The Behavior of Spatially Distributed Evolutionary Algorithms in Non-Stationary Environments

Jayshree Sarma
Computer Science Dept.
George Mason University
Fairfax, VA 22030

Kenneth De Jong
Computer Science Dept.
George Mason University
Fairfax, VA 22030

Abstract

Traditional EAs lose diversity fairly quickly due to the strong selection pressures used to achieve good optimization performance, and thus have difficulty with non-stationary environments (fitness landscapes) unless significant algorithmic changes are made. Decentralized and spatially distributed EAs intuitively appear to be more robust in their ability to perform well in both stationary and non-stationary problem domains. We explore this hypothesis with a set of empirical studies that, although preliminary in nature, supports this claim and provides some additional insights into properties of spatially distributed EAs.

1 INTRODUCTION

An evolutionary algorithm (EA) is adaptively searching an unknown space of potential solutions to find high performance solutions. When the environment varies with time it is essential that the EA be able to adapt to these changes by adjusting its search pattern. Most traditional EAs consist of a single population in which all individuals are competing with each other for reproductive and survival rights. To achieve good optimization performance, selection pressure is generally quite high and the population quickly loses diversity as it converges to a solution. Hence, when applied to problems in which the environment (fitness landscape) is changing, traditional EAs do not perform well without significant algorithmic modifications.

In the recent years there has been increasing interest in more decentralized and spatially distributed EAs to take advantage of the power of distributed and highly parallel computing environments. Of interest to us is the development of various forms of spatially distributed EAs in which the population is distributed over a grid of cells with each individual assigned to a cell. Each cell has a specified local neighborhood from which parents are selected. The process of selecting, recombining and evaluating individuals is done in parallel in all the cells. Since these neighborhoods overlap, there is an implicit migration of information.

Unlike the traditional EAs, in spatially distributed EAs the genetic information is spread out on the grid. Since selection is performed using local neighborhoods the genetic information is dispersed slowly through the grid. Each of these neighborhoods can conceivably be searching different regions of the search space and thus act as a reservoir which can be tapped into when the environment changes. Additionally the selection pressure can be controlled by either changing the local selection algorithm or altering the ratio of the neighborhood radius to the grid radius. Thus no algorithmic changes are necessary to change the selection pressure in a spatially distributed EA. Hence the spatially distributed EAs appear to have the potential to be applied to either stationary or non-stationary problems with no significant changes. In this paper we explore this idea and gain additional insights into the properties of spatially distributed EAs.

2 BACKGROUND

In this section we will first provide some additional details about the decentralized and spatially distributed EAs under evaluation. Then we will summarize the different studies which have applied traditional EAs to changing environments.

2.1 SPATIALLY DISTRIBUTED EAs

One has to make several important decisions in order to design effective spatially distributed EAs. The

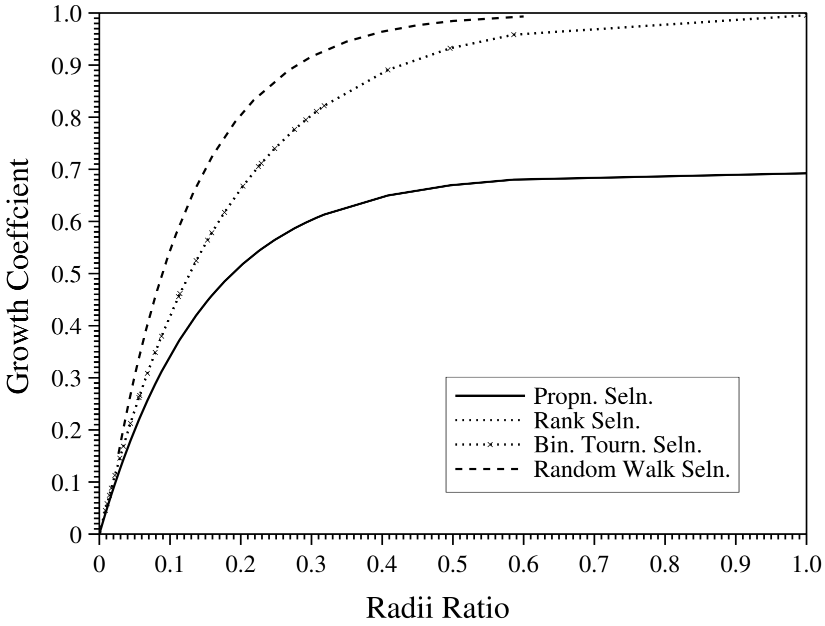

Figure 1: Growth curve coefficients for fitness proportional, linear ranking, binary tournament, and random walk (path elitism) selections as a function of the radii ratio.

topology to be used for the population is generally dictated by the underlying hardware. The other decisions are the neighborhood size and the selection algorithm to be used in the local neighborhoods. Usually these choices are made based on empirical studies (Collins and Jefferson 1991; Gorges-Schleuter 1989; McInerney 1992; Spiessens and Manderick 1991). An EA can effectively search a space of potential solutions only if a proper balance between exploration and exploitation is maintained. In traditional EAs the selection pressure induced by the different selection algorithms is well understood (Bäck 1995; Goldberg and Deb 1991). However, in spatially distributed EAs, selection pressure is the result of the interacting effects of both local selection algorithms and neighborhood topologies.

Recent studies have provided new analytical tools for designing spatially distributed EAs (Sarma and De Jong 1996; Sarma and De Jong 1997). In those papers it was shown that the *emergent* selection pressure induced by selection methods in spatially distributed EAs is qualitatively similar but quantitatively weaker than its traditional counterparts. In these same studies, it was also shown that for a given selection algorithm, there is an inverse exponential relationship between the radii ratio (the neighborhood radius to the radius of the population topology) and the growth coefficient of the logistic equation describing selection pressure. This result is shown in Figure 1. (The radius is computed using a distance measure adapted from spatial analysis techniques. The selection pressure induced by selection methods is formally studied by analyzing the growth curves of the best fitness class in an infinite population model in which only the selection operator is active. These growth curves are approximated by a logistic equation of the form

$$P_{b,t} = \frac{1}{1 + (\frac{1}{P_{b,0}} - 1)e^{-at}}$$

where a is the growth coefficient (which controls the growth rate), $P_{b,t}$ is the proportion of the best fitness class in the population at time t, and $P_{b,0}$ is the initial proportion of the best fitness class.)

This graph provides insight into the selection pressure induced by different selection algorithms for a given combination of population topology and neighborhood radius. If we read horizontally across the graph we can find the neighborhood radius that will induce similar selection pressure for the selection methods. Reading vertically across the graph the selection pressure induced by the selection strategies for a given neighborhood radius and population topology can be identified. These tools were tested on stationary optimization problems and shown to be useful in making effective design decisions in these domains (Sarma and De Jong 1998).

2.2 NON-STATIONARY ENVIRONMENTS

An environment is defined to be non-stationary if the response from the environment for the inputs varies with time. The value of the objective function *f(x)* is dependent on time t; in other words

$$f(x) = g(x, t)$$

The response can change either every time step by a certain quantity or every δ time steps. In this section we review the different types of modifications usually made to the traditional EAs to "track" the changing environment.

The different studies applying traditional GAs to non-stationary environments can be grouped into two categories, namely, (1) expanding the memory of the traditional GA, and (2) using some form of the mutation operator to introduce diversity in the population. The first approach is to modify the representation scheme so that redundant information from previous generations can be carried forward. This serves to increase the memory of the traditional GA. The reasoning is that if the environment suddenly changes, this redundant data will help the GA to quickly adapt to the new environment. The second approach modifies the mutation operator to introduce diversity into the population whenever the fitness landscape changes. Both these approaches have their advantages and disadvantages. Increasing the population size does not help the traditional GA track the moving optimum as it still needs some modification to adapt to the changing environment.

On of the earliest studies was done by Pettit and Swigger (1983) who applied the traditional GA to track a target solution which changed at random every generation. The landscape fluctuation was varied from slow to fast. They found that the traditional GA was unable to successfully track the changing environment even when the rate of fluctuation was slow.

Goldberg and Smith (1987) used a diploid (pairs of chromosomes) representation instead of a haploid (single chromosome) representation to effectively track a non-stationary 0–1 blind knapsack problem that periodically changed between two different states. A new operator called the "dominance" operator was introduced to handle the new representation. The diploid representation helped to carry redundant information along till the need arose to use them. They concluded that a GA with a diploid representation was able to track the changing environment better than a GA with a haploid representation. Another method to expand the memory of the traditional GA was to use a multi-layered (a tree structured) chromosome which helped to successfully track the optimum in a non-stationary 0–1 blind knapsack problem (Dasgupta and McGregor 1992).

The other studies have all concentrated on using the mutation operator to introduce diversity into the population whenever the environment changes. The *triggered hypermutation* operator was used in the traditional GA to introduce diversity into a population where the environment is changing continuously (Cobb 1990). This hypermutation operator was triggered when the time-averaged best performance declined. Essentially a higher mutation rate was used to change the population composition so that the new optimum can be tracked. Grefenstette (1992) used a method called the *random immigrants* to increase the diversity in the population to successfully track the moving optimum in a periodically changing environment. Each generation a certain percentage of the population is replaced by new randomly created individuals which kept the exploration level high.

Cobb and Grefenstette (1993) compared these two mechanisms (triggered hypermutation and random immigrants) and the traditional GA (with higher mutation rate) on different forms of non-stationary environments. They concluded that triggered hypermutation will be able to perform well in slowly changing non-stationary environments while the random immigrant approach is better suited for sudden massive changes in the environment. Yet another modification made to increase diversity in a traditional GA is the use of a *shift* operator which is initiated when the time averaged best performance decreased (Vavak, Fogarty, and Jukes 1996; Vavak, Jukes, and Fogarty 1997). The shift operator enabled a local search within a certain radius of each individual. This is similar to mutating the individual but here each individual had its own mutation level. The study showed that this mechanism was able to track the optimum in a slowly changing environment.

All these approaches involve making algorithmic changes to an EA in order to perform well in certain types of non-stationary environments. In general, in real world problems one is not sure when the changes are occurring, how fast or slow these changes are, and how drastic these changes are. Hence it is hard to decide beforehand which EA would be appropriate for the given non-stationary problem. It would be desirable to have a EA which can be used for both stationary and non-stationary environments without many modifications.

We have an intuitive sense that spatially distributed EAs may in fact have such a property. In these EAs the genetic information is spatially distributed and selection is done within specified neighborhoods. This spatial distribution of the population can act as an extended memory for the EA as the various neighborhoods can potentially search different areas in the search space, thus maintaining in a natural way the diversity needed to deal with non-stationary environments. Moreover, the selection pressure in these EAs can be controlled by a single parameter, the neighborhood radius, without any algorithmic changes. These properties suggest that spatially distributed EAs should be quite useful with changing environments. We will explore this hypothesis in the following sections.

3 EXPERIMENTAL SETUP

We are interested in understanding the behavior of the spatially distributed EAs in a time varying environment. To understand this better, our initial experiments were designed to test the ability of these EAs to track the location of a moving optimum. We used a two-dimensional problem which consists of 14 sinusoidal hills with varying heights and a single optimum. (This is referred to as Landscape A in Cobb and Grefenstette (1993).) The height of the tallest hill is 60 (the optimum). The range of each dimension is +32.768 to -32.768. Each dimension is represented by 16 bits.

The two types of non-stationarity we consider in this study are: (1) the translation of the maximum hill (moving only the optimum) while keeping the rest of the landscape fixed, and (2) the translation of the entire landscape (all the hills) either in a fast or slow mode. The fast or slow mode here refers to how far the hills move in the two dimensions. In both cases there is no distortion of the hills, which implies that the widths and heights are constant.

A spatially distributed GA can be designed such that the selection pressure induced is similar to the selection pressure induced in a traditional centralized GA. It would be interesting to see what, if any, performance difference can be seen in these two GAs. Hence, the comparisons involve two GAs: a traditional centralized version and a spatially distributed one.

All the experiments in this paper are averaged over 100 runs. The experiments were stopped after 200 generations was reached. The best individual found was kept track of every generation. Two point crossover and bit flipping mutation is used in all experiments. The standard setting of 0.60 for crossover rate is used. Cobb and Grefenstette (1993) in their study concluded that a traditional GA with higher than normal mutation rate was better able to track moving optima, so we used a mutation rate of 0.01 instead of the standard setting of 0.001. We used fitness proportional selection since weaker selection pressure is likely to be more beneficial in changing environments. A population size of 225 was used for both GAs. In the case of the spatially distributed GA this took the form of a 15×15 grid.

The spatially distributed GA was evaluated using two different selection pressures: one which was as similar as possible to the traditional GA and one which was considerably weaker. As noted earlier, this is easily achieved using the results in Figure 1. In this case a radii ratio of 0.33 results in a selection pressure similar to the traditional GA, and a radii radio of 0.15 produces much weaker selection pressure.

4 EXPERIMENTAL RESULTS

The first type of non-stationary environment used for testing the behavior of the spatially distributed EA was a periodically changing landscape in which the the location of the maximum hill was moved randomly to a different spot every 20 generations. The rest of the hills remained stationary. The reason for choosing 20 generations is because it was considered to be ample time for a traditional GA to converge to the new optimum (Grefenstette 1992).

Figure 2 shows the average best-so-far curves for the three EAs: the traditional GA, and the two spatially distributed GAs with differing selection pressures. Since the value of the maximum is not changing (fixed at 60), but only the location of the optimum, optimal tracking would result in best-so-far curve close to 60. The results indicate that the traditional GA is doing much worse than either of the spatially distributed versions. Recall that the spatially distributed GA with a radii ratio of 0.33 is as similar as possible in every respect to the traditional GA. Hence, the performance improvement appears to be specifically attributable to the spatially distributed population and local neighborhoods.

Notice also that, for this particular environment, weakening the selection pressure even further (radii ratio of 0.15) improved tracking capabilities initially, but weakened them somewhat in the long run.

The second type of non-stationarity studied involved shifting the entire landscape. We call this a drifting

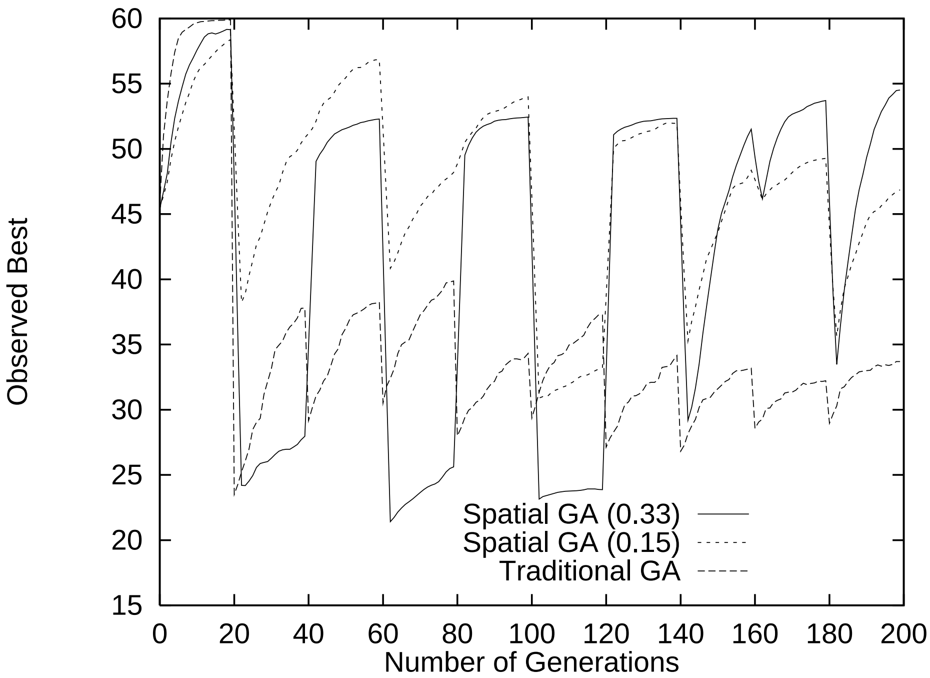

Figure 2: Performance curves for spatial GAs (radii ratios 0.33 and 0.15) and a traditional GA for the periodically changing environment.

environment. This was simulated by moving all the hills, either in slow mode or in fast mode, every 5 generations. In the slow mode, the location of the hills were incremented by ± 2.0. In the fast mode the location of the hills were incremented by ± 5.0. Each dimension can either increase or decrease by the specified amount. This movement is done independently in each dimension. The landscape is varied slowly (increments of ± 2.0) every 5 generations for the first 50 generations then switched to larger increments (± 5.0) for the next 50 generations. This translation is combined with the random movement of the location of the max hill at fixed intervals. Hence after 100 generations only the location of the maximum hill is varied every 20 generations. This setup is similar to the one used by Cobb and Grefenstette (1993).

Figure 3 shows the average best-so-far curves for the three EAs as before. Again, the value of the maximum is not changing (fixed at 60), but only the location of the optimum, so optimal tracking would result in best-so-far curve close to 60. There are three different phases in the graph. The first 50 generations is phase one and show the results when the landscape is drifting slowly. Generations 50 to 100 is phase two and the landscape is drifting in a fast mode. The last hundred generations is phase three and in this case only the optimum peak is moving every 20 generations.

Here the results are not as clear cut as before. In phase one, all three GAs appear to be able to track the slowly drifting aspects reasonably well initially. But in later generations of phase one both the spatial GAs seem to be able to track the optimum better than the traditional GA. In phase two where the landscape is drifting in a fast mode, the spatial GA with lower selection pressure (radii=0.15) is able to track the optimum at least initially. After generation 80 the three GAs are unable to track the optimum. In phase three there is occasional abrupt changes in location of the optimum. In this phase, both the spatial GAs appear to have better tracking ability than the traditional GA.

5 DISCUSSION AND FUTURE WORK

We have begun to explore the abilities of spatially distributed GAs for solving problems in domains involving non-stationary environments. Although preliminary in nature, the experiments presented suggest that spatially distributed GAs are able to handle non-stationary fitness landscapes considerably better than traditional GAs. Although there is related research that suggests that traditional GAs can be modified to work better in changing environments, an apparent virtue of spatially distributed GAs is that they can

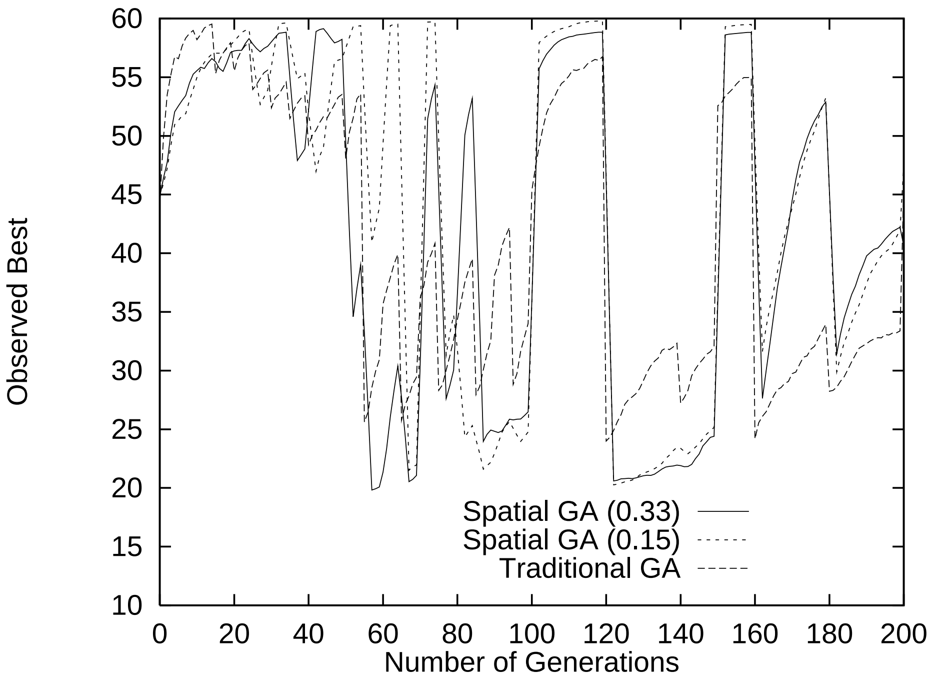

Figure 3: Performance curves for spatial GAs (radii ratios 0.33 and 0.15) and a traditional GA for the drifting environment.

handle both static and dynamic landscapes well without the need for algorithmic modification.

Clearly more work is required to understand these kinds of EAs better. We have not, for example, experimented with non-stationary landscape in which the heights of the peaks change over time. In addition, it is clear that the rate of change of the environment relative to the internal speed of evolution is an important factor.

Another interesting and unexplored aspect of these spatially distributed EAs is that they provide a simple way to dynamically alter the selection pressure by adjusting the radii ratio. The capability to do so might enhance their ability to better handle non-stationary environments.

Although our experiments so far have involved spatial and traditional GAs, we believe that similar behavior will be seen for other traditional EAs such as evolutionary strategies (ES) and evolutionary programming (EP). We are currently looking at these and other issues, and expect that such results will provide an even clearer picture of the properties of spatially distributed EAs.

References

Bäck, T. (1995). *Evolutionary Algorithms in Theory and Practice*. New York: Oxford University Press.

Cobb, H. G. (1990). An investigation into the use of hypermutation as an adaptive operator in genetic algorithms having continuous, time-dependent nonstationary environments. Technical Report Memorandum Report 6760, Naval Research Laboratory.

Cobb, H. G. and J. J. Grefenstette (1993). Genetic algorithms for tracking changing environments. In *Proceedings of the Fifth International Conference on Genetic Algorithms*, Urbana–Champaign, IL, pp. 523–530. Morgan Kaufmann.

Collins, R. J. and D. R. Jefferson (1991). Selection in massively parallel genetic algorithms. In *Proceedings of the Fourth International Conference on Genetic Algorithms*, San Diego, CA, pp. 249–256. Morgan Kaufmann.

Dasgupta, D. and D. R. McGregor (1992). Nonstationary function optimization using the structured genetic algorithm. In R. Männer and B. Manderick (Eds.), *Proceedings Parallel Problem Solving from Nature - PPSN II*, Brussels, Belgium, pp. 145–154. Elsevier Science Publishers.

Goldberg, D. E. and K. Deb (1991). A comparative analysis of selection schemes used in genetic algorithms. In *Foundations of Genetic Algorithms*, San Mateo, CA, pp. 69–93. Morgan Kaufmann.

Goldberg, D. E. and R. E. Smith (1987). Nonstationary function optimization using genetic algorithms with dominance and diploidy. In *Proceedings of the Second International Conference on Genetic Algorithms*, Cambridge, MA, pp. 59–68. Morgan Kaufmann.

Gorges-Schleuter, M. (1989). ASPARGOS an asynchronous parallel genetic optimization strategy. In *Proceedings of the Third International Conference on Genetic Algorithms*, Fairfax, VA, pp. 422–427. Morgan Kaufman.

Grefenstette, J. J. (1992). Genetic algorithms for changing environments. In R. Männer and B. Manderick (Eds.), *Proceedings Parallel Problem Solving from Nature - PPSN II*, Brussels, Belgium, pp. 137–144. Elsevier Science Publishers.

McInerney, J. (1992). *Biologically Influenced Algorithms and Parallelism in Non–linear Optimization*. Ph. D. thesis, University of California, San Diego.

Pettit, E. and K. M. Swigger (1983). An analysis of genetic–based pattern tracking and cognitive–based component tracking models of adaptation. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-83)*, pp. 327–332. Morgan Kaufmann.

Sarma, J. and K. De Jong (1998). Selection pressure and performance in spatially distributed evolutionary algorithms. In *Proceedings of the World Congress on Computatinal Intelligence*, Anchorage, Alaska, pp. 553–557. IEEE Press.

Sarma, J. and K. A. De Jong (1996). An analysis of the effects of neighborhood size and shape on local selection algorithms. In *Proceedings Parallel Problem Solving from Nature - PPSN IV*, Berlin, Germany, pp. 236–244. Springer-Verlag.

Sarma, J. and K. A. De Jong (1997). An analysis of local selection algorithms in a spatially structured evolutionary algorithm. In *Proceedings of the Seventh International Conference on Genetic Algorithms*, East Lansing, MI, pp. 181–186. Morgan Kaufmann.

Spiessens, P. and B. Manderick (1991). A massively parallel genetic algorithm: Implementation and first analysis. In *Proceedings of the Fourth International Conference on Genetic Algorithms*, San Diego, CA, pp. 279–287. Morgan Kaufmann.

Vavak, F., T. C. Fogarty, and K. Jukes (1996). A genetic algorithm with variable range of local search for tracking changing environments. In *Proceedings Parallel Problem Solving from Nature - PPSN IV*, Berlin, Germany, pp. 376–385. Springer-Verlag.

Vavak, F., K. Jukes, and T. C. Fogarty (1997). Adaptive combustion balancing in multiple burner boiler using a genetic algorithm with variable range of local search. In *Proceedings of the Seventh International Conference on Genetic Algorithms*, East Lansing, MI, pp. 719–726. Morgan Kaufmann.