problem. *Information Sciences* **277**, 609-642. [Crossref]
145. Wenbing Huang, Fuchun Sun. Using hierarchical dirichlet processes to regulate weight parameters of Restricted Boltzmann Machines 1-8. [Crossref]
146. Catalin Stoean, Ruxandra Stoean, Adrian Sandita. Investigation of Alternative Evolutionary Prototype Generation in Medical Classification 537-543. [Crossref]
147. Rohitash Chandra. 2014. Memetic cooperative coevolution of Elman recurrent neural networks. *Soft Computing* **18**:8, 1549-1559. [Crossref]
148. Antonio Bolufe-Rohler, Stephen Chen. Extending Minimum Population Search towards large scale global optimization 845-852. [Crossref]
149. Eman Sayed, Daryl Essam, Ruhul Sarker, Saber Elsayed. A decomposition-based algorithm for dynamic economic dispatch problems 1898-1905. [Crossref]
150. Grant Dick, Xin Yao. Model representation and cooperative coevolution for finite-state machine evolution 2700-2707. [Crossref]
151. Shelvin Chand, Rohitash Chandra. Multi-objective cooperative coevolution of neural networks for time series prediction 190-197. [Crossref]
152. Shelvin Chand, Rohitash Chandra. Cooperative coevolution of feed forward neural networks for financial time series problem 202-209. [Crossref]

153. Lianbo Ma, Kunyuan Hu, Yunlong Zhu, Hanning Chen. 2014. Cooperative artificial bee colony algorithm for multi-objective RFID network planning. *Journal of Network and Computer Applications* **42**, 143-162. [Crossref]
154. . Introduction to Evolutionary Algorithms 27-47. [Crossref]
155. Haoyang Chen, Yasukuni Mori, Ikuo Matsuba. 2014. Solving the balance problem of massively multiplayer online role-playing games using coevolutionary programming. *Applied Soft Computing* **18**, 1-11. [Crossref]
156. Catherine D. Schuman, J. Douglas Birdwell, Mark Dean. Neuroscience-inspired inspired dynamic architectures 1-4. [Crossref]
157. Chourouk Guettas, Foudil Cherif, Thomas Breton, Yves Duthen. Cooperative co-evolution of configuration and control for modular robots 26-31. [Crossref]
158. Su Nguyen, Mengjie Zhang, Mark Johnston, Kay Chen Tan. 2014. Automatic Design of Scheduling Policies for Dynamic Multi-objective Job Shop Scheduling via Cooperative Coevolution Genetic Programming. *IEEE Transactions on Evolutionary Computation* **18**:2, 193-208. [Crossref]
159. Doina Bucur, Giovanni Iacca, Giovanni Squillero, Alberto Tonda. 2014. The impact of topology on energy consumption for collection tree protocols: An experimental assessment through evolutionary computation. *Applied Soft Computing* **16**, 210-222. [Crossref]
160. Fang Liu, Zhiguang Zhou. 2014. An improved QPSO algorithm and its application in the high-dimensional complex problems. *Chemometrics and Intelligent Laboratory Systems* **132**, 82-90. [Crossref]
161. Si-fu Li, Xiao-Lin Wang, Jian-Zhong Xiao, Zheng-Jie Yin. 2014. Self-adaptive obtaining water-supply reservoir operation rules: Co-evolution artificial immune system. *Expert Systems with Applications* **41**:4, 1262-1270. [Crossref]
162. Xuesong Guo, Zhengwei Zhu, Jia Shi. 2014. Integration of semi-fuzzy SVDD and CC-Rule method for supplier selection. *Expert Systems with Applications* **41**:4, 2083-2097. [Crossref]
163. E. Parras-Gutierrez, V.M. Rivas, M. Garcia-Arenas, M.J. del Jesus. 2014. Short, medium and long term forecasting of time series using the L-Co-R algorithm. *Neurocomputing* **128**, 433-446. [Crossref]
164. Adeleh Ebrahimi, Mohammad-R. Akbarzadeh-T. Dynamic difficulty adjustment in games by using an interactive self-organizing architecture 1-6. [Crossref]
165. Rafal Kicinger, Moein Ganji, Jit-Tat Chen, Raghu Reddy, Mohamed Ellejmi. Co-evolutionary Approach to Improve Robustness of Routing Algorithms against Disruptive Events on the Airport Surface . [Crossref]
166. RAHUL KALA. 2014. Coordination in Navigation of Multiple Mobile Robots. *Cybernetics and Systems* **45**:1, 1-24. [Crossref]
167. Nicholas Tomko, Inman Harvey, Nathaniel Virgo, Andrew Philippides. 2014. Many Hands Make Light Work: Further Studies in Group Evolution. *Artificial Life* **20**:1, 163-181. [Abstract] [Full Text] [PDF] [PDF Plus]
168. Sheng Su, Haijie Yu, Zhenghua Wu, Wenhong Tian. 2014. A distributed coevolutionary algorithm for multiobjective hybrid flowshop scheduling problems. *The International Journal of Advanced Manufacturing Technology* **70**:1-4, 477-494. [Crossref]
169. Cheng-Hung Chen. 2014. Compensatory neural fuzzy networks with rule-based cooperative differential evolution for nonlinear system control. *Nonlinear Dynamics* **75**:1-2, 355-366. [Crossref]
170. Ivan Blecic, Arnaldo Cecchini, Giuseppe A. Trunfio. 2014. Fast and Accurate Optimization of a GPU-accelerated CA Urban Model through Cooperative Coevolutionary Particle Swarms. *Procedia Computer Science* **29**, 1631-1643. [Crossref]
171. Lianbo Ma, Kunyuan Hu, Yunlong Zhu, Ben Niu, Hanning Chen, Maowei He. 2014. Discrete and Continuous Optimization Based on Hierarchical Artificial Bee Colony Optimizer. *Journal of Applied Mathematics* **2014**, 1-20. [Crossref]
172. Kittipong Boonlong. 2014. Vibration-Based Damage Detection in Beams by Cooperative Coevolutionary Genetic Algorithm. *Advances in Mechanical Engineering* **6**, 624949. [Crossref]
173. Lianbo Ma, Hanning Chen, Kunyuan Hu, Yunlong Zhu. 2014. Hierarchical Artificial Bee Colony Algorithm for RFID Network Planning Optimization. *The Scientific World Journal* **2014**, 1-21. [Crossref]
174. Maowei He, Kunyuan Hu, Yunlong Zhu, Lianbo Ma, Hanning Chen, Yan Song. 2014. Hierarchical Artificial Bee Colony Optimizer with Divide-and-Conquer and Crossover for Multilevel Threshold Image Segmentation. *Discrete Dynamics in Nature and Society* **2014**, 1-22. [Crossref]
175. XueJun Zhang, XiangMin Guan, Inseok Hwang, KaiQuan Cai. 2013. A hybrid distributed-centralized conflict resolution approach for multi-aircraft based on cooperative co-evolutionary. *Science China Information Sciences* **56**:12, 1-16. [Crossref]
176. Bassem S. Rabil, Safa Tliba, Eric Granger, Robert Sabourin. 2013. Securing high resolution grayscale facial captures using a blockwise coevolutionary GA. *Expert Systems with Applications* **40**:17, 6693-6706. [Crossref]
177. Yu Liang, Yu Liu, Liang Zhang. An improved artificial bee colony (ABC) algorithm for large scale optimization 644-648. [Crossref]

178. Cheng-Hung Chen, Wen-Hsien Chen. Cooperatively coevolving differential evolution for compensatory neural fuzzy networks 264-267. [Crossref]

179. Syahaneim Marzukhi, Will N. Browne, Mengjie Zhang. 2013. Adaptive artificial datasets through learning classifier systems for classification tasks. *Evolutionary Intelligence* **6**:2, 93-107. [Crossref]

180. Maxine Tan, Rudi Deklerck, Jan Cornelis, Bart Jansen. 2013. Phased searching with NEAT in a Time-Scaled Framework: Experiments on a computer-aided detection system for lung nodules. *Artificial Intelligence in Medicine* **59**:3, 157-167. [Crossref]

181. Max Salichon, Kagan Tumer. 2013. A neuro-evolutionary approach to control surface segmentation for micro aerial vehicles. *International Journal of General Systems* **42**:7, 793-805. [Crossref]

182. Weiping Ding, Jiandong Wang. 2013. A novel approach to minimum attribute reduction based on quantum-inspired self-adaptive cooperative co-evolution. *Knowledge-Based Systems* **50**, 1-13. [Crossref]

183. Feng-Zhe Cui, Lei Wang, Zhi-Zheng Xu, Xiu-Kun Wang, Hong-Fei Teng. 2013. Dual-system Cooperative Coevolutionary Differential Evolution Algorithm for Solving Nonseparable Function Optimization. *Information Technology Journal* **12**:9, 1796-1803. [Crossref]

184. M. D. Perez-Godoy, A. J. Rivera, M. J. Del Jesus, F. Martinez. A first analysis of the effect of local and global optimization weights methods in the cooperative-competitive design of RBFN for imbalanced environments 1-8. [Crossref]

185. KevinSim, EmmaHart, BenPaechter. 2013. Learning to Solve Bin Packing Problems with an Immune Inspired Hyper-heuristic. *The 2018 Conference on Artificial Life: A Hybrid of the European Conference on Artificial Life (ECAL) and the International Conference on the Synthesis and Simulation of Living Systems (ALIFE)* 856-863. [Citation] [PDF] [PDF Plus]

186. Yu-Jun Zheng, Hai-Feng Ling. 2013. Emergency transportation planning in disaster relief supply chain management: a cooperative fuzzy optimization approach. *Soft Computing* **17**:7, 1301-1314. [Crossref]

187. Ilhem Boussaïd, Julien Lepagnot, Patrick Siarry. 2013. A survey on optimization metaheuristics. *Information Sciences* **237**, 82-117. [Crossref]

188. Jean-François Connolly, Eric Granger, Robert Sabourin. 2013. Dynamic multi-objective evolution of classifier ensembles for video face recognition. *Applied Soft Computing* **13**:6, 3149-3166. [Crossref]

189. Ruxandra Stoean, Catalin Stoean. 2013. Modeling medical decision making by support vector machines, explaining by rules of evolutionary algorithms with feature selection. *Expert Systems with Applications* **40**:7, 2677-2686. [Crossref]

190. Wenxiang Chen, Ke Tang. Impact of problem decomposition on Cooperative Coevolution 733-740. [Crossref]

191. Shigang Yue, F. C. Rind. 2013. Redundant Neural Vision Systems—Competing for Collision Recognition Roles. *IEEE Transactions on Autonomous Mental Development* **5**:2, 173-186. [Crossref]

192. David B. Knoester, Heather J. Goldsby, Philip K. McKinley. 2013. Genetic Variation and the Evolution of Consensus in Digital Organisms. *IEEE Transactions on Evolutionary Computation* **17**:3, 403-417. [Crossref]

193. Rahul Kumar, Samantha Kohli, Perwez Alam, Ritankur Barkotoky, Mohit Gupta, Sanjay Tyagi, S. K. Jain, M. A. Qadar Pasha. 2013. Interactions between the FTO and GNB3 Genes Contribute to Varied Clinical Phenotypes in Hypertension. *PLoS ONE* **8**:5, e63934. [Crossref]

194. Cheng-Hung Chen, Sheng-Yen Yang. 2013. A knowledge-based cooperative differential evolution for neural fuzzy inference systems. *Soft Computing* **17**:5, 883-895. [Crossref]

195. A.J. Rivera, B. García-Domingo, M.J. del Jesus, J. Aguilera. 2013. Characterization of Concentrating Photovoltaic modules by cooperative competitive Radial Basis Function Networks. *Expert Systems with Applications* **40**:5, 1599-1608. [Crossref]

196. Wenjing Zhao, Sameer Alam, Hussein A. Abbass. 2013. Evaluating ground–air network vulnerabilities in an integrated terminal maneuvering area using co-evolutionary computational red teaming. *Transportation Research Part C: Emerging Technologies* **29**, 32-54. [Crossref]

197. Erandi Lakshika, Michael Barlow, Adam Easton. Co-evolving semi-competitive interactions of sheepdog herding behaviors utilizing a simple rule-based multi agent framework 82-89. [Crossref]

198. Weiping Ding, Quan Shi, Senbo Chen, Zhijin Guan, Jiandong Wang. A novel quantum cooperative co-evolutionary algorithm for large-scale minimum attribute reduction optimization 280-286. [Crossref]

199. G. S. Nitschke, S. M. Tolkamp. Approaches to dynamic team sizes 66-73. [Crossref]

200. Pawel Lichocki, Steffen Wischmann, Laurent Keller, Dario Floreano. 2013. Evolving Team Compositions by Agent Swapping. *IEEE Transactions on Evolutionary Computation* **17**:2, 282-298. [Crossref]

201. Padhraig Gormley, Kang Li, Olaf Wolkenhauer, George W. Irwin, Dajun Du. 2013. Reverse Engineering of Biochemical Reaction Networks Using Co-evolution with Eng-Genes. *Cognitive Computation* **5**:1, 106-118. [Crossref]

202. Wente Zeng, Mo-Yuen Chow. 2013. Modeling and Optimizing the Performance-Security Tradeoff on D-NCS Using the Coevolutionary Paradigm. *IEEE Transactions on Industrial Informatics* **9**:1, 394-402. [Crossref]

203. Olivier Barrière, Evelyne Lutton, Pierre-Henri Wuillemin, Cédric Baudrit, Mariette Sicard, Nathalie Perrot. Cooperative Coevolution for Agrifood Process Modeling 247-287. [Crossref]

204. Jing Su, Xuejun Zhang, Xiangmin Guan. 4D-Trajectory Conflict Resolution Using Cooperative Coevolution 387-395. [Crossref]

205. Thomas Jansen. Select Topics in the Analysis of Evolutionary Algorithms 157-236. [Crossref]

206. François Legillon, Arnaud Liefooghe, El-Ghazali Talbi. CoBRA: A Coevolutionary Metaheuristic for Bi-level Optimization 95-114. [Crossref]

207. Jianhua Yang, Yabo Liu, Zhaohui Wu, Min Yao. 2012. The Evolution of Cooperative Behaviours in Physically Heterogeneous Multi-Robot Systems. *International Journal of Advanced Robotic Systems* **9**:6, 253. [Crossref]

208. G. S. Nitschke, A. E. Eiben, M. C. Schut. 2012. Evolving team behaviors with specialization. *Genetic Programming and Evolvable Machines* **13**:4, 493-536. [Crossref]

209. Yang Sun, Lingbo Zhang, Xingsheng Gu. 2012. A hybrid co-evolutionary cultural algorithm based on particle swarm optimization for solving global optimization problems. *Neurocomputing* **98**, 76-89. [Crossref]

210. Konstantinos E. Parsopoulos. 2012. Parallel cooperative micro-particle swarm optimization: A master–slave model. *Applied Soft Computing* **12**:11, 3552-3579. [Crossref]

211. Miguel Arturo Barreto-Sanz, Alexandre Bujard, Carlos Andres Pena-Reyes. Evolving very-compact fuzzy models for gene expression data analysis 356-361. [Crossref]

212. Hongliang Liu, Enda Howley, Jim Duggan. 2012. Co-evolutionary analysis: a policy exploration method for system dynamics models. *System Dynamics Review* **28**:4, 361-369. [Crossref]

213. Haoyang Chen, Yasukuni Mori, Ikuo Matsuba. Archive-shared cooperative coevolutionary algorithm using Nash equilibria preservation 16-20. [Crossref]

214. Yujun Zheng, Xinli Xu, Shengyong Chen, Wanliang Wang. Distributed agent based cooperative differential evolution: A master-slave model 376-380. [Crossref]

215. Manuel Avalos Godoy, Arturo Ferreira Duarte, Christian Von Lucken, Enrique Davalos. Radial Basis Neural Network design using a competitive cooperative coevolutionary multiobjective algorithm 1-9. [Crossref]

216. Alejandro Sosa-Ascencio, Manuel Valenzuela-Rendon, Hugo Terashima-Marin. Cooperative Coevolution of Automatically Defined Functions with Gene Expression Programming 89-94. [Crossref]

217. J. Derrac, I. Triguero, S. Garcia, F. Herrera. 2012. Integrating Instance Selection, Instance Weighting, and Feature Weighting for Nearest Neighbor Classifiers by Coevolutionary Algorithms. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* **42**:5, 1383-1397. [Crossref]

218. Thomas Weise, Raymond Chiong, Ke Tang. 2012. Evolutionary Optimization: Pitfalls and Booby Traps. *Journal of Computer Science and Technology* **27**:5, 907-936. [Crossref]

219. Rohitash Chandra, Marcus Frean, Mengjie Zhang. 2012. Crossover-based local search in cooperative co-evolutionary feedforward neural networks. *Applied Soft Computing* **12**:9, 2924-2932. [Crossref]

220. Haoyang Chen, Yasukuni Mori, Ikuo Matsuba. Evolutionary Approach to Balance Problem of On-Line Action Role-Playing Game 1-4. [Crossref]

221. Piotr Szymak, Tomasz Praczyk. Using neural-evolutionary-fuzzy algorithm for anti-collision system of Unmanned Surface Vehicle 286-290. [Crossref]

222. Nate Kohl, Risto Miikkulainen. 2012. An Integrated Neuroevolutionary Approach to Reactive Control and High-Level Strategy. *IEEE Transactions on Evolutionary Computation* **16**:4, 472-488. [Crossref]

223. Dan Zhang, Zhen Gao. 2012. Optimal Kinematic Calibration of Parallel Manipulators With Pseudoerror Theory and Cooperative Coevolutionary Network. *IEEE Transactions on Industrial Electronics* **59**:8, 3221-3231. [Crossref]

224. Jean-François Connolly, Eric Granger, Robert Sabourin. 2012. Evolution of heterogeneous ensembles through dynamic particle swarm optimization for video-based face recognition. *Pattern Recognition* **45**:7, 2460-2477. [Crossref]

225. Wente Zeng, Mo-Yuen Chow. 2012. Optimal Tradeoff Between Performance and Security in Networked Control Systems Based on Coevolutionary Algorithms. *IEEE Transactions on Industrial Electronics* **59**:7, 3016-3025. [Crossref]

226. Qiang Wang, Jian Sun, Ling Yun Wang, Dao Zheng Liao. 2012. Pitch Angle Control Based Improved Cooperative Co-Evolution in Power Generation System. *Applied Mechanics and Materials* **190-191**, 1162-1165. [Crossref]

227. E. Parras-Gutierrez, M. Garcia-Arenas, V. M. Rivas, M. J. del Jesus. 2012. Coevolution of lags and RBFNs for time series forecasting: L-Co-R algorithm. *Soft Computing* **16**:6, 919-942. [Crossref]
228. Rohitash Chandra, Marcus Frean, Mengjie Zhang. 2012. Adapting modularity during learning in cooperative co-evolutionary recurrent neural networks. *Soft Computing* **16**:6, 1009-1020. [Crossref]
229. Rohitash Chandra, Mengjie Zhang. 2012. Cooperative coevolution of Elman recurrent neural networks for chaotic time series prediction. *Neurocomputing* **86**, 116-123. [Crossref]
230. Rohitash Chandra, Marcus Frean, Mengjie Zhang. 2012. On the issue of separability for problem decomposition in cooperative neuro-evolution. *Neurocomputing* **87**, 33-40. [Crossref]
231. Sune S. Nielsen, Bernabe Dorronsoro, Gregoire Danoy, Pascal Bouvry. Novel efficient asynchronous cooperative co-evolutionary multi-objective algorithms 1-7. [Crossref]
232. Eman Sayed, Daryl Essam, Ruhul Sarker. Dependency Identification technique for large scale optimization problems 1-8. [Crossref]
233. Geoff Nitschke. Behavioral heterogeneity, cooperation, and collective construction 1-8. [Crossref]
234. Francois Legillon, Arnaud Liefooghe, El-Ghazali Talbi. CoBRA: A cooperative coevolutionary algorithm for bi-level optimization 1-8. [Crossref]
235. O. Ibáñez, O. Cordón, S. Damas. 2012. A cooperative coevolutionary approach dealing with the skull–face overlay uncertainty in forensic identification by craniofacial superimposition. *Soft Computing* **16**:5, 797-808. [Crossref]
236. Folkert K. de Boer, Paulien Hogeweg. 2012. Co-evolution and ecosystem based problem solving. *Ecological Informatics* **9**, 47-58. [Crossref]
237. A. Mochon, Y. Saez, J.L. Gomez-Barroso, P. Isasi. 2012. Exploring pricing rules in combinatorial sealed-bid auctions. *Journal of Economic Behavior & Organization* **82**:2-3, 462-478. [Crossref]
238. Wente Zeng, Mo-Yuen Chow. CGA based performance-security trade-off optimization in a networked DC motor system 1834-1839. [Crossref]
239. Nicolás García-Pedrajas, Aida de Haro-García. 2012. Scaling up data mining algorithms: review and taxonomy. *Progress in Artificial Intelligence* **1**:1, 71-87. [Crossref]
240. Jin Tian, Minqiang Li, Fuzan Chen, Jisong Kou. 2012. Coevolutionary learning of neural network ensemble for complex classification tasks. *Pattern Recognition* **45**:4, 1373-1385. [Crossref]
241. John A. Doucette, Andrew R. McIntyre, Peter Lichodzijewski, Malcolm I. Heywood. 2012. Symbiotic coevolutionary genetic programming: a benchmarking study under large attribute spaces. *Genetic Programming and Evolvable Machines* **13**:1, 71-101. [Crossref]
242. Liang Sun, Shinichi Yoshida, Xiaochun Cheng, Yanchun Liang. 2012. A cooperative particle swarm optimizer with statistical variable interdependence learning. *Information Sciences* **186**:1, 20-39. [Crossref]
243. Rahul Kala. 2012. Multi-robot path planning using co-evolutionary genetic programming. *Expert Systems with Applications* **39**:3, 3817-3831. [Crossref]
244. G.S. Nitschke, M.C. Schut, A.E. Eiben. 2012. Evolving behavioral specialization in robot teams to solve a collective construction task. *Swarm and Evolutionary Computation* **2**, 25-38. [Crossref]
245. Je-Gun Joung, Soo-Jin Kim, Soo-Yong Shin, Byoung-Tak Zhang. 2012. A probabilistic coevolutionary biclustering algorithm for discovering coherent patterns in gene expression dataset. *BMC Bioinformatics* **13**:Suppl 17, S12. [Crossref]
246. Xinchang Hao, Hao Wen Lin, Xili Chen, Tomohiro Murata. 2012. Cooperative Bayesian Optimization Algorithm: a Novel Approach to Multiple Resources Scheduling Problem. *IEEJ Transactions on Electronics, Information and Systems* **132**:12, 2007-2018. [Crossref]
247. Haoyang Chen, Yasukuni Mori, Ikuo Matsuba. 2012. A Competitive Markov Approach to the Optimal Combat Strategies of On-Line Action Role-Playing Game Using Evolutionary Algorithms. *Journal of Intelligent Learning Systems and Applications* **04**:03, 176-187. [Crossref]
248. Haoyang Chen, Yasukuni Mori, Ikuo Matsuba. 2012. Solving the Balance Problem of On-Line Role-Playing Games Using Evolutionary Algorithms. *Journal of Software Engineering and Applications* **05**:08, 574-582. [Crossref]
249. Elena Popovici, Anthony Bucci, R. Paul Wiegand, Edwin D. De Jong. Coevolutionary Principles 987-1033. [Crossref]
250. Oscar Cordón. A Historical Review of Mamdani-Type Genetic Fuzzy Systems 73-90. [Crossref]
251. Min Shi. Natural vs. Unnatural Decomposition in Cooperative Coevolution 138-147. [Crossref]
252. Shimon Whiteson. Evolutionary Computation for Reinforcement Learning 325-355. [Crossref]

253. Joaquín Derrac, Isaac Triguero, Salvador García, Francisco Herrera. A Co-evolutionary Framework for Nearest Neighbor Enhancement: Combining Instance and Feature Weighting with Instance Selection 176-187. [Crossref]

254. Juan Villegas-Cortez, Gustavo Olague, Humberto Sossa, Carlos Avilés. Evolutionary Associative Memories through Genetic Programming 171-188. [Crossref]

255. Kenneth De Jong. Generalized Evolutionary Algorithms 625-635. [Crossref]

256. Aaron Scoble, Mark Johnston, Mengjie Zhang. Local Search in Parallel Linear Genetic Programming for Multiclass Classification 373-384. [Crossref]

257. Eman Sayed, Daryl Essam, Ruhul Sarker. Using Hybrid Dependency Identification with a Memetic Algorithm for Large Scale Optimization Problems 168-177. [Crossref]

258. Alexander Fölling, Christian Grimme, Joachim Lepping, Alexander Papaspyrou. 2011. Connecting Community-Grids by supporting job negotiation with coevolutionary Fuzzy-Systems. *Soft Computing* **15**:12, 2375-2387. [Crossref]

259. X. Hao, X. Chen, H.W. Lin, T. Murata. Cooperative Bayesian Optimization Algorithm: A Novel Approach to Simultaneous Multiple Resources Scheduling Problem 212-217. [Crossref]

260. Nicolás García-Pedrajas. 2011. Evolutionary computation for training set selection. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* **1**:6, 512-523. [Crossref]

261. Matthieu Weber, Ferrante Neri, Ville Tirronen. 2011. Shuffle or update parallel differential evolution for large-scale optimization. *Soft Computing* **15**:11, 2089-2107. [Crossref]

262. Xiao Mingming, Zhang Jun, Cai Kaiquan, Cao Xianbin, Tang Ke. Cooperative Co-evolution with Weighted Random Grouping for Large-Scale Crossing Waypoints Locating in Air Route Network 215-222. [Crossref]

263. Silvia Curteanu, Hugh Cartwright. 2011. Neural networks applied in chemistry. I. Determination of the optimal topology of multilayer perceptron neural networks. *Journal of Chemometrics* **25**:10, 527-549. [Crossref]

264. Rohitash Chandra, Marcus Frean, Mengjie Zhang, Christian W. Omlin. 2011. Encoding subcomponents in cooperative co-evolutionary recurrent neural networks. *Neurocomputing* **74**:17, 3223-3234. [Crossref]

265. Tomasz Praczyk, Piotr Szymak. 2011. Decision system for a team of autonomous underwater vehicles—Preliminary report. *Neurocomputing* **74**:17, 3323-3334. [Crossref]

266. Jian Sun, Qiang Wang, Jian Xiu Xiao. 2011. A Cooperative Co-Evolutionary Controller for AC Induction Motor. *Applied Mechanics and Materials* **128-129**, 771-774. [Crossref]

267. Lei Fang, Sheng-Uei Guan, Haofan Zhang. 2011. Recursive Learning of Genetic Algorithms with Task Decomposition and Varied Rule Set. *International Journal of Applied Evolutionary Computation* **2**:4, 1-24. [Crossref]

268. X.B. Cao, Y.W. Xu, C.X. Wei, Y.P. Guo. 2011. Feature subset selection based on co-evolution for pedestrian detection. *Transactions of the Institute of Measurement and Control* **33**:7, 867-879. [Crossref]

269. Bruno H. G. Barbosa, Lam T. Bui, Hussein A. Abbass, Luis A. Aguirre, Antônio P. Braga. 2011. The use of coevolution and the artificial immune system for ensemble learning. *Soft Computing* **15**:9, 1735-1747. [Crossref]

270. Alexander Scheidler, Martin Middendorf. 2011. Learning classifier systems to evolve classification rules for systems of memory constrained components. *Evolutionary Intelligence* **4**:3, 127-143. [Crossref]

271. H. Huang, M. Pasquier, C. Quek. 2011. Decision support system based on hierarchical co-evolutionary fuzzy approach: A case study in detecting gamma ray signals. *Expert Systems with Applications* **38**:9, 10719-10729. [Crossref]

272. Oscar Cordón. 2011. A historical review of evolutionary learning methods for Mamdani-type fuzzy rule-based systems: Designing interpretable genetic fuzzy systems. *International Journal of Approximate Reasoning* **52**:6, 894-913. [Crossref]

273. Bin Jiao, Qunxian Chen, Shaobin Yan. 2011. A Cooperative Co-evolution PSO for Flow Shop Scheduling Problem with Uncertainty. *Journal of Computers* **6**:9. . [Crossref]

274. Rohitash Chandra, Marcus Frean, Mengjie Zhang. A memetic framework for cooperative coevolution of recurrent neural networks 673-680. [Crossref]

275. Rohitash Chandra, Marcus Frean, Mengjie Zhang. Modularity adaptation in cooperative coevolution of feedforward neural networks 681-688. [Crossref]

276. Antonio J. Rivera, Pedro Pérez-Recuerda, María Dolores Pérez-Godoy, María Jose del Jesús, María Pilar Frías, Manuel Parras. 2011. A study on the medium-term forecasting using exogenous variable selection of the extra-virgin olive oil with soft computing methods. *Applied Intelligence* **34**:3, 331-346. [Crossref]

277. Maria Battarra, Stefano Benedettini, Andrea Roli. 2011. Leveraging saving-based algorithms by master–slave genetic algorithms. *Engineering Applications of Artificial Intelligence* **24**:4, 555-566. [Crossref]

278. Eddy Parkinson, Adam Ghandar, Zbigniew Michalewicz, Andrew Tuson. Estimating the reproductive potential of offspring in evolutionary heuristics for combinatorial optimization problems 172-178. [Crossref]

279. C. K. Goh, D. Lim, L. Ma, Y. S. Ong, P. S. Dutta. A surrogate-assisted memetic co-evolutionary algorithm for expensive constrained optimization problems 744-749. [Crossref]

280. Yun Wen, Hua Xu. A cooperative coevolution-based pittsburgh learning classifier system embedded with memetic feature selection 2415-2422. [Crossref]

281. Taha Mansouri, Alireza Farasat, Mohammad B. Menhaj, Mohammad Reza Sadeghi Moghadam. 2011. ARO: A new model free optimization algorithm for real time applications inspired by the asexual reproduction. *Expert Systems with Applications* **38**:5, 4866-4874. [Crossref]

282. Myriam Abramson, Ranjeev Mittu. Learning and coordination: An overview 343-350. [Crossref]

283. Catalin Stoean, Ruxandra Stoean, Monica Lupsor, Horia Stefanescu, Radu Badea. 2011. Feature selection for a cooperative coevolutionary classifier in liver fibrosis diagnosis. *Computers in Biology and Medicine* **41**:4, 238-246. [Crossref]

284. Dan Zhang, Zhen Gao, Peigang Jiang. A novel calibration method of parallel kinematic manipulators based on multi-population coevolutionary neural network 205-211. [Crossref]

285. John Cartlidge, Djamel Ait-Boudaoud. 2011. Autonomous Virulence Adaptation Improves Coevolutionary Optimization. *IEEE Transactions on Evolutionary Computation* **15**:2, 215-229. [Crossref]

286. Anmer Daskin, Sabre Kais. 2011. Group leaders optimization algorithm. *Molecular Physics* **109**:5, 761-772. [Crossref]

287. Adam Campbell, Annie S. Wu. 2011. Multi-agent role allocation: issues, approaches, and multiple perspectives. *Autonomous Agents and Multi-Agent Systems* **22**:2, 317-355. [Crossref]

288. Yan Yang, Haoyong Chen, Yao Zhang, Zhiqiang Jin, Fangxing Li. Searching for Electricity Market Equilibrium Using Coevolutionary Computation Approach 1-5. [Crossref]

289. Qi Feng Tang, Liang Zhao, Rong Bin Qi, Hui Cheng, Feng Qian. 2011. Tuning the Structure and Parameters of a Neural Network by Using Cooperative Quantum Particle Swarm Algorithm. *Applied Mechanics and Materials* **48-49**, 1328-1332. [Crossref]

290. Qun Xian Chen, Bin Jiao, Shao Bin Yan. 2011. Flow Shop Production Scheduling under Uncertainty within Infinite Intermediate Storage. *Advanced Materials Research* **204-210**, 777-783. [Crossref]

291. Angus Wu, Jin Zhang, Henry Chung. 2011. Decoupled optimal design for power electronic circuits with adaptive migration in coevolutionary environment. *Applied Soft Computing* **11**:1, 23-31. [Crossref]

292. Yong Zhang, Xiao-bei Wu, Zong-yi Xing, Wei-Li Hu. 2011. On generating interpretable and precise fuzzy systems based on Pareto multi-objective cooperative co-evolutionary algorithm. *Applied Soft Computing* **11**:1, 1284-1294. [Crossref]

293. Lindsay Hanna Landry, Jonathan Cagan. 2011. Search Strategies in Evolutionary Multi-Agent Systems: The Effect of Cooperation and Reward on Solution Quality. *Journal of Mechanical Design* **133**:6, 061005. [Crossref]

294. Heather J. Goldsby, David B. Knoester, Jeff Clune, Philip K. McKinley, Charles Ofria. The Evolution of Division of Labor 10-18. [Crossref]

295. Steve DiPaola, Nathan Sorenson. CGP, Creativity and Art 293-307. [Crossref]

296. Lifeng Ai, Maolin Tang, Colin Fidge. Resource Allocation and Scheduling of Multiple Composite Web Services in Cloud Computing Using Cooperative Coevolution Genetic Algorithm 258-267. [Crossref]

297. Liviu Panait. 2010. Theoretical Convergence Guarantees for Cooperative Coevolutionary Algorithms. *Evolutionary Computation* **18**:4, 581-615. [Abstract] [PDF] [PDF Plus]

298. Christian Vecchiola, Mani Abedini, Michael Kirley, Xingchen Chu, Rajkumar Buyya. Gene Expression Classification with a Novel Coevolutionary Based Learning Classifier System on Public Clouds 92-97. [Crossref]

299. STEFAN SCHLIEBS, NIKOLA KASABOV, MICHAËL DEFOIN-PLATEL. 2010. ON THE PROBABILISTIC OPTIMIZATION OF SPIKING NEURAL NETWORKS. *International Journal of Neural Systems* **20**:06, 481-500. [Crossref]

300. María Dolores Pérez-Godoy, Alberto Fernández, Antonio Jesús Rivera, María José del Jesus. 2010. Analysis of an evolutionary RBFN design algorithm, CO2RBFN, for imbalanced data sets. *Pattern Recognition Letters* **31**:15, 2375-2388. [Crossref]

301. Lara Menezes, Andre L. V. Coelho. Tackling the biclustering problem with cooperative coevolutionary algorithms 1189-1194. [Crossref]

302. Jin Tian, Minqiang Li, Fuzan Chen. 2010. Dual-population based coevolutionary algorithm for designing RBFNN with feature selection. *Expert Systems with Applications* **37**:10, 6904-6918. [Crossref]

303. X. Hao, H. W. Lin, T. Murata. Application of Negotiable Evolutionary Algorithm in flexible manufacturing planning and scheduling 496-500. [Crossref]

304. Alireza Farasat, Mohammad B. Menhaj, Taha Mansouri, Mohammad Reza Sadeghi Moghadam. 2010. ARO: A new model-free optimization algorithm inspired from asexual reproduction. *Applied Soft Computing* **10**:4, 1284-1292. [Crossref]

305. Xuchu Dong, HaiHong Yu, Dantong Ouyang, Dianbo Cai, Yuxin Ye, Yonggang Zhang. Cooperative coevolutionary genetic algorithms to find optimal elimination orderings for Bayesian networks 1388-1394. [Crossref]

306. Kamal Adamu, Steve Phelps. An empirical study of collaboration methods for coevolving technical trading rules 1-6. [Crossref]

307. Aditya Rawal, Padmini Rajagopalan, Risto Miikkulainen. Constructing competitive and cooperative agent behavior using coevolution 107-114. [Crossref]

308. Peter Durr, Claudio Mattiussi, Dario Floreano. 2010. Genetic Representation and Evolvability of Modular Neural Controllers. *IEEE Computational Intelligence Magazine* **5**:3, 10-19. [Crossref]

309. Roman B. Sergienko, Eugene S. Semenkin. Competitive cooperation for strategy adaptation in coevolutionary genetic algorithm for constrained optimization 1-6. [Crossref]

310. Marcus Furuholmen, Kyrre Glette, Mats Hovin, Jim Torresen. A Coevolutionary, Hyper Heuristic approach to the optimization of Three-dimensional Process Plant Layouts — A comparative study 1-8. [Crossref]

311. Mani Abedini, Michael Kirley. A multiple population XCS: Evolving condition-action rules based on feature space partitions 1-8. [Crossref]

312. Maria Dolores Perez-Godoy, Antonio J. Rivera, Cristobal J. Carmona, Maria Jose del Jesus. A preliminary study on mutation operators in cooperative competitive algorithms for RBFN design 1-7. [Crossref]

313. Shimon Whiteson, Matthew E. Taylor, Peter Stone. 2010. Critical factors in the empirical performance of temporal difference and evolutionary methods for reinforcement learning. *Autonomous Agents and Multi-Agent Systems* **21**:1, 1-35. [Crossref]

314. María D. Perez-Godoy, Antonio J. Rivera, Francisco J. Berlanga, María José Del Jesus. 2010. CO2RBFN: an evolutionary cooperative–competitive RBFN design algorithm for classification problems. *Soft Computing* **14**:9, 953-971. [Crossref]

315. Yongming Li, Xiaoping Zeng. 2010. Multi-population co-genetic algorithm with double chain-like agents structure for parallel global numerical optimization. *Applied Intelligence* **32**:3, 292-310. [Crossref]

316. Yongming Li, Xiaoping Zeng, Liang Han, Pin Wang. 2010. Two coding based adaptive parallel co-genetic algorithm with double agents structure. *Engineering Applications of Artificial Intelligence* **23**:4, 526-542. [Crossref]

317. Joaquín Derrac, Salvador García, Francisco Herrera. 2010. IFS-CoCo: Instance and feature selection based on cooperative coevolution with nearest neighbor rule. *Pattern Recognition* **43**:6, 2082-2105. [Crossref]

318. Xin Ma, Hong-xiao Wu. Power system short-term load forecasting based on cooperative co-evolutionary immune network model V1-582-V1-585. [Crossref]

319. Tomasz Praczyk. 2010. Assembler Encoding versus Connectivity Matrix Encoding in the Inverted Pendulum Problem with a Hidden State. *Solid State Phenomena* **164**, 233-238. [Crossref]

320. C.K. Goh, K.C. Tan, D.S. Liu, S.C. Chiam. 2010. A competitive and cooperative co-evolutionary approach to multi-objective particle swarm optimization algorithm design. *European Journal of Operational Research* **202**:1, 42-54. [Crossref]

321. Zhen Gao, Dan Zhang, Yunjian Ge. 2010. Design optimization of a spatial six degree-of-freedom parallel manipulator based on artificial intelligence approaches. *Robotics and Computer-Integrated Manufacturing* **26**:2, 180-189. [Crossref]

322. G. S. Nitschke, M. C. Schut, A. E. Eiben. 2010. Collective neuro-evolution for evolving specialized sensor resolutions in a multi-rover task. *Evolutionary Intelligence* **3**:1, 13-29. [Crossref]

323. Nicolás García-Pedrajas, Juan Antonio Romero del Castillo, Domingo Ortiz-Boyer. 2010. A cooperative coevolutionary algorithm for instance selection for instance-based learning. *Machine Learning* **78**:3, 381-420. [Crossref]

324. A.-M. Farahmand, M.N. Ahmadabadi, C. Lucas, B.N. Araabi. 2010. Interaction of Culture-Based Learning and Cooperative Co-Evolution and its Application to Automatic Behavior-Based System Design. *IEEE Transactions on Evolutionary Computation* **14**:1, 23-57. [Crossref]

325. Rob Dekkers. A Co-evolutionary Perspective on Distributed Manufacturing 29-50. [Crossref]

326. John Doucette, Peter Lichodzijewski, Malcolm Heywood. Evolving Coevolutionary Classifiers Under Large Attribute Spaces 37-54. [Crossref]

327. Andrea Caponio, Anna V. Kononova, Ferrante Neri. Differential Evolution with Scale Factor Local Search for Large Scale Problems 297-323. [Crossref]

328. Juan Villegas-Cortez, Gustavo Olague, Carlos Aviles, Humberto Sossa, Andres Ferreyra. Automatic Synthesis of Associative Memories through Genetic Programming: A First Co-evolutionary Approach 344-351. [Crossref]

329. Marcus Furuholmen, Kyrre Glette, Mats Hovin, Jim Torresen. Evolutionary Approaches to the Three-dimensional Multi-pipe Routing Problem: A Comparative Study Using Direct Encodings 71-82. [Crossref]

330. Kees Pieters. Computation in Complex Environments; 299-324. [Crossref]

331. Asuncion Mochon, Yago Saez, Jose Luis Gomez-Barroso, Pedro Isasi. Co-evolutionary Agents in Combinatorial Sealed-bid Auctions for Spectrum Licenses Markets 53-63. [Crossref]

332. Wenxiang Chen, Thomas Weise, Zhenyu Yang, Ke Tang. Large-Scale Global Optimization Using Cooperative Coevolution with Variable Interaction Learning 300-309. [Crossref]

333. Mitchell A. Potter, Christine Couldrey. A Cooperative Coevolutionary Approach to Partitional Clustering 374-383. [Crossref]

334. Steven Schockaert, Philip D. Smart. Generating Fuzzy Regions from Conflicting Spatial Information 211-239. [Crossref]

335. Rohitash Chandra, Marcus Frean, Mengjie Zhang. An Encoding Scheme for Cooperative Coevolutionary Feedforward Neural Networks 253-262. [Crossref]

336. Grégoire Danoy, Pascal Bouvry, Olivier Boissier. A Multi-Agent Organizational Framework for Coevolutionary Optimization 199-224. [Crossref]

337. Joaquín Derrac, Salvador García, Francisco Herrera. IFS-CoCo in the Landscape Contest: Description and Results 56-65. [Crossref]

338. Alexander Fölling, Christian Grimme, Joachim Lepping, Alexander Papaspyrou, Uwe Schwiegelshohn. 2009. Competitive Coevolutionary Learning of Fuzzy Systems for Job Exchange in Computational Grids. *Evolutionary Computation* **17**:4, 545-560. [Abstract] [PDF] [PDF Plus]

339. Bo Yang, Hongfeng Xiao. On Large Scale Evolutionary Optimization Using Simplex-Based Cooperative Coevolution Genetic Algorithm 1-5. [Crossref]

340. Yongming Li, Sujuan Zhang, Xiaoping Zeng. 2009. Research of multi-population agent genetic algorithm for feature selection. *Expert Systems with Applications* **36**:9, 11570-11581. [Crossref]

341. Igor Walter, Fernando Gomide. 2009. Multiagent coevolutionary genetic fuzzy system to develop bidding strategies in electricity markets: computational economics to assess mechanism design. *Evolutionary Intelligence* **2**:1-2, 53-71. [Crossref]

342. Yi-Shou Wang, Hong-Fei Teng, Yan-Jun Shi. 2009. Cooperative co-evolutionary scatter search for satellite module layout design. *Engineering Computations* **26**:7, 761-785. [Crossref]

343. C.H. Yong, R. Miikkulainen. 2009. Coevolution of Role-Based Cooperation <newline/>in Multiagent Systems. *IEEE Transactions on Autonomous Mental Development* **1**:3, 170-186. [Crossref]

344. Alexander Folling, Christian Grimme, Joachim Lepping, Alexander Papaspyrou. Co-evolving fuzzy rule sets for job exchange in computational grids 1683-1688. [Crossref]

345. Michail Maniadakis, Panos Trahanias. 2009. Agent-Based Brain Modeling by Means of Hierarchical Cooperative Coevolution. *Artificial Life* **15**:3, 293-336. [Abstract] [PDF] [PDF Plus]

346. Stefan Schliebs, Michaël Defoin-Platel, Sue Worner, Nikola Kasabov. 2009. Integrated feature and parameter optimization for an evolving spiking neural network: Exploring heterogeneous probabilistic models. *Neural Networks* **22**:5-6, 623-632. [Crossref]

347. Liu Qing, Wang Zengzing. Coordinated design of multiple FACTS controllers based on fuzzy immune co-evolutionary Algorithm 1-6. [Crossref]

348. Kenneth De Jong. 2009. Evolutionary computation. *Wiley Interdisciplinary Reviews: Computational Statistics* **1**:1, 52-56. [Crossref]

349. Siang Yew Chong, Christopher Hill, Xin Yao. Co-Evolutionary Learning Of Contextual Asymmetric Actors 827-833. [Crossref]

350. Markus Waibel, Laurent Keller, Dario Floreano. 2009. Genetic Team Composition and Level of Selection in the Evolution of Cooperation. *IEEE Transactions on Evolutionary Computation* **13**:3, 648-660. [Crossref]

351. Bao-feng YUAN, Le-hua WU, Wei ZENG. 2009. Image segmentation algorithm based on coevolution with texture and gray scale. *Journal of Computer Applications* **29**:1, 54-56. [Crossref]

352. Zhenyu Yang, Jingqiao Zhang, Ke Tang, Xin Yao, Arthur C. Sanderson. An adaptive coevolutionary Differential Evolution algorithm for large-scale optimization 102-109. [Crossref]

353. Taras Kowaliw, Wolfgang Banzhaf. Augmenting artificial development with local fitness 316-323. [Crossref]

354. G. S. Nitschke. Neuro-Evolution approaches to collective behavior 1554-1561. [Crossref]

355. Min Shi, Boye Annfelt Hoverstad. PEEC: Evolving efficient connections using Pareto optimality 1578-1584. [Crossref]

356. Chun-Kit Au, Ho-Fung Leung. Investigating collaboration methods of random immigrant scheme in cooperative coevolution 2701-2707. [Crossref]

357. Marcus Furuholmen, Kyrre Glette, Mats Hovin, Jim Torresen. Coevolving heuristics for the Distributor's Pallet Packing Problem 2810-2817. [Crossref]

358. R. Dekkers. 2009. Distributed Manufacturing as co-evolutionary system. *International Journal of Production Research* **47**:8, 2031-2054. [Crossref]

359. Jin Tian, Minqiang Li, Fuzan Chen. 2009. A hybrid classification algorithm based on coevolutionary EBFNN and domain covering method. *Neural Computing and Applications* **18**:3, 293-308. [Crossref]

360. Nate Kohl, Risto Miikkulainen. 2009. Evolving neural networks for strategic decision-making problems. *Neural Networks* **22**:3, 326-337. [Crossref]

361. Sherri Goings, Charles Ofria. Ecological approaches to diversity maintenance in evolutionary algorithms 124-130. [Crossref]

362. Myriam Regattieri Delgado, Elaine Yassue Nagai, Lúcia Valéria Ramos de Arruda. 2009. A neuro-coevolutionary genetic fuzzy system to design soft sensors. *Soft Computing* **13**:5, 481-495. [Crossref]

363. Haoming Huang, Michel Pasquier, Chai Quek. 2009. Financial Market Trading System With a Hierarchical Coevolutionary Fuzzy Predictive Model. *IEEE Transactions on Evolutionary Computation* **13**:1, 56-70. [Crossref]

364. Chi-Keong Goh, Kay Chen Tan. 2009. A Competitive-Cooperative Coevolutionary Paradigm for Dynamic Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation* **13**:1, 103-127. [Crossref]

365. MICHAIL MANIADAKIS, PANOS TRAHANIAS. 2009. HIERARCHICAL COOPERATIVE CoEVOLUTION: PRESENTATION AND ASSESSMENT STUDY. *International Journal on Artificial Intelligence Tools* **18**:01, 99-120. [Crossref]

366. G. M. Behery, A. A. El-Harby, Mostafa Y. El-Bakry. 2009. Reorganizing Neural Network System for Two Spirals and Linear Low-Density Polyethylene Copolymer Problems. *Applied Computational Intelligence and Soft Computing* **2009**, 1-11. [Crossref]

367. Lindsay Hanna, Jonathan Cagan. 2009. Evolutionary Multi-Agent Systems: An Adaptive and Dynamic Approach to Optimization. *Journal of Mechanical Design* **131**:1, 011010. [Crossref]

368. André Vargas Abs da Cruz, Carlos Hall Barbosa, Juan Guillermo Lazo Lazo, Karla Figueiredo, Luciana Faletti Almeida, Marco Aurélio Cavalcanti Pacheco, Marley Maria Bernardes Rebuzzi Vellasco, Yván Jesús Túpac Valdivia. Decision Support Methods 23-96. [Crossref]

369. Vahab Akbarzadeh, Alireza Sadeghian, Marcus V. dos Santos. Inducing Relational Fuzzy Classification Rules by Means of Cooperative Coevolution 127-147. [Crossref]

370. Yu Wang, Bin Li. A Self-adaptive Mixed Distribution Based Uni-variate Estimation of Distribution Algorithm for Large Scale Global Optimization 171-198. [Crossref]

371. Tse Guan Tan, Jason Teo. Improving the Performance of Multiobjective Evolutionary Optimization Algorithms Using Coevolutionary Learning 457-487. [Crossref]

372. Joaquín Derrac, Salvador García, Francisco Herrera. A First Study on the Use of Coevolutionary Algorithms for Instance and Feature Selection 557-564. [Crossref]

373. Catalin Stoean, Ruxandra Stoean. Evolution of Cooperating Classification Rules with an Archiving Strategy to Underpin Collaboration 47-65. [Crossref]

374. Zhi-hui Zhan, Jun Zhang. Parallel Particle Swarm Optimization with Adaptive Asynchronous Migration Strategy 490-501. [Crossref]

375. Abdellah Bedrouni, Ranjeev Mittu, Abdeslem Boukhtouta, Jean Berger. Learning and Coordination 125-139. [Crossref]

376. Fei Su, Yuan Li, Hui Peng, Lincheng Shen. Multi-UCAV Cooperative Path Planning Using Improved Coevolutionary Multi-Ant-Colony Algorithm 834-845. [Crossref]

377. Piotr Jędrzejowicz. A-Teams and Their Applications 36-50. [Crossref]

378. Mani Abedini, Michael Kirley. CoXCS: A Coevolutionary Learning Classifier Based on Feature Space Partitioning 360-369. [Crossref]

379. Yingjie Hu, Nikola Kasabov. Coevolutionary Method for Gene Selection and Parameter Optimization in Microarray Data Analysis 483-492. [Crossref]

380. Rafael Lahoz-Beltra, Gabriela Ochoa, Uwe Aickelin. 2009. Cheating for Problem Solving: A Genetic Algorithm with Social Interactions. *SSRN Electronic Journal* . [Crossref]

381. Fangming Zhu, Sheng-Uei Guan. 2008. Cooperative co-evolution of GA-based classifiers based on input decomposition. *Engineering Applications of Artificial Intelligence* **21**:8, 1360-1369. [Crossref]

382. F.J. Martínez-Estudillo, C. Hervás-Martínez, P.A. Gutiérrez, A.C. Martínez-Estudillo. 2008. Evolutionary product-unit neural networks classifiers. *Neurocomputing* **72**:1-3, 548-561. [Crossref]

383. M.D. Schmidt, H. Lipson. 2008. Coevolution of Fitness Predictors. *IEEE Transactions on Evolutionary Computation* **12**:6, 736-749. [Crossref]

384. Peter Lichodzijewski, Malcolm I. Heywood. 2008. Coevolutionary bid-based genetic programming for problem decomposition in classification. *Genetic Programming and Evolvable Machines* **9**:4, 331-365. [Crossref]

385. Fernanda L. Minku, Teresa B. Ludermir. 2008. Clustering and co-evolution to construct neural network ensembles: An experimental study. *Neural Networks* **21**:9, 1363-1379. [Crossref]

386. Jian Sun, Yongsheng Ding, Kuangrong Hao. A cooperative co-evolutionary control method for Stewart platform 528-532. [Crossref]

387. Limin Jia, Ruyan Zhang, Yong Zhang, Zongyi Xing, Guoqiang Cai. Approach of Fuzzy Classification Based on Hybrid Co-Evolution Algorithm 266-271. [Crossref]

388. Zhenyu Yang, Ke Tang, Xin Yao. 2008. Large scale evolutionary optimization using cooperative coevolution. *Information Sciences* **178**:15, 2985-2999. [Crossref]

389. Siang Yew Chong, P. Tino, Xin Yao. 2008. Measuring Generalization Performance in Coevolutionary Learning. *IEEE Transactions on Evolutionary Computation* **12**:4, 479-505. [Crossref]

390. Gang Li, Jin Feng Wang, Kin Hong Lee, Kwong-Sak Leung. 2008. Instruction-Matrix-Based Genetic Programming. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* **38**:4, 1036-1049. [Crossref]

391. Michail Maniadakis, Panos Trahanias. 2008. Hierarchical Co-evolution of Cooperating Agents Acting in the Brain-Arena. *Adaptive Behavior* **16**:4, 221-245. [Crossref]

392. Jiachuan Wang, Zhun Fan, Janis P. Terpenny, Erik D. Goodman. 2008. Cooperative body–brain coevolutionary synthesis of mechatronic systems. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **22**:3, 219-234. [Crossref]

393. A. Agogino, K. Tumer. 2008. Efficient Evaluation Functions for Evolving Coordination. *Evolutionary Computation* **16**:2, 257-288. [Abstract] [PDF] [PDF Plus]

394. Bi Li, Tu-Sheng Lin, Liang Liao, Ce Fan. Genetic algorithm based on multipopulation competitive coevolution 225-228. [Crossref]

395. S. Supudomchok, N. Chaiyaratana, C. Phalakomkule. Co-operative co-evolutionary approach for flux balance in Bacillus subtilis 1226-1231. [Crossref]

396. Haoming Huang, Michel Pasquier, Chai Quek. Application of a hierarchical coevolutionary fuzzy system for financial prediction and trading 1252-1259. [Crossref]

397. Zhenyu Yang, Ke Tang, Xin Yao. Multilevel cooperative coevolution for large scale optimization 1663-1670. [Crossref]

398. Chun-Kit Au, Ho-Fung Leung. On the behavior of cooperative coevolution in dynamic environments 2827-2836. [Crossref]

399. Minqiang Li, Jin Tian, Fuzan Chen. 2008. Improving multiclass pattern recognition with a co-evolutionary RBFNN. *Pattern Recognition Letters* **29**:4, 392-406. [Crossref]

400. Igor Walter, Fernando Gomide. Coevolutionary fuzzy multiagent bidding strategies in competitive electricity markets 53-58. [Crossref]

401. Sergio Guadarrama, Ruben Vazquez. Tuning a fuzzy racing car by coevolution 59-64. [Crossref]

402. Jinwei Gu, Xingsheng Gu, Bin Jiao. A Coevolutionary Genetic Based Scheduling Algorithm for stochastic flexible scheduling problem 4160-4165. [Crossref]

403. Rafał Dreżewski, Jan Sepielak. Evolutionary System for Generating Investment Strategies 83-92. [Crossref]

404. Rafał Dreżewski, Jan Sepielak, Leszek Siwik. Generating Robust Investment Strategies with Agent-Based Co-evolutionary System 664-673. [Crossref]

405. Tina Yu, Dave Wilkinson. A Co-Evolutionary Fuzzy System for Reservoir Well Logs Interpretation 199-218. [Crossref]

406. Malek Aichour, Evelyne Lutton. Cooperative Co-evolution Inspired Operators for Classical GP Schemes 169-178. [Crossref]

407. Rafał Dreżewski, Leszek Siwik. Co-Evolutionary Multi-Agent System for Portfolio Optimization 271-299. [Crossref]

408. MICHAIL MANIADAKIS, PANOS TRAHANIAS. 2007. MODELLING ROBOTIC COGNITIVE MECHANISMS BY HIERARCHICAL COOPERATIVE COEVOLUTION. *International Journal on Artificial Intelligence Tools* **16**:06, 935-966. [Crossref]

409. Luciana Faletti Almeida, Yván Túpac Valdivia, Marley Maria Bernardes Rebuzzi Vellasco, Marco Aurélio Cavalcanti Pacheco. 2007. Otimização de alternativas para o desenvolvimento de campos de petróleo. *Gestão & Produção* **14**:3, 489-503. [Crossref]

410. Dominic Searson, Mark Willis, Gary Montague. 2007. Co-evolution of non-linear PLS model components. *Journal of Chemometrics* **21**:12, 592-603. [Crossref]
411. Tomasz Praczyk. 2007. Evolving Co-Adapted Subcomponents in Assembler Encoding. *International Journal of Applied Mathematics and Computer Science* **17**:4, 549-563. [Crossref]
412. Chun-Kit Au, Ho-Fung Leung. Guided Mutations in Cooperative Coevolutionary Algorithms for Function Optimization 407-414. [Crossref]
413. Michail Maniadakis, Panos Trahanias. Assessing Hierarchical Cooperative CoEvolution 391-398. [Crossref]
414. Tse Guan Tan, Hui Keng Lau, Jason Teo. Cooperative coevolution for pareto multiobjective optimization: An empirical study using SPEA2 1-4. [Crossref]
415. Krzysztof Krawiec, Bir Bhanu. 2007. Visual Learning by Evolutionary and Coevolutionary Feature Synthesis. *IEEE Transactions on Evolutionary Computation* **11**:5, 635-650. [Crossref]
416. Chien-feng Huang, Jasleen Kaur, Ana Maguitman, Luis M. Rocha. 2007. Agent-Based Model of Genotype Editing. *Evolutionary Computation* **15**:3, 253-289. [Abstract] [PDF] [PDF Plus]
417. Chun-Kit Au, Ho-Fung Leung. Biasing mutations in cooperative coevolution 828-835. [Crossref]
418. Maciej A. Mazurowski, Jacek M. Zurada. Solving decentralized multi-agent control problems with genetic algorithms 1029-1034. [Crossref]
419. L. Polat, A. Acan, A. Unveren. Cooperative coevolutionary algorithms for fuzzy vehicular routing problem: An analysis of efficiency vs. geographical distribution 1126-1132. [Crossref]
420. D. J. Cornforth. An investigation into dynamic problem solving in a hybrid evolutionary market-based multi-agent system 1732-1739. [Crossref]
421. Bo Liu, Harman Ma, Xuejun Zhang, Yan Zhou. A memetic co-evolutionary differential evolution algorithm for constrained optimization 2996-3002. [Crossref]
422. C.H. Tan, C.K. Goh, K.C. Tan, A. Tay. A cooperative coevolutionary algorithm for multiobjective particle swarm optimization 3180-3186. [Crossref]
423. G.S. Nitschke, M.C. Schut, A.E. Eiben. Emergent specialization in the extended multi-rover problem 3410-3417. [Crossref]
424. Haoming Huang, Michel Pasquier, Chai Quek. HiCEFS — A hierarchical coevolutionary approach for the dynamic generation of fuzzy system 3426-3433. [Crossref]
425. Michael A. Lones, Andy M. Tyrrell. A co-evolutionary framework for regulatory motif discovery 3894-3901. [Crossref]
426. CHRISTIAN GAGNÉ, MARC PARIZEAU. 2007. COEVOLUTION OF NEAREST NEIGHBOR CLASSIFIERS. *International Journal of Pattern Recognition and Artificial Intelligence* **21**:05, 921-946. [Crossref]
427. P.A. Castillo, J.J. Merelo, M.G. Arenas, G. Romero. 2007. Comparing evolutionary hybrid systems for design and optimization of multilayer perceptron structure along training parameters. *Information Sciences* **177**:14, 2884-2905. [Crossref]
428. Moshe Sipper, Yaniv Azaria, Ami Hauptman, Yehonatan Shichel. 2007. Designing an Evolutionary Strategizing Machine for Game Playing and Beyond. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)* **37**:4, 583-593. [Crossref]
429. Garnett Wilson, Malcolm Heywood. 2007. Introducing probabilistic adaptive mapping developmental genetic programming with redundant mappings. *Genetic Programming and Evolvable Machines* **8**:2, 187-220. [Crossref]
430. Stephan K. Chalup, Lukasz Wiklendt. 2007. Variations of the two-spiral task. *Connection Science* **19**:2, 183-199. [Crossref]
431. J.-G. Joung, K.-B. Hwang, J.-W. Nam, S.-J. Kim, B.-T. Zhang. 2007. Discovery of microRNA mRNA modules via population-based probabilistic learning. *Bioinformatics* **23**:9, 1141-1147. [Crossref]
432. Dana Vrajitoru. Competitive coevolution versus objective fitness for an autonomous motorcycle pilot 557-562. [Crossref]
433. Y.P. Guo, X.B. Cao, Y.W. Xu, Q. Hong. Co-Evolution based Feature Selection for Pedestrian Detection 2797-2801. [Crossref]
434. Guo-Hui Yang, Qun Wu, Xiao-guang Hu, Yu Jiang. Learning Technique for TSK Fuzzy Model Based on Cooperative 2691-2696. [Crossref]
435. A. J. Rivera, I. Rojas, J. Ortega, M. J. del Jesus. 2007. A new hybrid methodology for cooperative-coevolutionary optimization of radial basis function networks. *Soft Computing* **11**:7, 655-668. [Crossref]
436. A.E. Eiben, G.S. Nitschke, M.C. Schut. Evolutionary Design of Specialization 417-424. [Crossref]
437. Leonardo M. Simao, Douglas M. Dias, Marco Aur lio C. Pacheco. Refinery Scheduling Optimization using Genetic Algorithms and Cooperative Coevolution 151-158. [Crossref]

438. Edwin D. de Jong. 2007. A Monotonic Archive for Pareto-Coevolution. *Evolutionary Computation* **15**:1, 61-93. [Abstract] [PDF] [PDF Plus]

439. C. H. Liang, C. Y. Chung, K. P. Wong, X. Z. Duan. 2007. Parallel Optimal Reactive Power Flow Based on Cooperative Co-Evolutionary Differential Evolution and Power System Decomposition. *IEEE Transactions on Power Systems* **22**:1, 249-257. [Crossref]

440. Chao Chen, YuanXin Tian, XiaoYong Zou, PeiXiang Cai, JinYuan Mo. 2007. A cooperative fast annealing coevolutionary algorithm for protein motif extraction. *Chinese Science Bulletin* **52**:3, 318-323. [Crossref]

441. Jing Jie, Jianchao Zeng, Chongzhao Han. 2007. An extended mind evolutionary computation model for optimizations. *Applied Mathematics and Computation* **185**:2, 1038-1049. [Crossref]

442. Nachol Chaiyaratana, Theera Piroonratana, Nuntapon Sangkawelert. 2007. Effects of diversity control in single-objective and multi-objective genetic algorithms. *Journal of Heuristics* **13**:1, 1-34. [Crossref]

443. Kamran Shafi, Hussein A. Abbass. 2007. Biologically-inspired Complex Adaptive Systems approaches to Network Intrusion Detection. *Information Security Technical Report* **12**:4, 209-217. [Crossref]

444. Nicolás García-Pedrajas, Domingo Ortiz-Boyer. 2007. A cooperative constructive method for neural networks for pattern recognition. *Pattern Recognition* **40**:1, 80-98. [Crossref]

445. C. S. Ong, H. Y. Quek, K. C. Tan, A. Tay. Discovering Chinese Chess Strategies through Coevolutionary Approaches 360-367. [Crossref]

446. Bo Liu, Hannan Ma, Xuejun Zhang. A Co-evolutionary Differential Evolution Algorithm for Constrained Optimization 51-57. [Crossref]

447. Zhichun Wang, Minqiang Li. A Coevolution Approach for Learning Multimodal Concepts 389-393. [Crossref]

448. Myung Won Kim, Joung Woo Ryu. An Efficient Coevolutionary Algorithm Using Dynamic Species Control 431-435. [Crossref]

449. C. K. Goh, K. C. Tan, E. B. Tay. A Competitive-Cooperation Coevolutionary Paradigm for Multi-objective Optimization 255-260. [Crossref]

450. Yan Jin, Wei Li. 2007. Design Concept Generation: A Hierarchical Coevolutionary Approach. *Journal of Mechanical Design* **129**:10, 1012. [Crossref]

451. M. Dolores Pérez-Godoy, Antonio J. Rivera, M. José del Jesus, Ignacio Rojas. CoEvRBFN: An Approach to Solving the Classification Problem with a Hybrid Cooperative-Coevolutive Algorithm 324-332. [Crossref]

452. G. Castellano, C. Castiello, A. M. Fanelli, L. Jain. Evolutionary Neuro-Fuzzy Systems and Applications 11-45. [Crossref]

453. Minh Ha Nguyen, Hussein A. Abbass, Robert I. Mckay. 2006. A novel mixture of experts model based on cooperative coevolution. *Neurocomputing* **70**:1-3, 155-163. [Crossref]

454. Zsolt Kira, Alan Schultz. Continuous and Embedded Learning for Multi-Agent Systems 3184-3190. [Crossref]

455. Fernanda Minku, Teresa Ludermir. EFuNN Ensembles Construction Using CONE with Multi-objective GA 9-9. [Crossref]

456. K.C. Tan, Y.J. Yang, C.K. Goh. 2006. A distributed Cooperative coevolutionary algorithm for multiobjective optimization. *IEEE Transactions on Evolutionary Computation* **10**:5, 527-549. [Crossref]

457. Steven Gustafson, Edmund K. Burke. 2006. The Speciating Island Model: An alternative parallel evolutionary algorithm. *Journal of Parallel and Distributed Computing* **66**:8, 1025-1036. [Crossref]

458. H. Chen, K.P. Wong, C.Y. Chung, D.H.M. Nguyen. 2006. A Coevolutionary Approach to Analyzing Supply Function Equilibrium Model. *IEEE Transactions on Power Systems* **21**:3, 1019-1028. [Crossref]

459. Lei Qi, Wu Min. Fuzzy Optimization Control of the Temperature for the Heating Process in Coke Oven Based on Co-evolution 420-424. [Crossref]

460. Jun Zhang, H.S.H. Chung, W.L. Lo. 2006. Pseudocoevolutionary genetic algorithms for power electronic circuits optimization. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)* **36**:4, 590-598. [Crossref]

461. Sin Man Cheang, Kwong Sak Leung, Kin Hong Lee. 2006. Genetic Parallel Programming: Design and Implementation. *Evolutionary Computation* **14**:2, 129-156. [Abstract] [PDF] [PDF Plus]

462. Michail Maniadakis, Panos Trahanias. 2006. Modelling brain emergent behaviours through coevolution of neural agents. *Neural Networks* **19**:5, 705-720. [Crossref]

463. V.R. Khare, X. Yao, B. Sendhoff. 2006. Multi-network evolutionary systems and automatic decomposition of complex problems. *International Journal of General Systems* **35**:3, 259-274. [Crossref]

464. Kittipong Boonlong, Kuntinee Maneeratana, Nachol Chaiyaratana. Determination of Erroneous Velocity Vectors by Co-operative Co-evolutionary Genetic Algorithms 1-6. [Crossref]

465. Zhou Hong, Wang Jian. A Cooperative Coevolutionary Algorithm with Application to Job Shop Scheduling Problem 746-751. [Crossref]

466. Ang Yang, H.A. Abbass, R. Sarker. 2006. Characterizing warfare in red teaming. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)* **36**:2, 268-285. [Crossref]

467. H. Chen, K.P. Wong, D.H.M. Nguyen, C.Y. Chung. 2006. Analyzing Oligopolistic Electricity Market Using Coevolutionary Computation. *IEEE Transactions on Power Systems* **21**:1, 143-152. [Crossref]

468. PETER C. MATTHEWS, DAVID W.F. STANDINGFORD, CARREN M.E. HOLDEN, KEN M. WALLACE. 2006. Learning inexpensive parametric design models using an augmented genetic programming technique. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **20**:1, 1-18. [Crossref]

469. M.H. Maruo, M.R. Delgado. Co-evolutionary Genetic Fuzzy System: A Self-adapting Approach 1417-1424. [Crossref]

470. Alexandre Blansche, Annett Wania, Pierre Gancarski. Comparison of MACLAW with several attribute selection methods for classification in hyperspectral images 231-236. [Crossref]

471. Yu Jiang, Jun Cao, Guo-hui Yang. Learning Technique for TSK Fuzzy Model Based on Cooperative Coevolution 1924-1929. [Crossref]

472. Jae-Yoon Jung, James A. Reggia. 2006. Evolutionary Design of Neural Network Architectures Using a Descriptive Encoding Language. *IEEE Transactions on Evolutionary Computation* **10**:6, 676-688. [Crossref]

473. Liviu Panait, Sean Luke, R. Paul Wiegand. 2006. Biasing Coevolutionary Search for Optimal Multiagent Behaviors. *IEEE Transactions on Evolutionary Computation* **10**:6, 629-645. [Crossref]

474. Rafał Dreżewski, Marek Kisiel-Dorohinicki. Maintaining Diversity in Agent-Based Evolutionary Computation 908-911. [Crossref]

475. Michelle Galea, Qiang Shen. Simultaneous Ant Colony Optimization Algorithms for Learning Linguistic Fuzzy Rules 75-99. [Crossref]

476. Christian Gagné, Marc Schoenauer, Michèle Sebag, Marco Tomassini. Genetic Programming for Kernel-Based Learning with Co-evolving Subsets Selection 1008-1017. [Crossref]

477. Haoyong Chen, Xifan Wang, Kit Po Wong, Chi-yung Chung. A Framework of Oligopolistic Market Simulation with Coevolutionary Computation 860-869. [Crossref]

478. Jianxue Wang, Weichao Wang, Xifan Wang, Haoyong Chen, Xiuli Wang. Cooperative Co-evolutionary Approach Applied in Reactive Power Optimization of Power System 620-628. [Crossref]

479. Wasan Srikasam, Nachol Chaiyaratana, Suwat Kuntanapreeda. Nonlinear Discrete System Stabilisation by an Evolutionary Neural Network 1090-1099. [Crossref]

480. Michail Maniadakis, Panos Trahanias. Modelling Robotic Cognitive Mechanisms by Hierarchical Cooperative CoEvolution 224-234. [Crossref]

481. Wasan Srikasam, Nachol Chaiyaratana, Suwat Kuntanapreeda. Nonlinear System Stabilisation by an Evolutionary Neural Network 998-1006. [Crossref]

482. Anders Lyhne Christensen, Marco Dorigo. Incremental Evolution of Robot Controllers for a Highly Integrated Task 473-484. [Crossref]

483. Michail Maniadakis, Panos Trahanias. Hierarchical Cooperative CoEvolution Facilitates the Redesign of Agent-Based Systems 582-593. [Crossref]

484. Sunil Pranit Lal, Koji Yamada, Satoshi Endo. Studies on Motion Control of a Modular Robot Using Cellular Automata 689-698. [Crossref]

485. Liviu Panait, Sean Luke. 2005. Cooperative Multi-Agent Learning: The State of the Art. *Autonomous Agents and Multi-Agent Systems* **11**:3, 387-434. [Crossref]

486. Rafal Kicinger, Tomasz Arciszewski, Kenneth De Jong. 2005. Evolutionary computation and structural design: A survey of the state-of-the-art. *Computers & Structures* **83**:23-24, 1943-1978. [Crossref]

487. J. Teo, H.A. Abbass. 2005. Multiobjectivity and Complexity in Embodied Cognition. *IEEE Transactions on Evolutionary Computation* **9**:4, 337-360. [Crossref]

488. J.C. Bongard, H. Lipson. 2005. Nonlinear System Identification Using Coevolution of Models and Tests. *IEEE Transactions on Evolutionary Computation* **9**:4, 361-384. [Crossref]

489. Jianjun Hu, Erik Goodman, Kisung Seo, Zhun Fan, Rondal Rosenberg. 2005. The Hierarchical Fair Competition (HFC) Framework for Sustainable Evolutionary Algorithms. *Evolutionary Computation* **13**:2, 241-277. [Abstract] [PDF] [PDF Plus]

490. N. Garcia-Pedrajas, C. Hervas-Martinez, D. Ortiz-Boyer. 2005. Cooperative Coevolution of Artificial Neural Network Ensembles for Pattern Classification. *IEEE Transactions on Evolutionary Computation* **9**:3, 271-302. [Crossref]
491. K. Krawiec, B. Bhanu. 2005. Visual Learning by Coevolutionary Feature Synthesis. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)* **35**:3, 409-425. [Crossref]
492. Dong-Wook Lee, Kwee-Bo Sim. 2005. Co-Evolutionary Model for Solving the GA-Hard Problems. *Journal of Fuzzy Logic and Intelligent Systems* **15**:3, 375-381. [Crossref]
493. Shimon Whiteson, Nate Kohl, Risto Miikkulainen, Peter Stone. 2005. Evolving Soccer Keepaway Players Through Task Decomposition. *Machine Learning* **59**:1-2, 5-30. [Crossref]
494. S. Kimura, K. Ide, A. Kashihara, M. Kano, M. Hatakeyama, R. Masui, N. Nakagawa, S. Yokoyama, S. Kuramitsu, A. Konagaya. 2005. Inference of S-system models of genetic networks using a cooperative coevolutionary algorithm. *Bioinformatics* **21**:7, 1154-1163. [Crossref]
495. Chris Lucas. 2005. Evolving an Integral Ecology of Mind. *Cortex* **41**:5, 709-725. [Crossref]
496. Kuntinee Maneeratana, Kittipong Boonlong, Nachol Chaiyaratana. 2005. Co-Operative Co-Evolutionary Genetic Algorithms for Multi-Objective Topology Design. *Computer-Aided Design and Applications* **2**:1-4, 487-496. [Crossref]
497. Michail Maniadakis, Panos Trahanias. CoEvolutionary Incremental Modelling of Robotic Cognitive Mechanisms 200-209. [Crossref]
498. Myung Won Kim, Joung Woo Ryu, Eun Ju Kim. A Coevolutionary Algorithm with Spieces as Varying Contexts 179-185. [Crossref]
499. Antonio J. Rivera, Ignacio Rojas, Julio Ortega. Application of ANOVA to a Cooperative-Coevolutionary Optimization of RBFNs 297-305. [Crossref]
500. Thomas Jansen, R. Paul Wiegand. 2004. The Cooperative Coevolutionary (1+1) EA. *Evolutionary Computation* **12**:4, 405-434. [Abstract] [PDF] [PDF Plus]
501. KITTIPONG BOONLONG, NACHOL CHAIYARATANA, SUWAT KUNTANAPREEDA. 2004. OPTIMAL CONTROL OF A HYSTERESIS SYSTEM BY MEANS OF CO-OPERATIVE CO-EVOLUTION. *International Journal of Computational Intelligence and Applications* **04**:04, 321-336. [Crossref]
502. Anawat Pongpunwattana, Rolf Rysdyk. 2004. Real-Time Planning for Multiple Autonomous Vehicles in Dynamic Uncertain Environments. *Journal of Aerospace Computing, Information, and Communication* **1**:12, 580-604. [Crossref]
503. A. Zaritsky, M. Sipper. 2004. The Preservation of Favored Building Blocks in the Struggle for Fitness: The Puzzle Algorithm. *IEEE Transactions on Evolutionary Computation* **8**:5, 443-455. [Crossref]
504. Luo Ronghua, Hong Bingrong. 2004. Coevolution Based Adaptive Monte Carlo Localization (CEAMCL). *International Journal of Advanced Robotic Systems* **1**:3, 19. [Crossref]
505. Assaf Zaritsky, Moshe Sipper. 2004. Coevolving solutions to the shortest common superstring problem. *Biosystems* **76**:1-3, 209-216. [Crossref]
506. CHAIWAT PIMPAWAT, NACHOL CHAIYARATANA. 2004. THREE-DIMENSIONAL CONTAINER LOADING USING A COOPERATIVE CO-EVOLUTIONARYGENETIC ALGORITHM. *Applied Artificial Intelligence* **18**:7, 581-601. [Crossref]
507. Edwin D. de Jong, Jordan B. Pollack. 2004. Ideal Evaluation from Coevolution. *Evolutionary Computation* **12**:2, 159-192. [Abstract] [PDF] [PDF Plus]
508. John Cartlidge, Seth Bullock. 2004. Combating Coevolutionary Disengagement by Reducing Parasite Virulence. *Evolutionary Computation* **12**:2, 193-222. [Abstract] [PDF] [PDF Plus]
509. CARLOS ANDRÉS PEÑA-REYES. 2004. Evolutionary Fuzzy Modeling Human Diagnostic Decisions. *Annals of the New York Academy of Sciences* **1020**:1, 190-211. [Crossref]
510. O. Cordón, F. Gomide, F. Herrera, F. Hoffmann, L. Magdalena. 2004. Ten years of genetic fuzzy systems: current framework and new trends. *Fuzzy Sets and Systems* **141**:1, 5-31. [Crossref]
511. Myriam Regattieri Delgado, Fernando Von Zuben, Fernando Gomide. 2004. Coevolutionary genetic fuzzy systems: a hierarchical collaborative approach. *Fuzzy Sets and Systems* **141**:1, 89-106. [Crossref]
512. N. García-Pedrajas, D. Ortiz-Boyer, C. Hervás-Martínez. 2004. Cooperative coevolution of generalized multi-layer perceptrons. *Neurocomputing* **56**, 257-283. [Crossref]
513. P. A. Castillo, M. G. Arenas, J. J. Merelo, G. Romero, F. Rateb, A. Prieto. Comparing Hybrid Systems to Design and Optimize Artificial Neural Networks 240-249. [Crossref]

514. David Cornforth, Michael Kirley. Cooperative Problem Solving Using an Agent-Based Market 60-71. [Crossref]
515. Rafał Dreżewski. A Co-evolutionary Multi-agent System for Multi-modal Function Optimization 654-661. [Crossref]
516. Edwin D. de Jong. The Incremental Pareto-Coevolution Archive 525-536. [Crossref]
517. Antony W. Iorio, Xiaodong Li. A Cooperative Coevolutionary Multiobjective Algorithm Using Non-dominated Sorting 537-548. [Crossref]
518. Liviu Panait, R. Paul Wiegand, Sean Luke. A Sensitivity Analysis of a Cooperative Coevolutionary Algorithm Biased for Optimization 573-584. [Crossref]
519. Sohail Nadimi, Bir Bhanu. Cooperative Coevolution Fusion for Moving Object Detection 587-589. [Crossref]
520. Myung Won Kim, Joung Woo Ryu. Species Merging and Splitting for Efficient Search in Coevolutionary Algorithm 332-341. [Crossref]
521. Pieter J. 't Hoen, Edwin D. de Jong. Evolutionary Multi-agent Systems 872-881. [Crossref]
522. Kuntinee Maneeratana, Kittipong Boonlong, Nachol Chaiyaratana. Multi-objective Optimisation by Co-operative Co-evolution 772-781. [Crossref]
523. Vineet R. Khare, Xin Yao, Bernhard Sendhoff. Credit Assignment Among Neurons in Co-evolving Populations 882-891. [Crossref]
524. Liviu Panait, R. Paul Wiegand, Sean Luke. A Visual Demonstration of Convergence Properties of Cooperative Coevolution 892-901. [Crossref]
525. J. Timmis, T. Knight, L. N. de Castro, E. Hart. An Overview of Artificial Immune Systems 51-91. [Crossref]
526. Krzysztof Krawiec, Leszek Włodarski. Coevolutionary feature construction for transformation of representation of machine learners 139-150. [Crossref]
527. Krzysztof Krawiec, Bir Bhanu. Coevolutionary Computation for Synthesis of Recognition Systems 59-59. [Crossref]
528. N. Garcia-Pedrajas, C. Hervas-Martinez, J. Munoz-Perez. 2003. COVNET: a cooperative coevolutionary model for evolving artificial neural networks. *IEEE Transactions on Neural Networks* **14**:3, 575-596. [Crossref]
529. Ben Hua, Jinbiao Yuan, David C.W Hui. The coevolutionary supply chain 487-492. [Crossref]
530. Edwin D. de Jong. Representation Development from Pareto-Coevolution 262-273. [Crossref]
531. P.A. Castillo, M.G. Arenas, J.J. Merelo, G. Romero. Cooperative Co-evolution of Multilayer Perceptrons 358-365. [Crossref]
532. Shimon Whiteson, Nate Kohl, Risto Miikkulainen, Peter Stone. Evolving Keepaway Soccer Players through Task Decomposition 356-368. [Crossref]
533. Thomas Jansen, R. Paul Wiegand. Exploring the Explorative Advantage of the Cooperative Coevolutionary (1+1) EA 310-321. [Crossref]
534. Krzysztof Krawiec, Bir Bhanu. Coevolution and Linear Genetic Programming for Visual Learning 332-343. [Crossref]
535. Terence Soule. Cooperative Evolution on the Intertwined Spirals Problem 434-442. [Crossref]
536. Adina Florea, Cosmin Carabelea. Genetic Models for the Rational Exploitation of Resources 351-360. [Crossref]
537. Jamie Twycross, Steve Cayzer. An Immune-based Approach to Document Classification 33-46. [Crossref]
538. Jorge Casillas, Oscar Cordón, Francisco Herrera, Luis Magdalena. Accuracy Improvements to Find the Balance Interpretability-Accuracy in Linguistic Fuzzy Modeling: An Overview 3-24. [Crossref]
539. Carlos-Andrés Peña-Reyes, Moshe Sipper. Fuzzy CoCo: Balancing Accuracy and Interpretability of Fuzzy Models by Means of Coevolution 119-146. [Crossref]
540. N García-Pedrajas, C Hervás-Martínez, J Muñoz-Pérez. 2002. Multi-objective cooperative coevolution of artificial neural networks (multi-objective cooperative networks). *Neural Networks* **15**:10, 1259-1278. [Crossref]
541. J. A. Sánchez-Mesa, C. Galan, J. A. Martínez-Heras, C. Hervás-Martínez. 2002. The use of a neural network to forecast daily grass pollen concentration in a Mediterranean region: the southern part of the Iberian Peninsula. *Clinical & Experimental Allergy* **32**:11, 1606-1612. [Crossref]
542. C. Anglano, M. Botta. 2002. NOW G-Net: learning classification programs on networks of workstations. *IEEE Transactions on Evolutionary Computation* **6**:5, 463-480. [Crossref]
543. Francisco Herrera, Oscar Cordon, Rafael Alcalá, Jorge Casillas. Techniques for Designing and RefiningLinguistic Fuzzy Models to Improve Their Accuracy . [Crossref]
544. Chris Brinton, Jimmy Krozel, Brian Capozzi, Stephen Atkins. Improved Taxi Prediction Algorithms for the Surface Management System . [Crossref]

545. Jorge Casillas, O. Cordón, F. Herrera, J.J. Merelo. Cooperative Coevolution for Learning Fuzzy Rule-Based Systems 311-322. [Crossref]
546. Antony Iorio, Xiaodong Li. Parameter Control within a Co-operative Co-evolutionary Genetic Algorithm 247-256. [Crossref]
547. Nattavut Keerativuttitumrong, Nachol Chaiyaratana, Vara Varavithya. Multi-objective Co-operative Co-evolutionary Genetic Algorithm 288-297. [Crossref]
548. André L.V. Coelho, Daniel Weingaertner, Ricardo R. Gudwin, Ivan L.M. Ricarte. 2001. Emergence of multiagent spatial coordination strategies through artificial coevolution. *Computers & Graphics* **25**:6, 1013-1023. [Crossref]
549. C.A. Pena-Reyes, M. Sipper. 2001. Fuzzy CoCo: a cooperative-coevolutionary approach to fuzzy modeling. *IEEE Transactions on Fuzzy Systems* **9**:5, 727-737. [Crossref]
550. J.X. Chen, S. Wang. 2001. Data visualization: parallel coordinates and dimension reduction. *Computing in Science & Engineering* **3**:5, 110-113. [Crossref]
551. Myriam Regattieri Delgado, Fernando Von Zuben, Fernando Gomide. 2001. Hierarchical genetic fuzzy systems. *Information Sciences* **136**:1-4, 29-52. [Crossref]
552. Emma Hart, Peter Ross. Clustering Moving Data with a Modified Immune Algorithm 394-403. [Crossref]
553. N. García-Pedrajas, E. Sanz-Tapia, D. Ortiz-Boyer, C. Hervás-Martínez. Introducing Multi-objective Optimization in Cooperative Coevolution of Neural Networks 645-652. [Crossref]
554. Carlos Andrés Peña-Reyes, Moshe Sipper. The Flowering of Fuzzy CoCo: Evolving Fuzzy Iris Classifiers 304-307. [Crossref]
555. Zsolt Kira, Mitchell A. Potter. Exerting human control over decentralized robot swarms 566-571. [Crossref]
556. Yichuan Jiang, Yiping Zhong, Shiyong Zhang. The evolution of agents cooperation communication architecture based on graph theory 280-286. [Crossref]
557. A.M. Florea. Genetic estimation of competitive agents behavior 418-423. [Crossref]
558. C. Pimpawat, N. Chaiyaratana. Using a co-operative co-evolutionary genetic algorithm to solve a three-dimensional container loading problem 1197-1204. [Crossref]
559. M.R. Delgado, F. Von Zuben, F. Gomide. Coevolutionary design of Takagi-Sugeno fuzzy systems 1384-1389. [Crossref]
560. Kittipong Boonlong, Nachol Chaiyaratana, Suwat Kuntanapreeda. Using a co-operative co-evolutionary genetic algorithm to solve optimal control problems in a hysteresis system 1504-1509. [Crossref]
561. R.P. Wiegand, W.C. Liles, K.A. De Jong. Analyzing cooperative coevolution with evolutionary game theory 1600-1605. [Crossref]
562. Ming Chang, K. Ohkura, K. Ueda, M. Sugiyama. A symbiotic evolutionary algorithm for dynamic facility layout problem 1745-1750. [Crossref]
563. D. Sofge, K. De Jong, A. Schultz. A blended population approach to cooperative coevolution for decomposition of complex problems 413-418. [Crossref]
564. K.C. Tan, V.J. Yang, T.H. Lee. A distributed cooperative coevolutionary algorithm for multiobjective optimization 2513-2520. [Crossref]
565. Jun Zhang, H.S.H. Chung, W.L. Lo, E.P.W. Tam, J.J. Lee, A.K.M. Wu. Pseudo-coevolutionary genetic algorithms for power electronic circuits optimization 474-481. [Crossref]
566. E.D. de Jong. Towards a bounded Pareto-coevolution archive 2341-2348. [Crossref]
567. V.R. Khare, Xin Yao, B. Sendhoff, Yaochu Jin, H. Wersing. Co-evolutionary Modular Neural Networks for Automatic Problem Decomposition 2691-2698. [Crossref]
568. M. Maniadakis, P. Trahanias. Distributed Brain Modelling by means of Hierarchical Collaborative CoEvolution 2699-2706. [Crossref]
569. F.L. Minku, T.B. Ludermir. EFuNNs Ensembles Construction Using a Clustering Method and a Coevolutionary Genetic Algorithm 1399-1406. [Crossref]
570. V. Bevilacqua, G. Mastronardi, F. Menolascina. Hybrid Data Ananlysis Methods and Artificial Neural Network Design in Breast Cancer Diagnosis: IDEST Experience 373-378. [Crossref]
571. H. Huning. Convergence analysis of a segmentation algorithm for the evolutionary training of neural networks 70-81. [Crossref]
572. C. Coello, A. Aguirre, B. Buckles. Evolutionary multiobjective design of combinational logic circuits 161-170. [Crossref]
573. C.A. Pena-Reyes, M. Sipper, L. Prieto. Sensitive, specific, and interpretable: evolving a fuzzy mammographic-interpretation assessment tool 837-842. [Crossref]
574. Ping Yan, Ming-Yue Ding, Cheng-Ping Zhou. Game-theoretic route planning for team of UAVs 723-728. [Crossref]

575. S. Gustafson, E.K. Burke. A Niche for Parallel Island Models: Outliers and Local Search 612-619. [Crossref]
576. K.C. Tan, Y.H. Chew, T.H. Lee, Y.J. Yang. A cooperative coevolutionary algorithm for multiobjective optimization 390-395. [Crossref]
577. Y.-C. Tan, T.H. Lee, Y.J. Yang, D.S. Liu. A cooperative coevolutionary algorithm for multiobjective optimization 1926-1931. [Crossref]
578. M. Noura, M. Batouche. Ima e Enhancement sin redator- rey ptimi er Al orithm 1584-1589. [Crossref]
579. M. Maniadakis, P. Trahanias. A hierarchical coevolutionary method to support brain-lesion modelling 434-439. [Crossref]
580. J. Zhang, A.K.M. Wu, H.S.H. Chung. On the use of pseudo-coevolutionary genetic algorithms with adaptive migration for design of power electronics regulators 297-300. [Crossref]
581. A.L.V. Coelho. On the cooperation of fuzzy neural networks via a coevolutionary approach 785-790. [Crossref]
582. Maohai Li, Bingrong Hong, Ronghua Luo. Coevolution Particle Filter for Mobile Robot Simultaneous Localization and Mapping 808-813. [Crossref]
583. H. Chen, K.P. Wong, X. Wang, C.Y. Chung. A coevolutionary approach to modeling oligopolistic electricity markets! 248-254. [Crossref]
584. E. Parsopoulos Konstantinos, N. Vrahatis Michael. Established and Recently Proposed Variants of Particle Swarm Optimization 88-132. [Crossref]
585. Lei Fang, Sheng-Uei Guan, Haofan Zhang. Recursive Learning of Genetic Algorithms with Task Decomposition and Varied Rule Set 226-247. [Crossref]
586. Nicolás García-Pedrajas. Cooperative Coevolution of Neural Networks and Ensembles of Neural Networks 465-490. [Crossref]
587. Ana Claudia M. L. Albuquerque, Jorge D. Melo, Adrião D. Dória Neto. Evolutionary Computation and Parallel Processing Applied to the Design of Multilayer Perceptrons 181-203. [Crossref]
588. Nicolás García-Pedrajas, César Hervás-Martínez, Domingo Ortiz-Boyer. Cooperative Coevolutionary Methods 181-206. [Crossref]
589. M. G. Arenas, P. A. Castillo, G. Romero, F. Rateb, J. J. Merelo. Co-evolving Multilayer Perceptrons Along Training Sets 503-513. [Crossref]
590. Y.N. Lu, Y.L. Liu, Y.C. Liang. AN IMPROVED COEVOLUTION ALGORITHM FOR FUZZY MODELLING 1369-1373. [Crossref]
591. Jin Tian, Minqiang Li, Fuzan Chen. A Cooperative Coevolution Algorithm of RBFNN for Classification 809-816. [Crossref]
592. Daniel Kanevskiy, Konstantin Vorontsov. Cooperative Coevolutionary Ensemble Learning 469-478. [Crossref]
593. Michael D. Schmidt, Hod Lipson. Coevolving Fitness Models for Accelerating Evolution and Reducing Evaluations 113-130. [Crossref]
594. Rafał Dreżewski, Łukasz Dronka, Jarosław Koźlak. Co-operative Co-evolutionary System for Solving Dynamic VRPTW Problems with Crisis Situations 81-92. [Crossref]
595. Jin Wang, Chang Hao Piao, Chong Ho Lee. Implementing Multi-VRC Cores to Evolve Combinational Logic Circuits in Parallel 23-34. [Crossref]
596. Geoff S. Nitschke. Neuro-evolution Methods for Designing Emergent Specialization 1120-1130. [Crossref]
597. Hywel T. P. Williams, Timothy M. Lenton. Artificial Ecosystem Selection for Evolutionary Optimisation 93-102. [Crossref]
598. Rafał Dreżewski. A Model of Co-evolution in Multi-agent System 314-323. [Crossref]
599. Krzysztof Krawiec, Bir Bhanu. Coevolutionary Feature Learning for Object Recognition 224-238. [Crossref]
600. Edwin D. de Jong, Anthony Bucci. Objective Set Compression 357-376. [Crossref]
601. Lanshun Nie, Xiaofei Xu, Dechen Zhan. Coevolutionary Computation Based Iterative Multi-Attribute Auctions 461-469. [Crossref]
602. Gabriela Ochoa, Evelyne Lutton, Edmund Burke. The Cooperative Royal Road: Avoiding Hitchhiking 184-195. [Crossref]
603. Kay Chen Tan, Chi Keong Goh. Handling Uncertainties in Evolutionary Multi-Objective Optimization 262-292. [Crossref]
604. Rafał Dreżewski, Leszek Siwik. Agent-Based Co-Operative Co-Evolutionary Algorithm for Multi-Objective Optimization 388-397. [Crossref]
605. Min Shi, Haifeng Wu. Evolving Efficient Connection for the Design of Artificial Neural Networks 909-918. [Crossref]

606. Fangming Zhu, Sheng-Uei Guan. Enhanced Cooperative Co-evolution Genetic Algorithm for Rule-Based Pattern Classification 113-123. [Crossref]
607. Shuhei Kimura. Inference of Genetic Networks Using an Evolutionary Algorithm 31-51. [Crossref]
608. Rafał Dreżewski, Jan Sepielak, Leszek Siwik. Classical and Agent-Based Evolutionary Algorithms for Investment Strategies Generation 181-205. [Crossref]