A Nonparametric Approach to Noisy and Costly Optimization

Brigham S. Anderson BRIGHAM@CMU.EDU
Andrew W. Moore AWM@CS.CMU.EDU
Robotics Institute, Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA 15213 USA
http://www.cs.cmu.edu/~*AUTON/*

David Cohn COHN@JUSTRESEARCH.COM
Just Research, 4616 Henry St, Pittsburgh PA 15213 USA

Abstract

This paper describes PAIRWISE BISECTION: a nonparametric approach to optimizing a noisy function with few function evaluations. The algorithm uses nonparametric reasoning about simple geometric relationships to find minima efficiently. Two factors often frustrate optimization: noise and cost. Output can contain significant quantities of noise or error, while time or money allows for only a handful of experiments. Pairwise bisection is used here to attempt to automate the process of robust and efficient experiment design. Real world functions also tend to violate traditional assumptions of continuousness and Gaussian noise. Since nonparametric statistics do not depend on these assumptions, this algorithm can optimize a wide variety of phenomena with fewer restrictions placed on noise. The algorithm's performance is compared to that of three competing algorithms, Amoeba, PMAX, and Q2 on several different test functions. Results on these functions indicate competitive performance and superior resistance to noise.

1. Problem

The problem of optimizing a function $f : \Re^n \rightarrow \Re$ will be discussed here as finding a local minimum for the function, i.e. a point x^* such that there exists a neighborhood B of x^* with

$$f(x^*) \leq f(x) \quad \forall x \in B. \tag{1}$$

In the noisy case, the observed output y is a combination of the underlying function f and some amount of error or noise. Assuming noise has a mean of zero,

$$y(x) = f(x) + noise \tag{2}$$

For example, during the optimization of a chemical process, if the underlying function $f(x)$ is the yield of a chemical reaction using parameters x (say, temperature and pH,) then the noisy output $y(x)$ might represent the observed yield. The task of the optimizer is to find x^*, the combination of temperature and pH with the greatest expected yield. Many algorithms exist to perform this task, especially in the numerical analysis literature (Press et al., 1992), but they typically require many iterations of the experimental cycle. When experiments are costly or time-consuming, these algorithms are inappropriate.

2. Other Approaches

Many disciplines have methods that are relevant to noisy optimization. Space permits only a brief survey.

Numerical analysis: Numerical methods such as Levenberg-Marquardt (Press et al., 1992) have fast convergence properties, but they can oscillate or diverge to infinity. Furthermore, current numerical methods cannot survive noise.

Stochastic approximation: (Robbins & Monro, 1951) finds roots without the use of derivative estimates. Keifer-Wolfowitz (KW) (Kushner & Clark, 1978) is a related algorithm for noisy optimization. It estimates the gradient by performing experiments in both directions along each dimension of the input space. Based on the estimate, it moves its experiment center and repeats, using decreasing step sizes to ensure convergence. KW's strengths are its aggressive exploration, its simplicity, and its convergence guarantees. Noise can cause it to attempt wild experiments, however. The number of experiments can also be large, since it throws away all data after each gradient estimation.

Amoeba search: Amoeba (Press et al., 1992) searches k-d space using a simplex (i.e. a $k+1$ dimensional polyhedron). The function is evaluated at each vertex. The worst-performing vertex is reflected through the hyperplane defined by the remaining vertices to produce a new simplex that has moved up the estimated gradient. Ingenious simplex transformations let the simplex shrink near the optimum, grow in large linear zones, and ooze along ridges. Amoeba is sensitive to noise, and it is also not efficient with its experiments; it only keeps the most recent $k+1$ results.

Experiment design and Response surface methods (Box & Draper, 1987): A region of interest (ROI) is established at a starting point and experiments are made at positions that can best be used to identify local function properties with low-order polynomial regression. When the gradient is estimated confidently, the ROI is moved accordingly. Quadratic regression locates optima within the ROI. The strength of RSM is that it avoids changing operating conditions based on inadequate evidence, but moves once the data justifies it. Noise can be handled by adding experiments. Experimental efficiency is also good, but a weakness of RSM is that human judgment is needed for choosing the size and shape of the ROI to trade off bias and variance.

Evolutionary computation and Learning automata: Methods such as genetic algorithms begin by sampling uniformly, but then bias later samples in favor of the experiments that had good outcomes. There is a vast literature of refinements of such methods. These approaches need thousands, sometimes millions, of evaluations, so experimental efficiency is very low.

PMAX: Based on the data from the experiments so far, PMAX uses a non-linear function approximator to estimate the underlying function $f(x)$. The next experiment is taken at the point that maximizes the estimate of f. This approach has been used with a decision-tree approximator (Salganicoff & Ungar, 1995), with neural nets (in many commercial products), and with locally weighted regression (Moore & Schneider, 1996). Variations of PMAX include taking the next experiment not at the predicted optimum, but instead where the confidence intervals are widest, or where the top of the confidence interval is maximized (Moore & Schneider, 1996), or in accordance with the Interval Estimation heuristic (Kaelbling, 1990).

Empirically, we have found that PMAX using locally weighted regression as the function approximator is often faster than more sophisticated alternatives (Moore & Schneider, 1996). However it has some serious drawbacks:

- First, one must solve the bias-variance tradeoff. This is often determined automatically using cross-validation (Moore et al., 1994), but this proves difficult with a set of very few, weirdly distributed datapoints obtained during optimization. Empirically we have observed dismal performance when attempting this.
- Second, PMAX is very expensive. It needs to train a function approximator each time an experiment is made, and then the approximate function must be numerically optimized to produce the suggested experiment.
- Third, PMAX can get stuck in "hallucinated" optima.
- Fourth, and most importantly, it assumes a locally smooth function. Discontinuities are disastrous for PMAX.

Q2: the Q2 algorithm (Moore et al., 1998) attempts to achieve second order (quadratic) convergence by fitting local quadratics in a Newton-like method. Unlike RSM it attempts to entirely automatically determine a good region of interest, i.e. a region predicted to with high probability contain a local optimum, that is as small as possible (so as to minimize bias in the quadratic approximation) but not so small that there is no confidence in the optimum location. Q2 performs well on a variety of noisy tasks subject to the assumptions of a continuous underlying $f(x)$ and Gaussian noise. Experimental efficiency appears to be competitive with Amoeba and PMAX.

3. Nonparametric Statistics

Nonparametric statistics have been heavily used in many areas of data analysis and machine learning for years (Sprent, 1989). They rely on taking numerical operations on the data and then testing for evidence of some effect. They usually do so by empirically looking at the distribution of this property implied by the null hypothesis, and then testing whether the current data set's property lies in or out of the empirical null distribution.

Nonparametric statistics have the advantage of having fewer assumptions about the data, and can therefore be applied more broadly and with less trepidation.

Surprisingly, nonparametric statistics generally suffer little loss in power in removing these assumptions. Because of their more general nature, nonparametric statistics can handle a diverse array of distributions of data; in most arenas, nonparametric statistics are synonymous with distribution-free statistics. When experimental conditions depart substantially from the basic assumptions underlying parametric tests, such as normally distributed noise, nonparametric statistics can offer much more powerful and robust tests.

Nonparametric statistics have rarely been used for active learning or experiment design, and never previously (to our knowledge) for automated optimization of noisy functions.

4. Detecting Minima

Because the function to be optimized will typically be unknown before optimization takes place, we wish to create a guiding model for minima which has simple and hopefully robust assumptions about the shape of the minimum. Therefore, we restrict ourselves to the following assumption: the outputs of points closer to the minimum are smaller than those of points farther away. Given a set of points and a candidate minimum x^* we assume that, locally

$$f(x) = g(||x - x^*||) \qquad g\ monotonic \tag{3}$$

Since we do not specify the shape of the function g, the only parameter for a model k is x_k^*. Having few model parameters allows efficient use of the data to make inferences, thus requiring fewer experiments. The shape of the minimum can take any form so long as the function is monotonically increasing with respect to the distance from the minimum. Although this imposes a strong assumption of symmetry with respect to the distance metric $||\cdot||$, there is no requirement on $f(x)$ for continuousness or any prespecified noise distribution for $y(x)$. Another useful property of this model is immunity to all monotonic transformations of the data.

Observed data will fit this model of minima to varying degrees, so one would like to quantify the degree of fit. Given the assumption of Equation 3, we will use a nonparametric method to test the positive monotonic association between $||x - x^*||$ and $f(x)$. A typical nonparametric measure used for this purpose is *Kendall's correlation coefficient*, otherwise known as *Kendall's tau* (Kendall, 1938). We will denote the Kendall's tau for the relationship between $||x - x^*||$ and $y(x)$ in the data set D as $\tau_{x^*,D}$. In this paper, the subscript D will be omitted whenever possible.

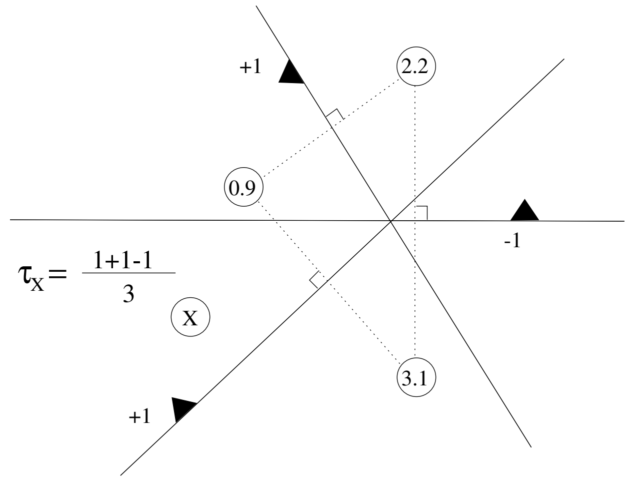


Figure 1. Example calculation of the statistic τ with two input dimensions. Each circle is a point in two-dimensional space, with the y value inside the circle. Solid lines are the perpendicular bisectors of each pair of points. For each pair's bisector, a solid triangle indicates the direction of the smaller of the two points. A +1 or -1 indicates whether the query point X is on the same side of the bisector as the smaller of the two points. The statistic τ_X is equivalent to the sum of these numbers divided by the number of bisectors.

An intuitive geometric interpretation of the appearance of Kendall's tau here is the following: if one creates a perpendicular bisector between a pair of points x_1 and x_2, our model of a minimum will be satisfied only if the minimum is on the same side of the bisector as the point with the smaller $y(x)$. One can imagine creating *n choose* 2 bisectors in space, one for each pair of points in the data set. The number of satisfied bisectors minus the number of unsatisfied bisectors, divided by the total number of bisectors, will have the same distribution as Kendall's tau. Figure 1 contains an example of our use of Kendall's tau and its calculation with three data points.

More generally, we wish to calculate τ_m for the model of Equation 3 with $x^* = m$ and data set D. To do this, we acquire the paired data, $||x_i - m||$ and y_i, and denote their ranks by r_i and s_i respectively. Next, arrange the r_i in ascending order (so that $r_i = i$.) A monotonic association between $||x - m||$ and $y(x)$ should cause the s_i to show either an increasing or decreasing trend. To measure this, we score each paired difference $s_j - s_i$ for $i = 1, 2, ..., n-1$ and $j > i$ as +1 if the difference is positive and as -1 if negative. Kendall called a positive difference a *concordance* and a negative difference a *discordance*. Denoting the sums of concordances and discordances by n_c and n_d respectively, the equation for our tau is

$$\tau_m = \frac{n_c - n_d}{\frac{1}{2}n(n-1)} \quad (4)$$

If the monotonic association is perfectly positive, $\tau_m = 1$, and perfectly negative association gives $\tau_m = -1$. If the rankings of $||x - m||$ and y are independent, we expect τ_m to be close to 0. Ties in the rankings pose no special difficulty; full explanation can be found in Sprent (1989).

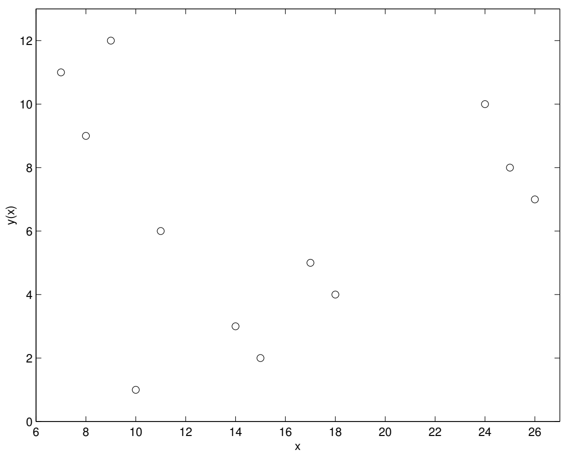


Figure 2. Example data set

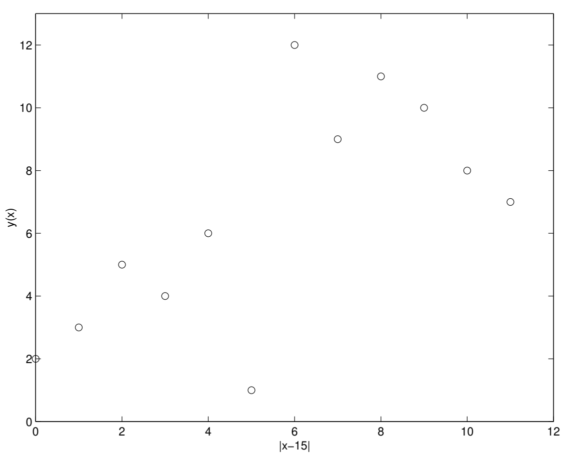


Figure 3. Data from Figure 2 plotting $|x - 15|$ against $y(x)$ in order to test the hypothesis that $x^* = 15$.

$\|x - 15\|$ rank	1	2	3	4	5	6	7	8	9	10	11	12
$y(x)$ rank	2	3	5	4	6	1	12	9	11	10	8	7

Table 1. Ranked data from Figure 3. The paired ranks are used to calculate τ for the hypothesis that $x^* = 15$ for the data in Figure 2.

For example, suppose that we have observed the data in Figure 2 in which there is one input x and one output $y(x) = f(x) + noise$. A possible model for a minimum might be Equation 3 with $x^* = 15$. In order to test this model's degree of fit, we will test for a positive monotonic association between $||x - x^*||$ and $f(x)$. This relationship is illustrated in Figure 3. Calculating $\tau_{m=15}$ and its significance for this model's fit to the data in Table 1 yields the following: $n_c = 10+9+7+7+6+6+0+2+0+0+0+0 = 47$, $n_d = 1+1+2+1+1+0+5+2+3+2+1+0 = 19$, and $\tau = (47-19)/66 = 0.4242$. A test for significance of the hypothesis $H_0 : \tau = 0$ against $H_1 : \tau > 0$, will require a precalculated table of critical values (found in Sprent (1989).) Such a table gives a threshold value of 0.3929 when $n = 12$ for 5% significance. Since $0.4242 > 0.3929$, we reject H_0 and accept a positive monotonic association between $|x - 15|$ and $f(x)$ with a confidence of 95%.

5. The PB1 Algorithm

The Pairwise Bisection (PB1) algorithm has two major phases. The first phase is *sample reduction*, in which the data set is narrowed to a subset of promising points. The second phase is *experiment selection*, which consists of choosing a point x from the area defined by the reduced sample generated in the first phase.

Phase 1: Sample Reduction

The purpose of sample reduction is to select a promising subset of the known datapoints. The reduced sample will ideally possess the following properties:

A. Covers a small enough section of the input space to eliminate the bias of the monotonic model (Equation 3)

B. Contains enough points to counter the effects of noise

C. Convexity

Not all these conditions can usually be satisfied at once, so sample reduction makes tradeoffs among them. The sample reduction phase used by PB1 consists of finding the reduced set of points, D_{red}. At the global level, the monotonic-minimum model is almost certainly biased. However, we will assume that it is still useful as a heuristic indicator of unpromising points. These unpromising points are gradually eliminated until a small set of points remains.

The sample reduction phase consists of the following three steps :

1. Include all n known points into sample set D_0.

2. For every point j in D_i, calculate τ_j. Generate D_{i+1} by removing the exterior [1] point in D_i with the smallest τ. Repeat this step for D_0 through D_{n-1}.

3. Decide which of these sample sets, D_0 through D_{n-1}, to use by choosing the *smallest sample set with a significant* τ. The determination of significance in this case is decided by applying a *two-tailed* hypothesis test to all points in D_i (i.e., one of the points' τ must be either significantly high or low.)

Intuitively, since the τ measure detects basin-like structures, steps 1 and 2 will tend to produce reduced samples that converge on local minima. When this is the case, always selecting the smallest reduced sample in step 3 will favor condition **A**, and the requirement of structural significance will enforce condition **B**. The final condition **C** is directly ensured by step 2, because points are only allowed to be removed from the "exterior" of the sample.

PHASE 2: EXPERIMENT SELECTION

The purpose of this phase is to select a point inside the region defined by the points in the reduced sample. The experiment selection phase consists of two steps:

1. Define the Region of Interest (ROI). The *ROI* is defined as the set of all points in the input space which meet the criteria given by Equations 5 and 6. For all points x in ROI,

$$\min_{y \in D_{red}} \|x-y\| \leq \min_{z \in (D-D_{red})} \|x-z\| \qquad (5)$$

This describes a Voronoi manifold from within which the experiment must be drawn. The other requirement is that all points in the ROI must also fit the data in the reduced sample, i.e., for all x in ROI,

$$\tau_{x,D_{red}} \geq \tau^{\alpha} \qquad (6)$$

Where τ^{α} is a critical value for τ that indicates a significance level of α (typically 0.05). This value only depends on the number of points in D_{red}, and is precalculated.

2. Choose g experiments uniformly randomly from within ROI. From these g points, with probability p choose the point with the best τ. With probability $(1-p)$ choose the point furthest away from all previously seen points.

The above procedure is currently implemented by a simple algorithm that randomly draws points from the full input space, then eliminates all points that do not meet the criteria of 5 and 6. For our experiments the somewhat arbitrarily chosen g and p values were 25 and 0.25, respectively. The values were selected prior to running the algorithm on the test functions. The performance of the algorithm does not seem sensitive to the values.

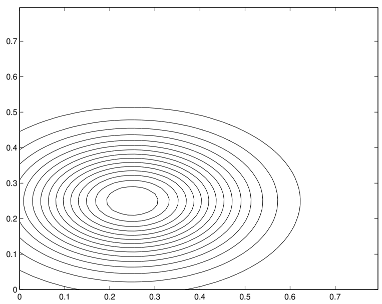
6. Results

We compare PB1 with four versions of Amoeba and three versions of PMAX. Amoeba is the classic search algorithm from Press et al. (1992). Amoeba2 is the same except it is made resistant to noise by doing two evaluations at each simplex vertex and taking their average. Amoeba4 and Amoeba8 similarly average four and eight evaluations at each vertex. All the Amoebas begin with a medium-sized simplex started randomly in input space. The three different versions of PMAX used locally weighted quadratic regression with different kernel widths, meaning different positions on the bias-variance tradeoff.

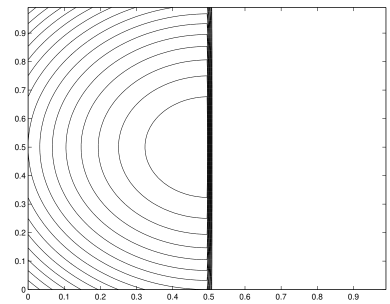
We use exactly the same experimental methodology as Moore et al. (1998). All algorithms were tested on the functions from Figure 4; Table 2 contains the results. In all experiments we performed 25 independent runs of each optimizer, with each run consisting of 60 experiments. In addition to selecting the datapoints for the experiments, at every stage the optimizers also gave their estimate of the location of the optimum, and so we look at the true value of the underlying function at these estimates of the optimum. For PB1, the estimate of the optimum was taken to be the point in the reduced sample with the highest τ value. For the ith run of a particular optimizer, let s_i denote the mean of the true values at the estimates of the optimum. The figures in the columns are the mean s_i value of the optimizer on the final 15 of the 60 experiments over all 25 runs (i.e. $(\sum_i s_i)/25$).

PB1 performs well on all the functions, although it performs especially well in domains which are noisy, contain outliers, or contain discontinuities. An initial suspicion was that the symmetric nature of the minimum model that PB1 uses would render the algorithm susceptible to "ridges" in the objective function; we

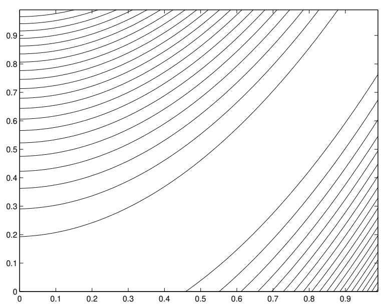
[1] An exterior point is defined here to be any point which is furthest from the centroid of the sample when all points are projected onto the line through it and the centroid. This is intended to be an approximation of the exterior points of the minimum-area convex polyhedron.



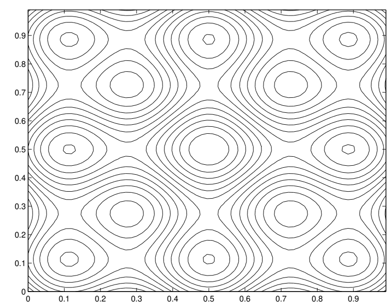
$$f_{gauss} = 1 - e^{(-20(x-0.25)^2+2(y-0.25)^2)}$$



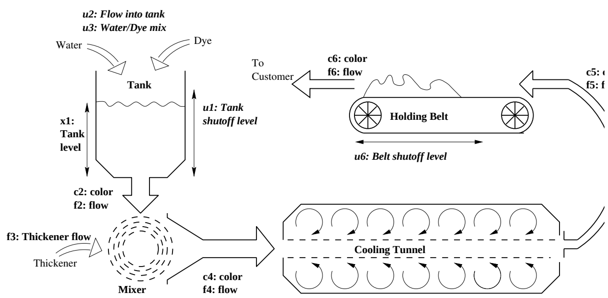
$$f_{discont} = \begin{cases} 1 - 2((x-0.5)^2 + (y-0.5)^2) & x < 0.5 \\ 0 & x \geq 0.5 \end{cases}$$



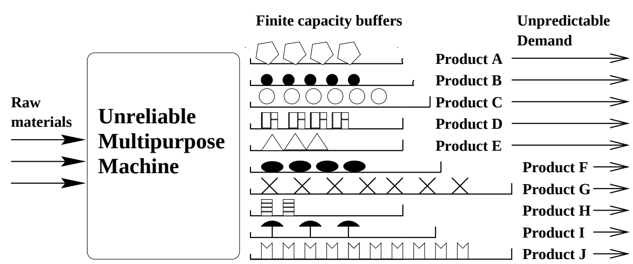
$$f_{Rosenbrock} = 10 - 100(y - x^2)^2 - (1-x)^2$$



$$f_{cosines} = 1 - (u^2 + v^2 - 0.3cos(3\pi u) - 0.3cos(3\pi v) + 0.7)$$
$$u = 1.6x - 0.5,\ v = 1.6y - 0.5$$



Chemical Plant Simulation (see text)



Multi-Buffer Simulation (see text)

Figure 4. The test functions used to evaluate performance. The task was to maximize the value of the test function.

investigate this by adding two irrelevant dimensions to the f_{guass} function. This created a long ridge in four-dimensional space. PB1 fared as well as or better than all other methods except Amoeba on this function. The $f_{Rosenbrock}$ function contained a more pronounced and difficult ridge, which was responsible for PB1's worst performance, although PB1 became competitive in the noisy case.

Figure 4 shows a simulated version of a real industrial chemical process described in Moore et al. (1998). The optimization task is to maximize the yield, which is determined by five parameters: reactant feed rate, mix-ratio, liquid level, thickener feed rate, and target inventory level. The yield is a very noisy and highly non-quadratic function; one input is almost irrelevant, the others are all important, and two of the inputs must run to their maximum legal value for best performance. The results are given in Table 2, and show that PB1 performs significantly better than all other methods tested.

Next, we examine a domain where experiments are time-consuming, a generalization of the multi-buffer machine task described in Mahadevan et al. (1997) (this makes ten products instead of five). There are two inputs defining a simple parameterized policy for when to service an unreliable machine. This task is evaluated by a computationally expensive simulation; each function evaluation requires 10000 simu-

Table 2. The functions used are those from Figure 4, with the goal of maxmization. Numbers in columns are mean score of last 15 experiments out of 60. The sample size is 25. 'Noise' is normally distributed with a standard deviation of 0.3. 'Outliers' indicates that with probability 0.1, noise will be an order of magnitude greater (standard deviation = 3.0). '2dims' indicates that two irrelevant input dimensions were added to the function. Significant ($p \leq 0.05$) results are denoted by an asterisk, and indicate that the algorithm tested significantly better than all other methods on that task. Due to the highly non-normal distribution of these means (bimodal in many cases) a nonparametric rank-sum test was used to determine significance.

	Amoeba 1	Amoeba 2	Amoeba 4	Amoeba 8	PMAX Global	PMAX Local	PMAX V.Local	Q2	PB1
f_{gauss}	**1.00**	0.99	0.80	0.61	0.99	0.99	**1.00**	**1.00**	**1.00**
f_{gauss}+Noise	0.36	0.52	0.52	0.45	0.42	0.57	0.54	0.69	**0.79***
f_{gauss}+Noise+Outliers	0.48	0.29	0.38	0.41	0.20	0.36	0.35	0.43	**0.72**
f_{gauss}+2dims	**0.99***	0.82	0.56	0.21	0.40	0.41	0.34	0.78	0.79
f_{gauss}+2dims+Noise	0.21	0.21	0.22	0.20	0.36	0.36	0.28	0.31	**0.60***
f_{gauss}+2dims+Noise+Outliers	0.17	0.14	0.13	0.16	0.20	0.25	0.19	0.18	**0.48***
$f_{discont}$	0.67	0.66	0.59	0.61	0.93	0.93	0.89	0.53	**1.00***
$f_{discont}$+Noise	0.76	0.81	0.78	0.76	0.86	**0.90**	0.78	0.86	**0.90**
$f_{discont}$+Noise+Outliers	0.71	0.73	0.70	0.74	0.65	0.71	0.65	0.71	**0.89***
$f_{Rosenbrock}$	**9.97***	9.75	8.94	5.40	8.96	9.23	9.43	9.71	9.59
$f_{Rosenbrock}$+Noise	9.60	9.58	8.93	5.42	8.89	9.28	8.93	**9.72***	9.46
$f_{Rosenbrock}$+Noise+Outliers	9.27	**9.56**	8.79	5.36	8.90	9.07	8.47	9.53	9.51
$f_{cosines}$	0.75	0.75	0.65	0.42	0.58	0.60	0.56	0.88	**0.90**
$f_{cosines}$+Noise	0.54	0.59	0.56	0.36	0.53	0.60	0.34	0.56	**0.79**
$f_{cosines}$+Noise+Outliers	0.41	0.39	0.38	0.34	0.33	0.34	0.21	0.29	**0.77***
Chemical Plant Simulation	26.39	30.04	25.24	17.90	40.33	39.52	24.80	42.79	**48.86***
Multi-Buffer Simulation	1.68	2.41	2.82	1.02	-1.38	-1.45	-1.52	3.06	**3.49**

lation steps. Evaluations are very stochastic (with highly non-Gaussian noise). The results are shown for runs of only 24 experiments. PB1 learns a good policy in these 24 experiments, i.e. a total of only 24×10000 simulation steps. This compares favorably with the tens of millions of simulation steps needed for reinforcement learning in Mahadevan et al. (1997).

7. Discussion

The ability of PB1 to choose points in the experiment selection phase is still being improved upon. The current method has difficulty in moving along some ridges, for example the ridge in the $f_{Rosenbrock}$ function. This difficulty may be a result of the model's inherent bias towards basin-like structures. Ways to address this, such as a different methods of experiment selection or manipulation of the distance metric, are being investigated.

Since pairwise bisection only requires distance measurements to evaluate points for their potential as minima, future work will apply it to optimization in hybrid continuous-discrete input spaces. The only aspect of the current algorithm that uses measurements other than distance is the determination of which points are "exterior" points. Ongoing work eliminates this dependency and allows PB1 to be used in any domain where a distance or similarity function can be defined, e.g., combinatorial optimization of Bayes net structure fits to data. The nonparametric nature of this algorithm already allows it to optimize in non-continuous output spaces.

8. Conclusion

We have developed a statistical method for characterizing optima, pairwise bisection, which uses a Kendall's tau distribution. A new optimization algorithm based on this nonparametric approach uses the statistic for sample reduction and experiment selection, and it seems to be competitive with existing algorithms. Pairwise bisection attempts to introduce discrete reasoning into optimization by assuming a set of binary constraints on the the minimum, namely, that the smaller member of every pair of data points is closer to a minimum. Nonparametric statistical reasoning decides whether a set of points indicates the presence of a minimum, and where to choose future experiments. This initial foray indicates a promising line of research for integrating nonparametric statistics into optimization and active learning. In future work we will apply these methods to learning parameters for robot control and other types of control systems.

References

Box, G. E. P., & Draper, N. R. (1987). *Empirical model-building and response surfaces.* Wiley.

Kaelbling, L. P. (1990). *Learning in embedded systems.* Doctoral dissertation, Department of Computer Science, Stanford University, Stanford, CA). Stanford University, Department of Computer Science.

Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika*, *30*, 81–93.

Kushner, H., & Clark, D. (1978). *Stochastic approximation methods for constrained and unconstrained systems.* Springer-Verlag.

Mahadevan, S., Marchalleck, N., Das, T., & Gosavi, A. (1997). Self-improving factory simulation using continuous-time average-reward reinforcement learning. *Proceedings of the Fourteenth International Conference on Machine Learning* (pp. 182–190). Morgan Kaufmann.

Moore, A. W., Hill, D. J., & Johnson, M. P. (1994). An empirical investigation of brute force to choose features, smoothers and function approximators. In S. Hanson, S. Judd and T. Petsche (Eds.), *Computational Learning Theory and Natural Learning Systems, Volume 3*, 361–379. MIT Press.

Moore, A. W., & Schneider, J. (1996). Memory-based stochastic optimization. *Neural Information Processing Systems 8* (pp. 1066–1072). MIT Press.

Moore, A. W., Schneider, J., Boyan, J., & Lee, M. S. (1998). Q2: A memory-based active learning algorithm for blackbox noisy optimization. *Proceedings of the Fifteenth International Conference on Machine Learning* (pp. 386–394). Morgan Kaufmann.

Press, W., Teukolsky, S., Vetterling, W., & Flannery, B. (1992). *Numerical recipes in C.* Cambridge University Press.

Robbins, H., & Monro, S. (1951). A stochastic approximation method. *Annals of Mathematical Statistics*, *22*, 400–407.

Salganicoff, M., & Ungar, L. H. (1995). Active exploration and learning in real-valued spaces using multi-armed bandit allocation indices. *Proceedings of the Twelfth International Conference on Machine Learning* (pp. 480–488). Morgan Kaufmann.

Sprent, P. (1989). *Applied non-parametric statistical methods.* Chapman and Hall.