Fuzzy Dominance Based Multi-objective GA-Simplex Hybrid Algorithms Applied to Gene Network Models

Praveen Koduru[1], Sanjoy Das[1], Stephen Welch[2], and Judith L. Roe[3]

[1] Electrical and Computer Engineering
[2] Department of Agronomy
[3]Division of Biology
Kansas State University
Manhattan, KS 66506

Abstract. Hybrid algorithms that combine genetic algorithms with the Nelder-Mead simplex algorithm have been effective in solving certain optimization problems. In this article, we apply a similar technique to estimate the parameters of a gene regulatory network for flowering time control in rice. The algorithm minimizes the difference between the model behavior and real world data. Because of the nature of the data, a multi-objective approach is necessary. The concept of fuzzy dominance is introduced, and a multi-objective simplex algorithm based on this concept is proposed as a part of the hybrid approach. Results suggest that the proposed method performs well in estimating the model parameters.

1 Gene Regulatory Network Models

Molecular geneticists are rapidly deciphering the genomes of an increasing number of organisms. As of November 2003, 166 organisms had completely sequenced genomes with another 775 in progress [1]. The current challenge is to understand how the genes in each organism interact with each other and the environment to determine the characteristics (*i.e.*, the *phenotype*) of the organism. In the agricultural contexts familiar to the authors, this is called the "genotype to phenotype" or "GP" problem. In [2] it has been stated that this problem is the most significant issue confronting crop improvement efforts today.

For over 40 years plant physiologists, systems engineers, and computer scientists have been employing "top-down" analysis methods to predict plant phenotypes based on varietal and environmental inputs [3, 4]. Recently, a "bottom" up approach has been applied [5, 6, 7] that models gene interactions directly at the expression level. Small groups of one to four interacting genes can synthesize a wide variety of signal processing functions including Boolean logic gates, linear arithmetic units, delays, differentiators, integrators, oscillators, coincidence detectors, and bi-stable devices [8]. This is consistent with the apparent small-scale modularity of gene networks [9]. Models of this type extrapolate phenotypes by explicitly tracking the status of key genetic developmental switches, accumulators, *etc.*

K. Deb et al. (Eds.): GECCO 2004, LNCS 3102, pp. 356–367, 2004.
© Springer-Verlag Berlin Heidelberg 2004

Models of this type require efficient, multi-dimensional, multi-objective, derivative-free, global methods for parameter estimation. The problem is characterized by high dimensionality due to the large numbers of genes. Multi-objective optimization methods are appropriate because (1) multiple data types (continuous, discrete, and/or categorical) for both dependent and independent variables make the design of a single objective function problematic, (2) individual data sets come from different sources and may contain within- or between-set inconsistencies not apparent in the metadata, and (3) the models are incomplete and, therefore, may not be equally consistent with every data set. Because actual biophysical systems cannot harbor internal inconsistencies, the Pareto fronts associated with these problems are ideally single points. However, when data and/or model inconsistency exists, the size of the front is a useful measure of its magnitude. Finally, nonlinearities and data discontinuities can lead to exceptionally rough, multi-modal response surfaces (e.g., [7]) that mandate global, derivative free methods.

The following sections of this paper present a new algorithm that posses these features. The algorithm is described and then the algorithm is tested with the following single-gene model that demonstrates the features just described. In [25] the levels of messenger RNA was measured every 3h under short-days (SD, 9h) and long-days (LD, 15h) for *HEADING DATE 1* (*Hd1*), an important flowering time control gene in rice (Oryza sativa). In [8] the authors modeled this data with the equation(s):

$$\frac{d}{dt}(Hd1) = \begin{Bmatrix} R_D \\ R_L \end{Bmatrix} g_{NN}(C(t)) - (Hd1) \begin{Bmatrix} \lambda_D \\ \lambda_L \end{Bmatrix} \quad (1, 2)$$

where R's and λ's are constants and L and D denote light and dark periods. The clock input is C(t) = A*Sin(2π/p + θ) + μ, where A is amplitude, p is period, θ is phase angle, μ is a phase factor and $g_{NN} = \frac{1}{1+\exp(-c)}$ [5]. The state variable, Hd1, is dimensionless as expression levels are routinely normalized. The parameters have to be found such that model satisfies both SD and LD data with minimal MSE error with experimental data. So the approach of multi-objective optimization is used to find the possible solutions. Agronomic research on this point is underway. Thus, possible objective functions are the MSE between the model predicted SD and LD time series data with actual data obtained experimentally.

2 The Multi-objective Evolutionary Approach

Evolutionary algorithms have emerged as one of the most popular approaches for the complex optimization problems [10]. They draw upon Darwinian paradigms of evolution to search through the solution space (the set of all possible solutions). Starting with a set (or population) of solutions, in each generation of the algorithm, new solutions are created from older ones by means of two operations, mutation and crossover. Mutation is accomplished by imparting a small, usually random perturbation to the solution. In a manner similar to the Darwinian paradigm of survival of the fittest, only the better solutions are allowed to remain in a population, the degree of optimality of the solution being assessed through a measure called fitness.

When dealing with optimization problems with multiple objectives, the conventional concept of optimality does not hold good [11, 12, 13, 14]. Hence, the concepts of dominance and Pareto-optimality are applied. Without a loss of generality, if we assume that the optimization problem involves minimizing each objective $e_i(.)$, $i = 1...M$, a solution u is said to dominate over another solution v iff $\forall i \in \{1,2,\ldots,M\}$, $e_i(u) \le e_i(v)$ with at least one of the inequalities being strict, i.e. for each objective, u is better than or equal to v and better in at least one objective. This relationship is represented as $u \prec v$. In a population of solution vectors, the set of all non-dominating solutions is called the Pareto front. In other words, if S is the population, the Pareto Front Γ is given by,

$$\Gamma = \{u \in S \mid \forall v \in S, \neg(v \succ u)\} \tag{3}$$

The simplistic approach of aggregating multiple objectives into a single one often fails to produce good results. It produces only a single solution. Multi-objective optimization on the other hand involves extracting the entire Pareto front from the solution space. In recent years, many evolutionary algorithms for multi-objective optimization have been proposed [14, 15, 16, 17].

We propose a hybrid algorithm that combines genetic algorithms (GAs), an evolutionary algorithm, with a well-known approach for function optimization known as the simplex algorithm [18]. While several GA-simplex algorithms have been proposed, our version is the only one that is equipped to carry out multi-objective optimization. This is accomplished by means of a concept, that we introduce, called fuzzy dominance.

2.1 Fuzzy Dominance

We first introduce the concept of fuzzy dominance. We assume a minimization problem involving M objective functions $e_i(\cdot)$, $i = 1...M$. The solution space, the set of all possible solution vectors, will be denoted as $\Psi \subset \Re^n$, where n is the dimensionality of the multi-objective problem.

Definition 1 *Fuzzy i-dominance by a solution*
Given a monotonically non-decreasing function $\mu_i^{dom} : \Psi \to [0,1]$, $i \in \{1,2,\ldots,n\}$ such that $\mu_i^{dom}(0) = 0$, solution $u \in \Psi$ is said to *i*-dominate solution $v \in \Psi$, if and only if $e_i(u) < e_i(v)$. This relationship will be denoted as $u \prec_i^F v$. If $u \prec_i^F v$, the degree of fuzzy *i*-dominance is equal to $\mu_i^{dom}(e_i(v) - e_i(u)) \equiv \mu_i^{dom}(u \prec_i^F v)$. Fuzzy dominance can be regarded as a fuzzy relationship $u \prec_i^F v$ between u and v [19].

Definition 2 *Fuzzy dominance by a solution*
Solution $u \in \Psi$ is said to fuzzy dominate solution $v \in \Psi$ if and only if $\forall i \in \{1,2,\ldots,M\}, u \prec_i^F v$. This relationship will be denoted as $u \prec^F v$. The degree of

fuzzy dominance can be defined by invoking the concept of fuzzy intersection [19]. If $u \prec^F v$, the degree of fuzzy dominance $\mu^{dom}(u \prec^F v)$ is obtained by computing the intersection of the fuzzy relationships $u \prec_i^F v$ for each i. The fuzzy intersection operation is carried out using a family of functions called *t*-norms, denoted with a $*$. Hence,

$$\mu^{dom}(u \prec^F v) = \underset{i=1}{\overset{M}{*}} \mu_i^{dom}(u \prec_i^F v). \tag{4}$$

Definition 3 *Fuzzy dominance in a population*
Given a population of solutions $S \subset \Psi$, a solution $v \in S$ is said to be fuzzy dominated in S iff it is fuzzy dominated by any other solution $u \in S$. In this case, the degree of fuzzy dominance can be computed by performing a union operation over every possible $\mu^{dom}(u \prec^F v)$, carried out using *t*-co norms, that are denoted with a $\oplus$. Hence the degree of fuzzy dominance of a solution $v \in S$ in the set S is given by,

$$\mu^{dom}(S \prec^F v) = \underset{u \in S}{\oplus} \mu^{dom}(u \prec^F v). \tag{5}$$

Using the above definitions, one can redefine the Pareto front as the set of all solutions in S that are not dominated in S. In other words,

$$\Gamma = \left\{ u \in S \mid \neg(S \prec^F u) \right\}. \tag{6}$$

2.2 The Simplex Algorithm

A simplex in n-dimensions consists of $n+1$ solutions u_k, $k = \{1,2,\ldots n+1\}$ [18]. In a plane, this corresponds to a triangle as shown in Figure 1. The solutions are evaluated in each step and the worst solution w is identified. The centroid of the simplex is then evaluated, excluding the worst solution and the worst point is reflected along the centroid. If c is centroid such that $nc = \sum_k u_k - w$, the reflected solution is

$$r = c + (c - w) \tag{7}$$

Usually, the worst point w is replaced with the reflected point r in the simplex, but if the r is better than any solution in the simplex, the simplex is further expanded as,

$$r_e = c + \eta(c - w) \tag{8}$$

where η is called the expansion coefficient. However, if the reflected solution r is worse than w, the simplex is contracted and the reflected solution is placed on the same side of the centroid. When solution r is not worse than w, but worse than any other solution in the simplex, the simplex is still contracted, but the reflection is

allowed to remain on the other side of the simplex. Reflection is carried out as follows,

$$r_c = c \pm \kappa(c - w) \tag{9}$$

In the above equation, κ is called the contraction coefficient. Solution w is replaced with the new one, r, r_e, or r_c in the next step. The simplex algorithm is allowed to run for multiple steps before it converges.

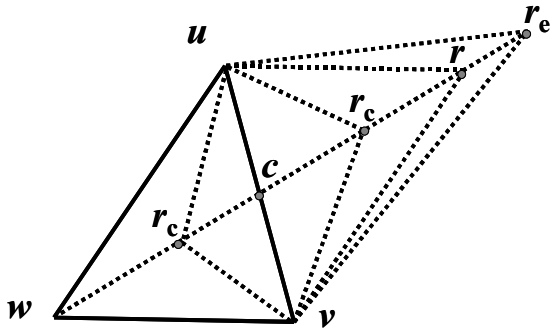

Fig. 1. A simplex in 2 dimensions

2.3 The GA-Simplex Hybrid Algorithm

Since genetic algorithms use a population of individuals, they are capable of performing an exploratory search over the entire solution space. In many complex optimization problems, they are hybridized with local search operations. The local algorithms improve single solutions by exploiting local information around the vicinity of the solutions. Hybrid algorithms combine the advantages of exploration and exploitation forms a new area of research. Hybrid algorithms that use the simplex algorithm discussed earlier, for local search are popular in continuous optimization problems [20, 21, 22, 23].

One of the hybrid approaches proposed uses the simplex algorithm as a post-processor for improving the solutions obtained by a GA [23]. The simplex approach has been used as an operator to improve the solutions obtained from the genetic operations of mutation and crossover in accordance with Lamarckian theory [20]. In our approach, the simplex has been used as an operation within each iteration of the genetic algorithm as in [20, 22]. But only a fraction of the next generation is obtained by carrying out crossover and mutation with the solutions in the present population. The rest of the population is established by using the simplex algorithm. Our approach is similar to [21]. However unlike the approach of [21] where only the elite individuals are used by the simplex algorithm, our approach uses solutions that are chosen from the entire population. Figure 2 is a schematic that shows the approach used in the present research.

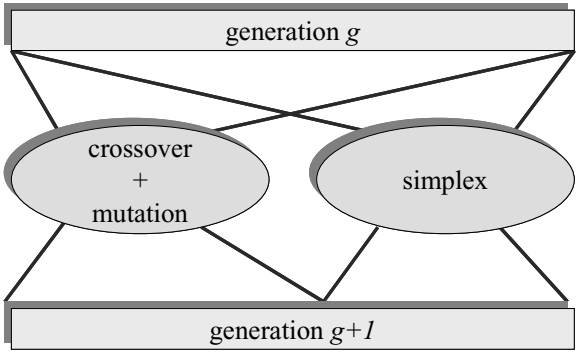

Fig. 2. A schematic of the hybrid approach

In order to apply the simplex algorithm, a total of $n+1$ solutions must be selected from the population S. This is done by first picking at random the $n+1$ solutions from S and computing their centroid C. Any solution vector u at a distance $\|c-u\| > \rho_{simplex}$ is rejected and replaced with another one drawn at random, where $\rho_{simplex}$ is the radius parameter of the simplex approach and $\|\cdot\|$ is the Euclidean norm. This process is repeated until either all the sample solutions fit within the radius $\rho_{simplex}$, or the total replacements exceed r_{max}. After selecting the initial vectors, the simplex algorithm is run for a total of α times. The best $n+1$ solutions are selected to be inserted into the population in the next generation. The genetic algorithm is applied by selecting individuals based on the fuzzy dominance and assigning performing standard crossover and mutation operations that are discussed in the next section.

3 Implementation

3.1 Fuzzy Dominance

In order to calculate the fuzzy dominance relationship between two solution vectors, trapezoidal membership functions were used. Therefore,

$$\mu_i^{dom}\left(u \prec_i^F v\right) = \begin{cases} 0 & \text{if } e_i(v) - e_i(u) < 0, \\ \left(e_i(v) - e_i(u)\right)/p_i & \text{if } 0 \le e_i(v) - e_i(u) < p_i, \\ 1 & \text{otherwise.} \end{cases} \tag{10}$$

In the above equation, the parameter p_i determines the length of the linear region of the trapezoid for the objective function $e_i(\cdot)$. The *t*-norm and *t*-co norms were defined as $x * y = xy$ and $x \oplus y = x + y - xy$. Both are standard forms of operators [19]. This

choice has an interesting property that makes it attractive for our application. While solutions located away from the Pareto front in any population S are always fuzzy dominated in S, those that are more towards the periphery are less dominated. Figure 3 explains this clearly.

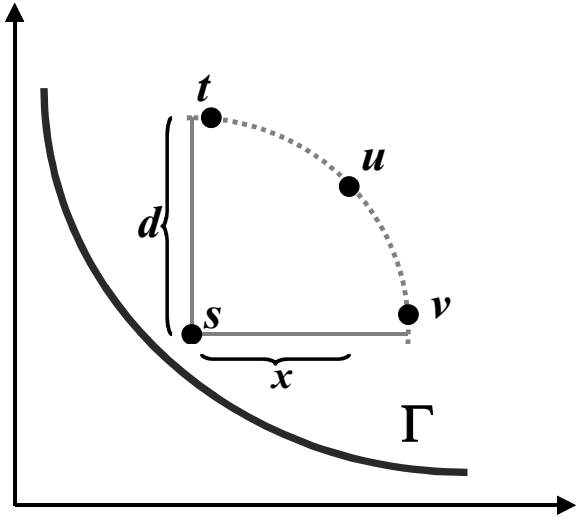

Fig. 3. Effect of fuzzy dominance

Figure 3 shows three solutions t, u and v that are all located at the same distance d from another solution s. Assume that this distance lies within the linear region of the trapezoidal membership function and that the distances of u from s along the horizontal and vertical directions are x and $\sqrt{d^2 - x^2}$ respectively. Hence, $\mu^{dom}\left(u \prec^F v\right) = x\sqrt{d^2 - x^2}$. This will be maximized when $x = \frac{1}{\sqrt{2}} d$, i.e. when u is as far away from the Pareto front as possible. In other words, t and v will be less dominated than u. This property can be extended to more than two objectives and aids the genetic algorithm that uses fuzzy dominance as a measure of inverse fitness during the selection. Solutions that are more peripherally located along the front are preferred to those closer to the center. Hence it assists the GA in maintaining a diverse Pareto front that is as spread as possible. The simplex algorithm works efficiently also, since it identifies worse solutions based on this measure as well and hence, when the simplex is 'flipped' along the centroid, the movement is kept approximately orthogonal to the Pareto front.

The fuzzy dominances of all solutions in the population are calculated at the beginning of each iteration and stored as a two dimensional array, each entry of which is a fuzzy dominance relationship between two solution vectors. However, in order to simplify the calculations, within the simplex algorithm, the fuzzy dominances are considered among the $n+1$ solutions only that are selected by the simplex algorithm.

3.2 Genetic Operators

A tournament selection was implemented in the GA that selected λ individuals at random from the population with replacement, and picked the one with the least fuzzy dominance. An offspring t, was computed from two parents u and v in the following manner,

$$t = \zeta u + (1-\zeta)v \tag{11}$$

where ζ is a uniformly distributed random number in [0, 1].

Solutions were mutated with a probability of β, by adding a random number with zero mean, that followed a Gaussian distribution with a spread σ, according to,

$$u = u + N(0,\sigma) \tag{12}$$

Elitism was implemented by selecting the non-dominated points in a population and copying them to an elite-set. Selection is done on a union of elite-set and current population.

4 Results

We have applied the proposed method to estimate the parameters of the genetic network model in (1,2). The equations involve a total of 8 parameters. Additionally, two initial conditions for both a LD and SD periods exist, which makes up a total of 10 parameters to be computed. The network is simulated and the predicted Hd1 values for the SD as well as LD periods are compared with corresponding experimental data. The mean squared error (MSE) in each of the LD and SD periods predictions with the experimental data is computed, and the goal is to find a set of parameters that simultaneously minimize the MSE for the two periods.

In order to compare the algorithm performance with a standard multi-objective algorithm we applied SPEA as explained in [24] to the same problem. SPEA was chosen over other algorithms since it is one of the most recently proposed algorithms for multi-objective optimization that is also fairly easy to implement. In our test runs we have used a population size of 100 for the proposed method. The mutation rate was set at 0.4 and crossover rate at 0.7. These were found to be optimal for the parameter estimation problem after multiple trails. One simplex was implemented in each generation and the parameters used are α=10, η=1.5 and κ=.5. SPEA was implemented as explained in [24] with a population size of 70. In SPEA clustering was invoked to reduce Pareto fronts whose size exceeded 30 individuals [24]. Tournament selection was used in each algorithm. Both the algorithms were run for a total of 30,000 function evaluations. Figure 4 shows the Pareto front obtained by both algorithms. It is clear that, unlike the proposed algorithm, SPEA was unable to converge to the Pareto front. SPEA was observed to be slower in convergence than our algorithm. Multiple runs were done with different data sets on both algorithms and it was evident that SPEA was unable to reach the Pareto front in the given number of function evaluations. However, in case of our algorithm with the help of simplex and fuzzy dominance a significantly better front was obtained in the same number of function evaluations. We believe that when estimating parameters of genetic network

the fitness landscape contains good minima with basins large enough for the simplex algorithm to converge to with little effort. Figure 5 shows the time series simulation of one of the solutions in the Pareto front along with the experimental data. Figures 6, 7 show the convergence plots of minimum of each of the objectives vs. function evaluations for different mutation rates of 0.1, 0.4 and 0.7. A high mutation rate of 0.4 produced the best possible results. Further research is necessary to explain this phenomenon.

5 Conclusions

In this paper, we have proposed an effective approach to perform parameter estimation of gene regulatory network models. The algorithm, which hybridizes a multi-objective version of the simplex algorithm, based on a newly introduced concept of fuzzy dominance, with a standard genetic algorithm, is shown to converge well for a genetic network model of flowering time control in *Oryza sativa*. Our algorithm consistently outperformed a standard multi-objective optimization approach, SPEA.

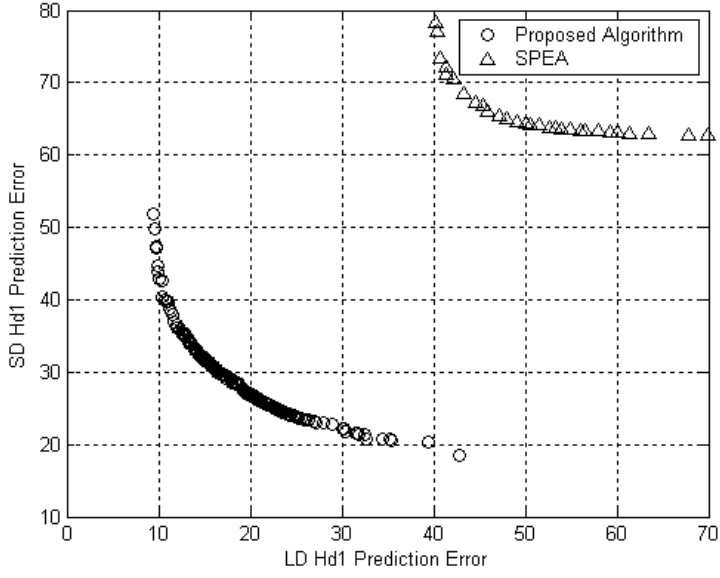

Fig. 4. Pareto front obtained by the proposed algorithm on Heading date prediction and its comparison with SPEA for same initial population.

The proposed algorithm combines the exploratory nature of genetic algorithms with the exploitative behavior of simplex search to carry out parameter estimation of gene regulatory models effectively. We believe that this approach can be applied to similar multi-objective optimization problems as well. Future work will be directed in testing the proposed method for parameter estimation problems with similar network model

with additional objectives and more extensive comparison with more multi-objective methods.

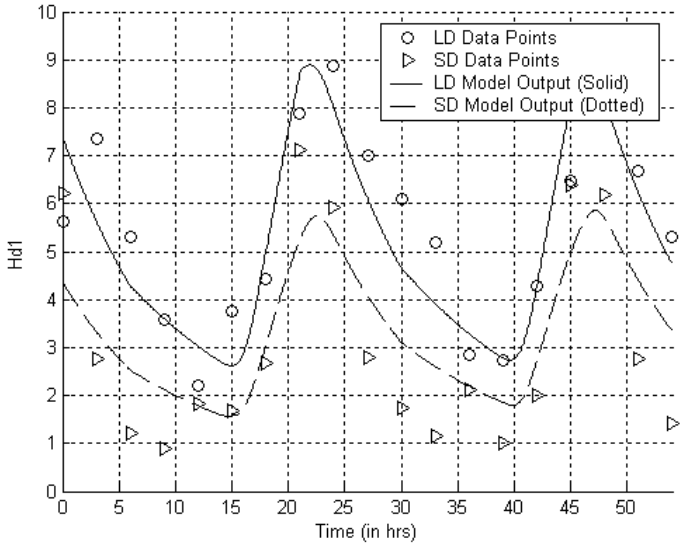

Fig. 5. Time series plot of Hd1 for one of the solutions in Pareto front from proposed algorithm with experimental data

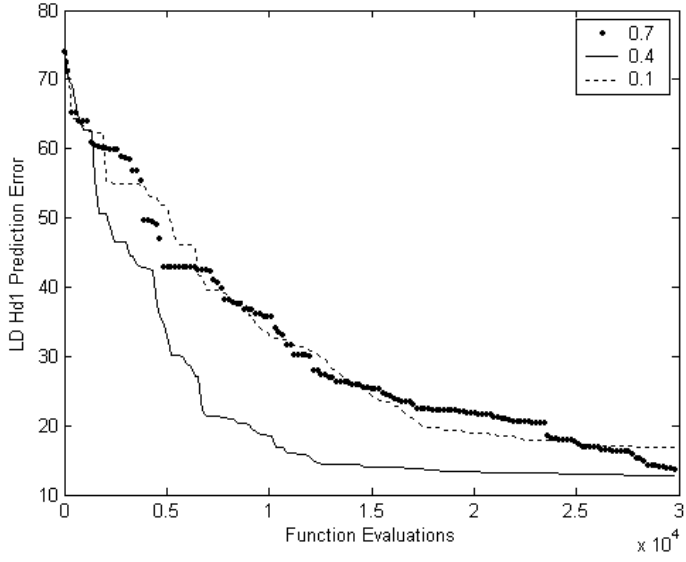

Fig. 6. Convergence of LD Hd1 Prediction error vs. Function evaluations for different mutation rates

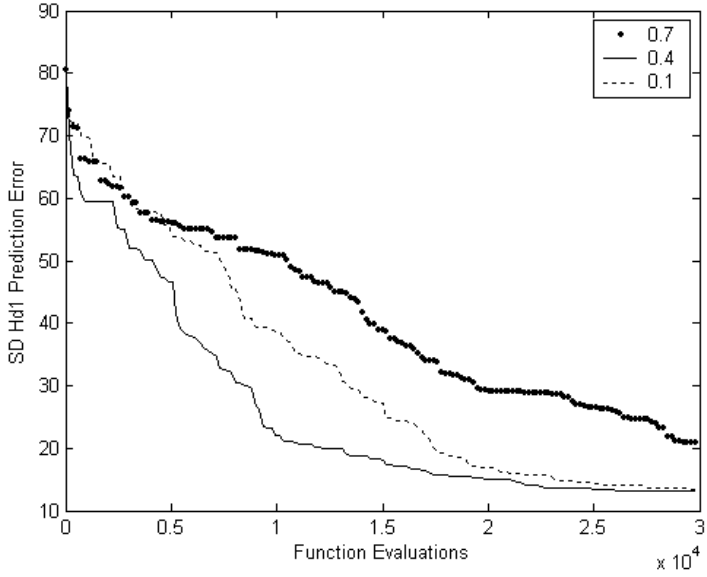

Fig. 7. Convergence of SD Hd1 Prediction error vs. Function evaluations for different mutation rates

Acknowledgment. This research was supported in part by USDA grant 2003-35304-13217 to Kansas State University.

References

1. http://ergo.integratedgenomics.com/GOLD/
2. Cooper, M., Chapman, S.C., Podlich, D.W. & Hammer, G.L. *In Silico Biol.* 2 (2002), 151-164.
3. Sinclar, T.R. and Seligman, N.G.: Crop modelling: From infancy to maturity. Agron. J. 88 (1966) 698-704.
4. Hammer, G. T., Sinclair, S. Chapman, E. van Oostererom.: On systems thinking, systems biology and the *in silico* plant. Plant Physiology. Scientific Correspondence (2004) (*in press*).
5. Welch, S.M., Roe, J.L., and Dong, Z.: A genetic neural network model of flowering time control in Arabidopsis thaliana. Agron. J. (2003) 95, 71-81.
6. Welch, S.M., Dong, Z., and Roe, J.L.: Modelling gene networks controlling transition to flowering in *Arabidopsis.* Proceedings of the 4th International Crop Science Congress, Brisbane, Au. Sep 26 – Oct 1, 2004. (*Under review*).
7. Dong, Z. Incorporation of genomic information into the simulation of flowering time in *Arabidopsis thaliana.* Ph.D. dissertation, Kansas State University (2003).
8. Welch, S.M., Roe, J.L., Das, S., Dong, Z., R. He, M.B. Kirkham.: Merging genomic control networks with soil-plant-atmosphere-continuum (SPAC) models. Agricultural Systems (2004b) (*submitted*).
9. Ravasz, E., Somera, A., Mongru, D., Oltvai, Z., and Baraba´si, A.L.:. Hierarchical organization of modularity in metabolic networks. Science 297 (2002) 1551-1555.

10. D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison Wesley, Reading, MA (1989).
11. Fonseca, C.M., Fleming, P.J.: An overview of evolutionary algorithms in multiobjective optimization, Evolutionary Computation, Vol. 3, no. 1 (1995) 1-16, Spring.
12. Coello Coello, C.A. : A comprehensive survey of evolutionary-based multiobjective optimization techniques, Knowledge and Information Systems, Vol. 1, no. 3 (Aug 1999) 269-308.
13. Van Veldhuizen, D.A., Lamont, G.B.: Multiobjective evolutionary algorithms: Analyzing the state-of-the-art, Evolutionary Computation, Vol. 8, no. 2, (2000) 125-147.
14. Jaszkiewicz, A.: Do multiple-objective metaheuristics deliver on their promises? A computational experiment on the set-covering problem, IEEE Transactions on Evolutionary Computation, Vol. 7, no. 2 (Apr. 2003) 133-143.
15. Haiming, L., Yen,G.G.: Rank-density-based multiobjective genetic algorithm and benchmark test function study, IEEE Transactions on Evolutionary Computation, Vol. 7, no. 4 (Aug. 2003).
16. Knowles, J., Corne, D.: Properties of an adaptive archiving algorithm for storing nondominated vectors**,** IEEE Transactions on Evolutionary Computation, Vol. 7, no. 2 (Apr. 2003) 100- 116.
17. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., da Fonseca, V.G.: Performance assessment of multiobjective optimizers: An analysis and review, IEEE Transactions on Evolutionary Computation, Vol. 7, no. 2 (Apr. 2003) 117-132.
18. Nelder, J.A., Mead, R., A simplex method for function minimization, Computer Journal, Vol 7, no. 4 (1965) 308-313.
19. Mendel, J.M.: Fuzzy logic systems for engineering: A tutorial, Proceedings of the IEEE, Vol 83, No. 3 (March 1995) 345-377.
20. Renders, J.M., Flasse, S.P.: Hybrid methods using genetic algorithms for global optimization, IEEE Transactions on Systems, Man and Cybernetics Part-B, , Vol. 28, no. 2 (Apr. 1998) 73-91.
21. Yen, J., Liao, J.C., Lee, B., Randolph, D.: A hybrid approach to modeling metabolic systems using a genetic algorithm and simplex method, IEEE Transactions on Systems, Man and Cybernetics Part-B, Vol. 7, no. 1 (Feb. 2003) 243-258.
22. Bersini, H.: The immune and chemical crossovers, IEEE Transactions on Evolutionary Computation, Vol. 6, no. 3 (June 2002) 306-313.
23. "Simulation and evolutionary optimization of electron-beam lithography with genetic and simplex-downhill algorithms", *IEEE Transactions on Evolutionary Computation*, pp. 69-82, Vol. 7, no. 1, Feb. 2003.
24. Zitzler, E., Thiele, L.: Multiobjective Evolutionary Algorithms: A comparative case study and the strength Pareto approach, IEEE Transactions on Evolutionary Computation, Vol. 3, no. 4 (Nov. 1999) 257-271.