Controlling the Cooperative-Competitive Boundary in Niched Genetic Algorithms

Jeffrey Horn*
Department of Mathematics and Computer Science
Northern Michigan University

Abstract

Niching can allow a diverse population to cooperatively represent a single, distributed solution to the problem at hand. Successful niching mechanisms must promote both **cooperation** (i.e., co-existence of separate "species" for each desired niche), and **competition** (i.e., intensive search for the best species for each niche). In this paper we seek to control the competitive-cooperative boundary in the space of possible niche relationships, so that we can choose which pairs of interacting niches will survive under GA selection and which niche pairs will be resolved to yield a single winner. We introduce the concept of *resource replenishment period,* τ, as a control on the relative importance of objective fitness over diversity pressure. We find that by varying τ between zero and one, we can smoothly transition between pure selection and full niching.

1 INTRODUCTION

The term *niching* has often been applied to the formation of various species to exploit different resources (a.k.a., "niches") in the environment. In the context of evolutionary computation, *niching* usually refers to algorithms that promote survival of subpopulations of individuals corresponding to different, high fitness regions of the search space. In particular, we use the term to refer to one particular class of algorithms inspired by nature: the *sharing* of finite resources among competing individuals.

* (address) 1401 Presque Isle Ave., Marquette, MI 49855 USA; (e-mail) jhorn@nmu.edu; (phone) 906/227-1607; (fax) 906/227-2010; (www) http: // euclid.nmu.edu/ ~jeffhorn .

In *resource sharing* each unit of a resource is split among all individuals in the population that are competing for that resource. For example, in an immune system algorithm, all of the evolving antibodies that bind to a particular antigen *share* the credit/reward for correctly identifying that object as foreign.

In previous work (e.g., Horn, 1997; Horn & Goldberg, 1999), we developed predictive models of niching success and failure under resource sharing. We learned how to predict which niching scenarios will result in *competition* between species (and therefore elimination of the less fit species), and which will result in *cooperation* (i.e., long-term survival of both species, in proportion to their relative fitnesses). These models have been verified experimentally, and include the complex situation of overlapping niches. As the overlap between two species' niches increases from zero to complete overlap, somewhere a phase transition is crossed from cooperation to competition. We have analyzed that boundary previously.

What we have not yet done is learned to *control* that boundary. That is, we have not yet been able to reliably change the boundary between cooperation and competition in a natural way. In this paper we introduce a very simple and natural mechanism for controlling the effects of sharing/niching on evolution. By incorporating a single new parameter τ to model the rate at which finite resources are replenished, we actually adjust the cooperative competitive border, moving from pure selection ($\tau = 0$) to "normal" full niching ($\tau = 1$) and even beyond ($1 < \tau$) to rather bizarre evolution trajectories.

1.1 THE NEED FOR NICHING

In a genetic algorithm (GA), selection drives the evolving population toward a uniform distribution of N copies of the most highly fit individual (Goldberg, 1989). In many applications of the GA uniform convergence is undesirable. In the learning classifier system (LCS), we ask the GA to search through the space of all possible rules to find and maintain a diverse, cooperative subpopulation of rules that together represent a *concept*.

To prevent the best individual in the population from replacing all copies of competing rivals, some kind of *niching* (a.k.a. *speciation*) is necessary. Niching induces *restorative pressure* (Horn, 1993), to balance the *convergence pressure* of selection.

1.2 NICHING VIA RESOURCE SHARING

A natural niching effect is implicitly induced by competition for limited resources (i.e., finite rewards). Resource sharing is applied to tasks in which multiple, finite, explicit resources are known and can be "attributed" or "credited" differentially to various individuals in the population[1].

The basic algorithm common to all resource sharing systems is simple and intuitive:

1. For each of the finite resources r_i, divide it up among all qualified individuals contending for it, in proportion to their various merits (that is, the relative strengths of their claims). Thus two equally deserving individuals should be allocated equal amounts of the resource. If the resource is discrete, and cannot be evenly divided, then for each indivisible unit of the resource, randomly choose among equally deserving individuals. This random choice results in an *expected* uniform distribution of resources among equally deserving candidates.

2. For each individual, add all rewards/credits earned in the first step, and use this amount (perhaps scaled) as the fitness for GA selection.

3. After a new generation is produced, replenish/renew the resources and start over at the first step above. Continue to loop until some stopping criterion is met.

The idea of splitting up a limited resource among all competing individuals seems evident in nature and is simple to implement. Resource sharing is often incorporated in adaptive, or simulated, systems, as we illustrate below.

[1] If no such explicit resources are identifiable in the problem domain, objective fitness can be considered a finite resource to be "shared", resulting in *fitness sharing* (Goldberg & Richardson, 1987; Deb, 1989; Horn, 1993; Mahfoud, 1995), which has been successfully applied to function optimization and multi-objective problem solving. Fitness and resource sharing are indeed so similar that a unified model of *sharing* for niching is possible (Horn, 1997).

1.3 RESOURCE SHARING APPLICATIONS

This simple and natural scheme has been abstracted into a number of population-based algorithms, including: learning classifier systems (LCS) (Booker, 1982; Wilson, 1987), immune system models (Smith, Forrest, & Perelson, 1993), multi-agent systems, and ecological simulations (Huberman, 1988). The resource sharing approach seems well-suited to several major open problem domains, such as classification, general covering problems, and layout problems. Sharing attacks two major difficulties of population-based solutions to hard problems: credit assignment, and problem decomposition. Below we illustrate how sharing deals with credit assignment and dynamic subtasking/decomposition by examining the learning classifier system under "example sharing".

1.3.1 The Classifier System as an Example of Resource Sharing

An example of resource sharing occurs in most implementations of the *Michigan-style* learning classifier system (Horn, Goldberg, & Deb, 1994). In an LCS, the population consists of classification rules, or simply *classifiers*. These rules attempt to classify examples (either from some training set or some test set). The rules are rewarded for correct classifications of examples, earning credit for each correct classification of an example. The sum of credits earned, over all examples, contributes to each rule's fitness. This fitness is then used in normal GA selection.

In the Michigan LCS, individuals (classification rules, or *classifiers*) compete for the rewards (or credit) given for proper classification of a finite number of examples. Several researchers have shown that simply dividing up an example's reward/credit among all rules that successfully classify that example (i.e., sharing), effectively and robustly maintains a diverse set of rules that together "cover" the examples (e.g., Booker, 1982; Wilson, 1987). Thus LCS sharing is an instance of resource sharing in which the resources are the rewards/credits for the examples. Henceforth, we will consider the examples themselves to be the resources to be shared, in order to simplify our discussion. Thus this strategy is often known as *example sharing* (McCallum & Spackman, 1990; Neri & Saitta, 1995).

The notions of competition and niche overlap are easy to visualize in the case of resource sharing. In Figure 1, the large rectangle represents the space of all *positive examples* given to the LCS for learning. The size of a circle represents the number of *examples* covered by

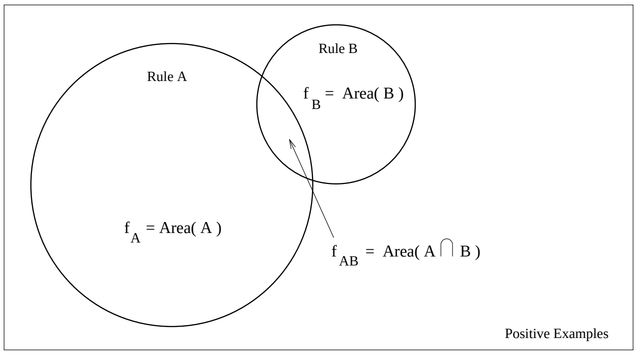

Figure 1: In the case of the learning classifier system (LCS), *implicit niching* is induced by rules competing to classify examples. We can use area in the space of examples to indicate a rule's coverage, which is also its *objective* (i.e., unshared) fitness.

the corresponding rule, and hence the number of correct classifications it makes. The overlaps of circles represent overlaps of coverage among rules, and thus contain the examples "shared" by two or more rules[2].

To illustrate the actual sharing of resources that leads to implicit niching: let f_A and f_B be the objective fitnesses for rules **A** and **B** respectively. The objective fitness could be taken as the number of examples covered[3] by that rule, in the case of binary classification. Let f_{AB} be the amount of resources in the overlapping coverage of rules **A** and **B**. That is, f_{AB} is the amount of resources shared by **A** and **B** (e.g., the number of examples covered by both). Let n_A, n_B be the number of copies of rules **A** and **B**, respectively, in our population of size N (thus $N = n_A + n_B$). Then we can calculate the shared fitness of rule **A**:

$$f_{sh,A} = \frac{f_A - f_{AB}}{n_A} + \frac{f_{AB}}{n_A + n_B}. \tag{1}$$

Similarly for rule **B**,

$$f_{sh,B} = \frac{f_B - f_{AB}}{n_B} + \frac{f_{AB}}{n_A + n_B}. \tag{2}$$

Thus resources are shared equally among all competitors. The set of examples covered only by **A**, which is $f_A - f_{AB}$, is divided evenly among the n_A copies of **A**, while the set of examples covered by both **A** and **B**, f_{AB}, is shared evenly among all $n_A + n_B$ individuals.

1.4 PREVIOUS WORK: PREDICTIVE MODELS OF RESOURCE SHARING

Horn (1997, and Horn & Goldberg, 1996, 1999) developed a model for the two-niche case (i.e., only two

[2] That is, the examples correctly classified by *both* rules.

[3] In other words, classified correctly.

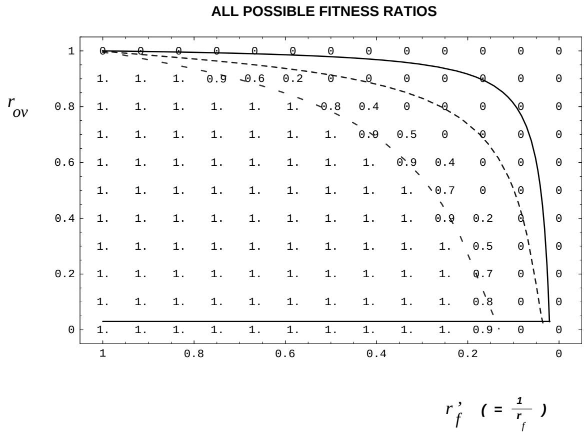

Figure 2: A predictive map for niching success for any two-niche situation (i.e., ratio of fitness r'_f and ratio of overlap r_{ov}) can predict niching success (ones) and failures (zeros) as well as the less clear "gray area" in between. The solid and dashed lines shown are theoretical bounds derived in previous studies.

species) to include all possible scenarios of fitness ratios and niche overlap. To briefly review, we define the ratio of fitness between species **A** and **B** to be $r'_f \equiv \frac{f_B}{f_A}$, and the ratio of niche overlap to be $r_{ov} \equiv \frac{f_{AB}}{f_B}$. Note that if we always assume, without loss of generality, that $f_B \leq f_A$ (**B** is always the less fit), then $0 \leq r'_f \leq 1$. Note also that $0 \leq r_{ov} \leq 1$.

Using Markov chain analyses of proportionate selection in genetic algorithms (GAs), calculating equilibrium conditions under selection, and modeling convergence to equilibrium under pure selection (i.e., no recombination or mutation), we developed predictive maps of cooperation and competition, as in Figure 2. In the figure, we can see a region of clear niching success, what we call cooperation, in which both species survive for many generations ($g = 200$ in this case) under pure selection (proportionate) with no crossover or mutation. The numbers are expected success probabilities (i.e., percentage of random runs in which both species are still present after 200 generations). These probabilities are exact, in that they were calculated via the Markov model, but are shown rounded to the nearest tenth.

Figure 2 also shows the competitive region, with near zero probability of both species surviving. Cooperation is most likely where fitness difference and overlap are low, as one would expect.

But Figure 2 is purely predictive. We cannot "move" the regions of cooperation and of competition, short of changing the population size (increasing population size does indeed increase the area of the cooperative region). What if, for example, we want selection to choose the better of two species when their ratios of fitness and overlap are 0.5 and 0.5, in a population of size 50? As Figure 2 shows, our current algorithm for resource sharing would keep both species as "cooperators", whether or not we want competition here.

I will argue here that what is needed is a way to ameliorate the effect of sharing. Currently our model (and indeed most implementations of sharing) allows us only to turn sharing fully on or fully off. Either individuals share resources equally among all competitors, or they each get a "copy" of the entire resource themselves (i.e., $f_{sh,A} = f_A$; pure selection). What could lie between these extremes?

2 CONTROLLING THE EFFECT OF SHARING THROUGH RESOURCE REPLENISHMENT

Consider a particular resource, say f_A above, to be shared among n_A individuals "covering" that resource. Normally, we assume that each individual will use as much of each resource as it can, and so we divide up the resource completely and evenly, thus $f_{A,sh} = \frac{f_A}{n_A}$. We are assuming that each resource, such as f_A, is not "replenished" until the beginning of the next round of fitness evaluations (i.e., the next generation). The amount of resource A, which is f_A, is fixed and finite for one generation.

At another extreme, we assume an infinite supply of f_A, or at least enough so that each individual competing for f_A gets its full share, thus $f_{A,sh} = f_A$, and we are back to a simple GA with no niching at all. Another way to look at this situation is to consider that each resource, such as A in f_A, is replenished immediately after each individual takes its full f_A amount. Thus the individuals do not conflict with each other.

If we now define τ to be a rough model of the *resource replenishment period*, then when $\tau = 1$ generation we get only f_A of resource A to be divided (equally) among the n_A copies of species **A**. But if $\tau = 0$, then resources are resupplied instantaneously, and all n_A copies get their full amount of A, namely f_A.

If we make τ an "exponent of resource niche count", as in

$$f_{sh,A} = \frac{f_A}{n_A^\tau},$$

then at one extreme, $\tau = 1$, and we have normal, full sharing:

$$f_{sh,A} = \frac{f_A}{n_A},$$

while at the other extreme of no sharing, $\tau = 0$, and we have the normal selective pressure of the simple GA:

$$f_{sh,A} = f_A.$$

Between these two extremes are situations in which the resource is neither completely divided up nor "replenished" infinitely often. Rather, the resource is limited, but not to a fixed value f_A. Thus instead of receiving just $1/n_A$ of the resource, each individual receives some share between $1/n_A$ and 1 of the resource. For example, what happens when we set τ to $\frac{1}{2}$?

$$f_{sh,A} = \frac{f_A}{\sqrt{n_A}}$$

For $0 < \tau \to 1$, we expect to see an increasing emphasis on niching and a corresponding decrease in emphasis on the purely selective fitness ratio $r_f = \frac{f_A}{f_B}$. However, we have only studied the cases $\tau = 0$ (pure selection, see Deb and Goldberg (1989)), and $\tau = 1$ (pure niching, see (Horn, 1997)). In the following, we examine the effects of the resource replenishment period τ on our models and hence on our evolutionary algorithms.

3 NICHING MODELS REVISED

We follow the sequence of model derivation presented in (Horn, 1997), plugging in τ and highlighting the essential results.

3.1 A New Equilibrium

We now recalculate niching equilibrium to include the new niching parameter, τ. The original equilibrium condition (Deb, 1989), sets all (shared) fitnesses equal:

$$f_{sh,A} = f_{sh,B}. \tag{3}$$

Plugging in the old shared fitness expressions,

$$\frac{f_A - f_{AB}}{n_A} + \frac{f_{AB}}{n_A + n_B} = \frac{f_B - f_{AB}}{n_B} + \frac{f_{AB}}{n_A + n_B},$$

and solving for the ratio $r_{eq,n}$ of **B**s to **A**s at equilibrium (that is, the *equilibrium ratio* $r_{eq,n} \equiv \frac{n_B}{n_A}$):

$$r_{eq,n} = \frac{r'_f - r_o}{1 - r_o}, \tag{4}$$

where r_o was defined in earlier work (Horn, Deb, & Goldberg, 1994) to be the ratio of overlap: $r_o \equiv \frac{f_{AB}}{f_A}$, which is related to the alternate overlap ratio $r_{ov} = \frac{f_{AB}}{f_B}$ as $r_o = \frac{r_{ov}}{r'_f}$.

Now to update the equilibrium condition to include τ, we simply plug in the new shared fitnesses $f_{sh,A} = \frac{f_A - f_{AB}}{n_A^\tau} + \frac{f_{AB}}{(n_A+n_B)^\tau}$, and similarly for $f_{sh,B}$, we can solve again for $r_{eq,n}$ to get

$$r_{eq,n} = \left(\frac{r'_f - r_o}{1 - r_o}\right)^{\frac{1}{\tau}}. \tag{5}$$

We check our algebra by noting that when $\tau = 1$ the above reduces to the old equilibrium expression above, while when $\tau = 0$ the equilibrium ratio blows up. This is to be expected, since we said earlier that $\tau = 0$ effectively turns off sharing, and under pure selection there is no such equilibrium to be described.

So our new $r_{eq,n}$ is simply our old $r_{eq,n}$ to the τ^{th} root. We can adjust the equilibrium ratio by changing τ.

3.2 A NEW MARKOV CHAIN

Markov chain analysis of GA selection has proven illustrative and practical, at least for the two-species case (for which the analysis is actually tractable). A Markov chain provides an *exact* model, giving the expected performance of evolution over all random initial starting situations (e.g., random seeds). All that is need to set up the Markov chain for proportionate selection is the probability of choosing a copy of **A** given the current state (which is the number of copies of **A** in the current generation). Under proportionate selection, the probability of selecting an **A** is simply:

$$p_A = \frac{f_{sh,A} * n_A}{f_{sh,A} * n_A + f_{sh,B} * n_B}.$$

Plugging in our shared fitnesses above, yields $p_A =$

$$\frac{1 - r_o + r_o(\frac{n_A}{N})^\tau}{1 - r_o + (\frac{n_A}{n_B})^{\tau-1}(r'_f - r_o) + r_o(\frac{n_A}{N})^\tau + r_o\frac{n_B}{n_A}(\frac{n_A}{N})^\tau}, \tag{6}$$

where n_A is the number of **A**s currently in the population, and $N = n_A + n_B$ is the fixed population size. Note that $p_B = 1 - p_A$, and that $n_B = N - n_A$. If $n_{A,t}$ is the number of **A**s at generation t, and $n_{A,t+1}$ is the number of copies in the next generation, and we define $P(n_{A,t}, n_{A,t+1})$ as being the probability of going from $n_{A,t}$ copies to $n_{A,t+1}$ copies of **A** in a single generation, then the transition matrix entries of the Markov chain are defined by

$$P(n_{A,t}, n_{A,t+1}) = \binom{N}{n_{A,t+1}}(p_A)^{n_{A,t+1}}(p_B)^{n_{B,t+1}}.$$

We can look at a particular situation by setting population size $N = 50$, $r_f = 2$ (**A** has twice the fitness of **B**), and $r_o = 0$ (no overlap for now). We can then compute all 51 by 51 probabilities in the transition matrix of the Markov chain. In Figure 3 we show three such transition matrices, plotted as contour plots, with the lighter shades representing regions of higher probability. The vertical axis is labeled with the state information for the current generation t. Thus 22 is the state with 22 out of 50 **A**s, and its row shows the 51 probabilities (totalling to one) for transitioning (in a single generation) to any of the 51 states (of zero to fifty copies of **A**) at time $t + 1$. Figure 3 shows three such matrices, in which only τ is varied.

In the upper plot $\tau = 1$ and we have the familiar case of "perfect sharing" (i.e., no niche overlap), in which the population tends to transition immediately to the equilibrium state of $2/3 * 50$ **A**s. A "restorative pressure" tends to maintain this equilibrium. This chain has been studied in several papers (e.g., Horn, 1993; Mahfoud, 1994). In the bottom plot, we have $\tau = 0$ and we are back to pure selection, in which the population is driven toward the absorbing state of $n_A = N$. As an example of an intermediate τ, we set $\tau = 1/2$ for the middle plot. Interestingly, this matrix has the S-shaped ridge of a fitness sharing Markov chain (Horn, 1993), rather than the linear ridge of a resource sharing Markov chain (Horn, Goldberg, & Deb, 1994).

4 EMPIRICAL RESULTS

So far we have justified this resource replenishment mechanism in two ways: (1) arguing its relevance as an abstraction of what really happens in nature, and (2) performing some preliminary analysis to verify its smooth convergence to "normal" sharing and selection at its extremes. Now it is time to analyze its practical effect. We perform some runs to see if indeed the adjustment of τ can significantly alter the boundary between cooperation and competition.

4.1 MOVING THE COOPERATIVE-COMPETITIVE BOUNDARY

We begin the experiments by simply regenerating the data in Figure 2 but for different values of τ. Again, these data come from running our initial population distribution vector through the transition matrix 200 times, and then adding up the transient state probabilities to see what is the expected proportion of runs that maintain both niches for 200 generations.

4.1.1 Small τ

In Figure 4 we show three such predictive maps, using three different values of τ. At the top of Figure 4 we

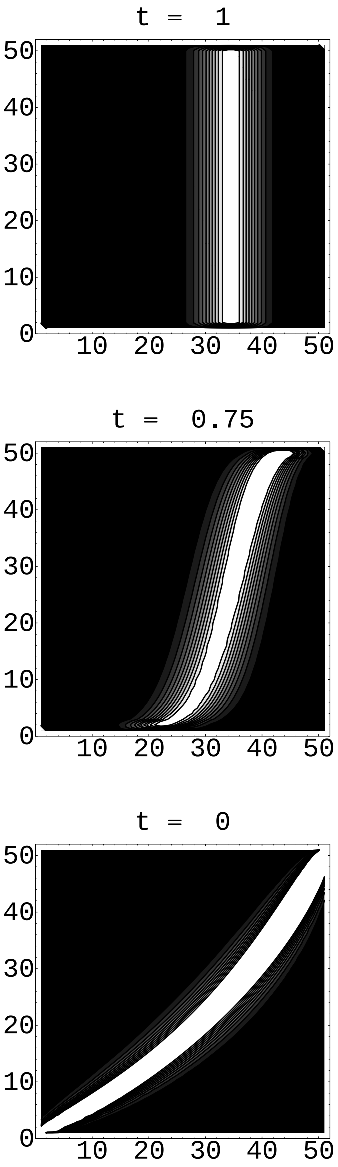

Figure 3: Markov chain models (transition matrices), showing the probabilities of transitioning from a population with $y = n_{A,t}$ copies of species **A** to a state with $x = n_{A,t+1}$ copies of **A** in a single generation. The lighter the contour the higher the probability. In the plot labels, "t" represents τ, the period of resource replenishment.

have the same situation as earlier, with $\tau = 1$ normal niching and therefore the same boundaries between cooperation and competition. But as we reduce τ, first to $\tau = 0.75$, then to $\tau = 0.25$, we see that the boundary does indeed move, and the region of niching success shrinks. In the map for $\tau = 0.25$ we have accomplished our task of making the $r_{ov} = r'_f = 0.6$ point a competitive situation rather than a cooperative one, as we were stuck with when $\tau = 1$ by default. That is, we now have some choice, by setting τ, over which degrees of overlap at which ratios of fitness are allowed to continue to coexist.

Intuitively, the meaning of a shrinking τ is to decrease the importance of the niche counts in the denominators, thereby elevating the role of the objective fitnesses (e.g., f_A, f_{AB}, and f_B). As we continue to decrease τ toward 0, we approach the purely selective niching success map, which is all zeros!

4.1.2 Large τ

The results for $0 \leq \tau \leq 1$ are encouraging, in that they indicate that we can have some limited control over the location, if not the shape, of the cooperative-competitive boundary. In particular, we can decrease the area of cooperation. But this ability begs the question: can we *expand* the area of cooperation? What happens if we *increase* τ past one? Do we move the coop-comp boundary in the other direction? If $\tau = 0$ makes the entire $< r_{ov}, r'_f >$ space competitive, does $\tau \rightarrow \infty$ drive the entire space toward cooperation (all ones)? What is the meaning of $\tau > 1$?

In this section we only begin to consider these questions. First, we look at the empirical data, again for $N = 50$, $g = 200$ generations, and under proportionate selection, using our same Markov chains. Figure 5 shows three such cases, for $\tau = 1.5, 1.8, 2.0$. For the first two, we can see that indeed the coop-comp boundary has moved toward the corner of extreme fitness difference and overlap. But then something strange happens. As τ increases above 1.8, the region of niching failure stops shrinking and begins to "curl around" the bottom of the cooperative region, showing a distinct disadvantage for non-overlapping niches!

What is the physical meaning of this trend? As we increase τ above one, we are punishing higher niche counts much more than smaller niche counts. This tends to help the survival rate of niche pairs that include high, but not complete, overlap. One could imagine a natural analog of this situation, if one is creative enough! It might be the case that competition for a resource so drains the antagonists, or their resource, that everyone loses, in that the total out-

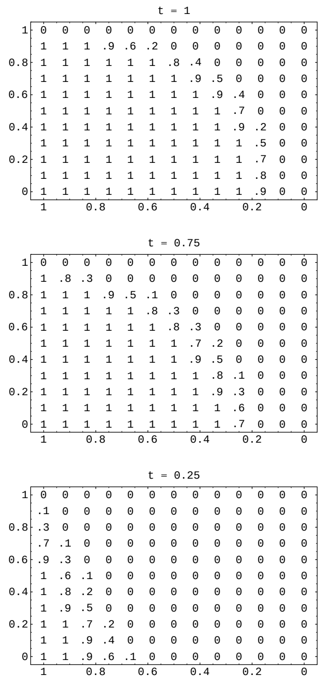

Figure 4: Decreasing τ (labeled "t" above) below 1 results in more competition and less tolerance of overlap and fitness difference, until $\tau = 0$ and one niche is *always* selected.

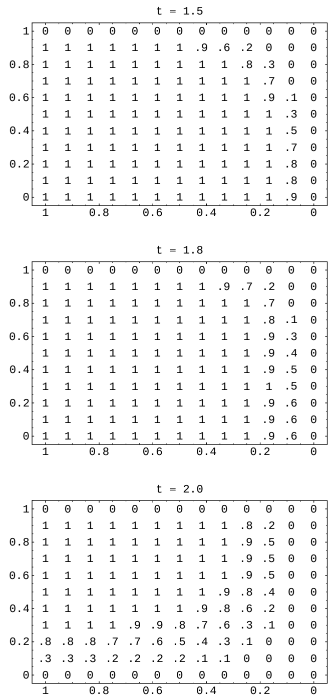

Figure 5: Increasing τ above 1 leads to increased areas of cooperation, up to a point! (See Figure 2 for axes labels.)

put of the resource is lower than it would be for fewer competitors. But for high enough τ, as in the bottom of Figure 5, I believe that the feedback effect of the niche count on the shared fitness becomes so significant that larger subpopulations suffer catastrophic swings in population size. If the two niches are overlapped, the large swings are damped somewhat, as the strong negative effect of high niche counts is shared through the overlap.

At any rate, it is clear that values of τ greater than one have some potential to be useful, but warrant further investigation and must be set with care in the meantime.

5 RELATIONSHIP TO OTHER WORK: τ AND POWERSHARING

Other researchers, particularly in the areas of classifier systems and immune networks, have similar mechanisms for controlling the relative effects of niche counts versus objective fitness. In one particular case, that of "powersharing", the effects are the same. We consider the use of powersharing to solve the *massively multimodal problem.*

As Goldberg, Deb, and Horn (1992) observed, current implementations of sharing are inflexible. They do not allow the user to increase or decrease the relative importance of absolute (i.e., unshared) fitness. Thus the straightforward implementation of fitness sharing failed to maintain all 32 global optima on the *massively multimodal* (MM) problem. Selection with sharing was allocating too many population slots to some of the > 5 million local optima. To increase the relative importance of objective fitness, hence favoring the global optima, Goldberg, et. al. scaled objective fitnesses before dividing by the niche counts:

$$f_{sh,A} = \frac{f_A^\beta}{n_A},$$

where $1 \leq \beta$. Thus the "powershared" equilibrium condition,

$$\frac{f_A^\beta}{n_A} = \frac{f_B^\beta}{n_B},$$

allocates more individuals to more fit niches as β increases. In particular, they were able to successfully discriminate (and maintain) all 32 globals when $\beta = 17$.

We note here that taking the βth root of both sides of the equation above yields

$$\frac{f_A}{\sqrt[\beta]{n_A}} = \frac{f_B}{\sqrt[\beta]{n_B}},$$

where $0 < 1/\beta \leq 1$. Thus powersharing can be mapped to our resource replenishment scheme, with $\tau = \frac{1}{\beta}$, in that both mechanisms yield the same ordering of shared fitness. Although the actual shared fitness values will in general be scaled differently between the two schemes, the ordering will be the same and so any rank based selection method should yield identical selection dynamics. Indeed, Horn (1997) used the inverse of β in the denominator and called this *rootsharing*. He showed that rootsharing performed identically to powersharing on the MM problem.

Thus we are encouraged to think that resource replenishment is a simple and natural way to explain and justify the success of numerous constructions for controlling the niching effect in various domains. We also hold out hope that this approach can piggyback on the success of similar and related methods by solving the same hard problem domains via resource sharing.

References

Booker, L. B. (1982). Intelligent behavior as an adaptation to the task environment. *Dissertation Abstracts International, 43*(2), 469B. (University Microfilms No. 8214966)

Deb, K. (1989). *Genetic algorithms in multimodal function optimization.* Masters thesis and TCGA Report No. 89002. Tuscaloosa, AL: Department of Engineering Mechanics, University of Alabama.

Deb, K., & Goldberg, D. E. (1989). An investigation of niche and species formation in genetic function optimization. In J. D. Schaffer (Ed.), *Proceedings of the Third International Conference on Genetic Algorithms.* San Mateo, CA: Morgan Kaufmann, 42–50.

Goldberg, D. E. (1989a). *Genetic algorithms in search, optimization, and machine learning.* Reading, MA: Addison-Wesley.

Goldberg, D. E., Deb, K., & Horn, J. (1992). Massive multimodality, deception, and genetic algorithms. In R. Männer & B. Manderick, (Ed.s), *Parallel Problem Solving From Nature, 2.* Amsterdam: North-Holland, 37–46.

Goldberg, D. E., & Richardson, J. (1987). Genetic algorithms with sharing for multimodal function optimization. In J. Grefenstette, (Ed.), *Proceedings of the Second International Conference on Genetic Algorithms*, Hillsdale, NJ: Lawrence Erlbaum Associates, 41–49.

Horn, J. (1993). Finite Markov chain analysis of genetic algorithms with niching. In S. Forrest, (Ed.), *Proceedings of the Fifth International Conference on Genetic Algorithms.* San Mateo, CA: Morgan Kaufmann, 110–117.

Horn, J. (1997). *The Nature of Niching: Genetic Algorithms and the Evolution of Optimal, Cooperative Populations.* Ph.D. thesis, University of Illinois at Urbana-Champaign, (UMI Dissertation Services, No. 9812622).

Horn, J., & Goldberg, D. E. (1996). Natural niching for evolving cooperative classifiers. In J. R. Koza, D. E. Goldberg, D. B. Fogel, & R. L. Riolo (Ed.s), *Genetic Programming, Proceedings of the First Annual Conference 1996.* Cambridge, MA: The MIT Press. 553–564.

Horn, J., & Goldberg, D. E. (1999). Toward a control map for niching. *Foundations of Genetic Algorithms, Volume 5.* Morgan Kaufmann. (to appear)

Horn, J., Goldberg, D. E., & Deb, K. (1994). Implicit niching in a learning classifier system: nature's way. *Evolutionary Computation, 2*(1). 37–66.

Huberman, B. A. (1988). The ecology of computation. In B. A. Huberman (Ed.), *The Ecology of Computation.* Amsterdam, Holland: Elsevier Science Publishers B. V. 1–4.

Mahfoud, S. W. (1995). Population size and genetic drift in fitness sharing. In L. D. Whitley & M. D. Vose, (Ed.s), *Foundations of Genetic Algorithms, 3 (FOGA 3).* San Francisco, CA: Morgan Kaufmann. 185–224.

McCallum, R. A., & Spackman, K. A. (1990). Using genetic algorithms to learn disjunctive rules from examples. In B. W. Porter & R. J. Mooney, (Ed.s), *Machine Learning: Proceedings of the Seventh International Conference.* Palo Alto, CA: Morgan Kaufmann. 149–152.

Neri, F., & Saitta, L. (1995). Analysis of genetic algorithms evolution under pure selection. In L. J. Eshelman, (Ed.), *Proceedings of the Sixth International Conference on Genetic Algorithms.* San Francisco, CA: Morgan Kaufmann. 32–39.

Smith, R. E., Forrest, S., & Perelson, A. S. (1993). Searching for diverse, cooperative populations with genetic algorithms. *Evolutionary Computation, 1*(2). 127–150.

Wilson, S. W. (1987). Classifier systems and the animat problem. *Machine Learning, 2.* 199–228.