Comparison study of pivot methods for global optimization

Pablo Serra,[a] Aaron F. Stanton, and Sabre Kais[b]
Department of Chemistry, Purdue University, West Lafayette, Indiana 47907

Richard E. Bleil
Kettering College of Medical Arts, 3737 Southern Boulevard, Kettering, Ohio 45429

(Received 17 October 1996; accepted 23 January 1997)

We compare two implementations of a new algorithm called the pivot method for the location of the global minimum of a multiple minima problem. The pivot method uses a series of randomly placed probes in phase space, moving the worst probes to be near better probes iteratively until the system converges. The original implementation, called the ''lowest energy pivot method,'' chooses the pivot probes with a probability based on the energy of the probe. The second approach, called the ''nearest neighbor pivot method,'' chooses the pivot probes to be the nearest neighbor points in the phase space. We examine the choice of distribution by comparing the efficiency of the methods for Gaussian versus generalized q-distribution, based on the Tsallis entropy in the relocation of the probes. The two implementations of the method are tested with a series of test functions and with several Lennard-Jones clusters of various sizes. It appears that the nearest neighbor pivot method using the generalized q-distribution is superior to previous methods. © *1997 American Institute of Physics.* [S0021-9606(97)01417-7]

I. INTRODUCTION

Many different fields of science require finding the location of the global minimum in a multiple minima function. A more efficient algorithm for global optimization would find applications in a wide range of fields, such as drug design, molecular modeling, mathematical biological calculations, and quantum mechanical calculations. One of the more difficult problems that frequently arises is that of molecular structure. Aside from the obvious difficulty of creation of a potential that accurately models the system in question, another serious problem lies in finding the minimas of this potential. One is presented with the dilemma that if one has a potential that can portray a system in a useful fashion, then that same potential is highly complex and difficult to minimize.[1–9] It is therefore a productive endeavor to investigate possible improvements to established methods of minimization of functions.

Methods of minimization can be largely classified into two groups, deterministic and stochastic. Deterministic methods have the strength of being extremely fast, but have the weakness of being liable to be caught in a local minimum fairly easily. Conversely, a stochastic method is far less likely to be trapped in a local minima, but it can be shown that no stochastic method has a probability of one to converge to the global minimum in a finite number of steps.

Among the deterministic methods known are variations on Newton's method, such as discrete Newton, quasi-Newton, and truncated Newton.[10] Additionally, work has been done to significantly improve these methods for very large systems, specifically limited memory quasi-Newton and truncated Newton approaches.[11] More recent work has focused also on deterministic global optimization, such as the tunneling method[12] and the renormalization group method.[13] At the same time, one can include simulated annealing,[14] quantum annealing,[15] J-walking,[16] tabu search,[17] and genetic algorithms[18] in the roster of stochastic methods.

Recently, we have developed two optimization methods, both based on pivot moves through phase space. In the original method,[19] the pivots were chosen based on their energies, while in a more efficient version,[20] the pivots were chosen as the nearest neighbor point. The major difference between the methods is the way in which the pivot points are chosen. In effect, phase space is visited in a very different way for the two methods.

To determine what circumstances favor one method over the other, we have run a series of test functions, as well as Lennard–Jones clusters, using both methods in order to better compare the relative strengths and weaknesses of the two approaches to the pivot method. In the next section we discuss the pivot method in general, the generalized distribution function, and its use in our methods. Section III goes into detail on the comparison via test functions. In Sec. IV we discuss the use of the nearest neighbor pivot method on Lennard-Jones clusters of sizes ranging from 6 to 20. Finally, we review the results and project further research to improve the method, as well as additional applications.

II. PIVOT METHODS

We begin by assuming a continuous phase space S. Within phase space is defined a real function, $f:S\rightarrow\Re$. Our goal is to determine the global minimum value of this function within the defined phase space, $\min f(\mathbf{x}):\mathbf{x}\in S$.

The general pivot approach to this problem is presented diagrammatically in the flowchart (Fig. 1). Briefly, we begin by locating a series of probes randomly distributed in the

[a]On leave from: Facultad de Matemática, Astronomía y Física, Universidad Nacional de Córdoba, Ciudad Universitaria, 5000 Córdoba, Argentina.
[b]Author to whom correspondence should be addressed.

 0021-9606/97/106(17)/7170/8/$10.00

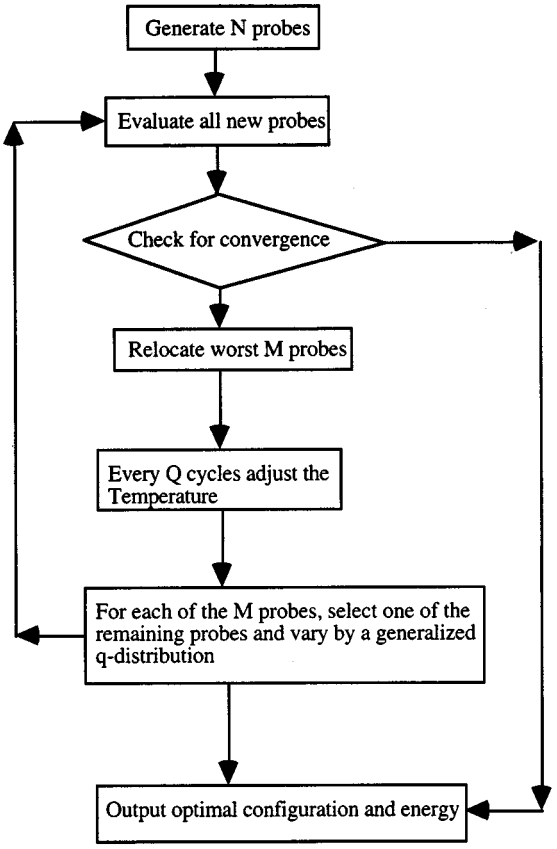

FIG. 1. Flow chart for pivot methods for global optimization. N is the total number of probes, M is the number of probes to be relocated, and Q is the number of cycles between temperature scalings.

phase space S, where each point in S is a series of values for all parameters of the problem. These points are called probes because they are probing the value of the function $f(\mathbf{x})$ for some set of parameters. The probes are chosen initially either completely at random or based on some given information about the problem, or some mixture of the two, by choosing some probes at random and some as our best guesses. Because the probes are themselves a complete set of values for all parameters in the problem, each probe would have some given value associated with it, that is, the value of the function at the point where the probe has been located. According to the ranking of the probes, we developed two different approaches: The lowest energy pivot (LEP) method and the nearest neighbor pivot (NNP) method. Both are based on the general idea of pivots, but they explore phase space in a very different way.

A. Lowest energy pivot method

For the lowest energy approach, we begin by ranking the probes from best (lowest function value) to worst (highest function value). If we have chosen enough probes, the chances are good that one or more of them are near the global minimum, and have good values. Probes near local minimas can also have good values, so it is important to avoid moving too many probes at once. Therefore, we choose some number of probes with poor values to be replaced, which is some fraction of the total number of probes with which we had started.

The number of probes we begin with and the number that we move at each iteration are arbitrarily adjustable parameters. The probability of choosing a probe close to the global minimum and low enough in the function well, is based on the number of probes we initially choose and the width of the function well at the global minimum relative to the size of phase space.

When we begin to move the probes with poor values, we want to place them near probes with better values in order to better explore the phase space in the region of the probes with good values. However, we don't want to place them all near the very best probe, because it is possible that any single probe is in a local, rather than global, minimum. Therefore, each probe that is moved will be placed near a probe chosen at random, but with a probability of being chosen as a pivot probe, based on its energy. The probe chosen as that probe which the relocated probe will be placed near we call the ''pivot probe,'' and the probability for each probe to be chosen as the pivot probe is given by

$$P_i=\frac{\exp[-f(\mathbf{x}_i)]}{P}, \quad P=\sum_{i=1}^{n-m} \exp[-f(\mathbf{x}_i)]. \tag{1}$$

We choose pivot probes at random because of the chance that we may be too close to a local minimum in the first iteration. We therefore assign a probability of choosing a particular probe, i, as a pivot probe, $f(i)$, where n is the total number of probes, m is the number of probes relocated at each iteration, and $f(\mathbf{x}_i)$ is the function value of probe i.

By choosing pivot probes according to this probability, we favor the probes with lower energy, but we also can choose probes with higher energy, thereby avoiding accidentally deleting the probe closest in phase space to the global minimum. In any given iteration, several pivot probes are chosen. Therefore, we are continuously moving into regions of phase space with lower function values.

Once a pivot probe is chosen, we reassign each parameter of the probe being moved to a value close to, but not equal to, the pivot probe. The relocated probe parameters are placed in a Gaussian distribution centered on the parameters of the pivot probe, with a standard deviation input as an arbitrarily adjustable parameter, σ. Initially, this size should be chosen to be large so that a large portion of phase space is covered by the pivot probes in the early iterations of the algorithm.

Once each relocated probe has been placed near a pivot probe, the values for the new probes are calculated, and the process begins again with the ranking of the values of the probes. For the LEP method, whether or not the relocated probe has a higher or lower energy than the pivot probe is not a consideration, because in the reranking any poorly placed probe will most likely be moved again in the next iteration anyway. Therefore, all pivot moves are accepted, regardless of the function value of the new probe relative to the pivot. The number of iterations for any given standard

deviation value can be varied. The more steps taken, the better phase space is sampled, but the longer it will take for the algorithm to converge. Once a predetermined number of iterations have elapsed using this standard deviation, this value is then decreased at some given rate.

This rate, R, decreases the standard deviation σ to some new value, σ', where $\sigma' = R\sigma$. At each new standard deviation, the algorithm repeats as many times as with the initial standard deviation, at which time σ is decreased again. In effect, this means that the probes will in time converge on some given small point, and, with a good sampling of phase space, this point should be the global minimum. The iterations continue until the stopping criteria are met.

B. Nearest neighbor pivot method

The nearest neighbor pivot (NNP) method[20] is another variant of the pivot method. In NNP we begin with a series of probes within the phase space of the problem, defined exactly as above. The method diverges from the lowest energy pivot method, however, in the way in which the pivot probes are chosen, and therefore, in the way in which phase space is explored.

In the NNP method, we start with $N = 2m$ initial probes, of which m probes will act as the pivot probes, and the remaining m probes will be relocated. A local selection of the m pivot probes begins with a search at each probe for its nearest neighbor, based on the distance of the probes. There are different ways to do this selection, but it is not relevant to the method. We began with finding the nearest neighbor for probe 1 and removed those two probes from further consideration. This was repeated until all points had been paired.

Once we have paired the probes, the probe with the lower value for the function $f(\mathbf{x})$ is defined as the pivot probe, the other probe being the probe that will be relocated.

For each pivot probe with parameter values $\mathbf{x}_{B,i}$, we explore phase space by placing the probe to be relocated near the pivot probe by changing its parameters $\mathbf{x}_{R,i}$ as

$$\mathbf{x}_{R,i} = \mathbf{x}_{B,i} + \Delta\mathbf{x}_i, \tag{2}$$

where $\Delta\mathbf{x}_i$ is a randomly generated vector according to a distribution $g_e(\Delta\mathbf{x})$. As opposed to the LEP method, which uses a Gaussian distribution, the NNP method works best with the generalized q-distribution based on the Tsallis entropy[21] for the placement of the probes near the pivot probes. This distribution was recently used with good results in generalized simulated annealing methods,[22–27] and is given by

$$g_q(x) = \sqrt{\frac{q-1}{\pi}}\,\frac{\Gamma[1/(q-1)]}{\Gamma[1/(q-1)-\frac{1}{2}]} \times \frac{[\beta(t)]^{1/(3-q)}}{[1+(q-1)([\beta(t)]^{1/(3-q)}x)^2]^{1/(q-1)}}, \tag{3}$$

where β is defined to be $1/T$ and T is an artificial temperature given by[23]

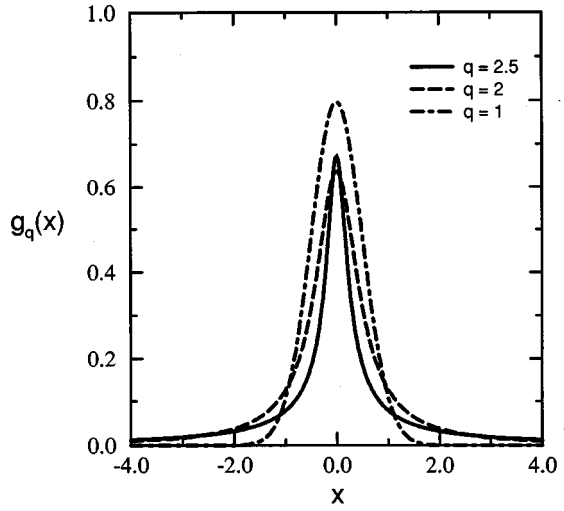

FIG. 2. Generalized q-distribution as a function of x for $q = 1$, $q = 2$, and $q = 2.5$.

$$T(t) = \frac{2^{q-1}-1}{(1+t)^{q-1}-1}\,T(1), \quad t = 1,2,3,\ldots, \tag{4}$$

where $T(1)$ is the initial temperature, and t is the discrete time corresponding to the computer iterations.[28]

The Tsallis q-distribution[23] defined above, Eq. (3), has been suggested as a generalization to the Gaussian distribution. This distribution function introduces a new parameter q as an arbitrarily adjustable parameter. This arbitrarily adjustable parameter q allows one to vary the length of the tail of the distribution curve. Special cases that should be denoted are the limit $q \to 1$, where the q-distribution approaches the Gaussian distribution, and for $q = 2$, where the q-distribution is equal to the Cauchy–Lorentz distribution used in fast simulated annealing.[29] The second moment of this distribution diverges for $q \geqslant 5/3$, and the distribution becomes not normalizable for $q \geqslant 3$. As we show in several test functions, we have found $q = 2.5$ to be a good value for our global optimization method, as in generalized simulated annealing.[23]

Figure 2 illustrates visually the differences between the Gaussian ($q = 1$), Cauchy ($q = 2$), and generalized $q = 2.5$ distributions. From this, one can easily see the transition from a localized distribution to a long tail one. Figures 3–5 demonstrate a two-dimensional random walk taken by each of these three distributions. Again, one can see that for a Gaussian distribution the majority of the walk is confined to one region of phase space, the Cauchy distribution occasionally has large ''jumps'' from one region to another, yet spends a large amount of time focusing on specific areas, and the $q = 2.5$ distribution has a high nonlocal character. Such a high nonlocal character insures that all of phase space is being adequately searched on a continuing basis, even at a low temperature.

We tested two ''pivot acceptance criteria.'' The first is the Metropolis algorithm,[30] where the acceptance probability $P_A(\mathbf{x}_t \to \mathbf{x}_{t+1})$ is given by

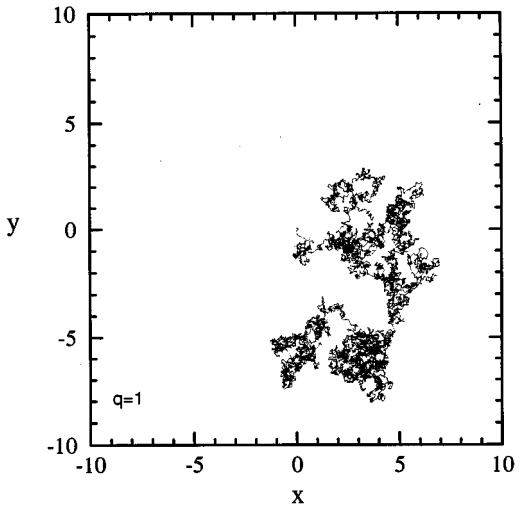

FIG. 3. Two-dimensional random walk with a Gaussian distribution ($q = 1$) and a temperature of 0.01.

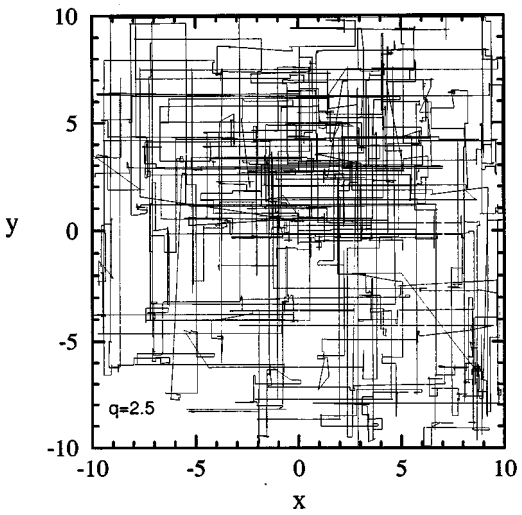

FIG. 5. Two-dimensional random walk with a q-distribution ($q = 2.5$) and a temperature of 0.01.

$$P_A(\mathbf{x}_t \rightarrow \mathbf{x}_{t+1}) = \begin{cases} 1 & \text{if } f(\mathbf{x}_{t+1}) < f(\mathbf{x}_t) \\ e^{-\beta(t)[f(\mathbf{x}_{t+1}) - f(\mathbf{x}_t)]} & \text{if } f(\mathbf{x}_{t+1}) \geqslant f(\mathbf{x}_t), \end{cases} \tag{5}$$

where β is a fictitious inverse temperature.

The second criterion used is an absolute acceptance criterion, which is identical to the Metropolis algorithm Eq. (5) in the limit $T \rightarrow 0$

$$P_A(\mathbf{x}_t \rightarrow \mathbf{x}_{t+1}) = \begin{cases} 1 & \text{if } f(\mathbf{x}_{t+1}) < f(\mathbf{x}_t), \\ 0 & \text{if } f(\mathbf{x}_{t+1}) \geqslant f(\mathbf{x}_t). \end{cases} \tag{6}$$

It has been determined that there is no significant difference between these two move acceptance criteria, and we use the absolute criterion Eq. (6) in the present work.

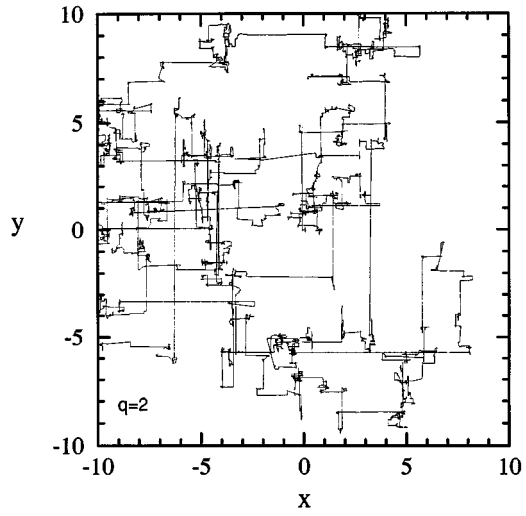

FIG. 4. Two-dimensional random walk with a Lorentz distribution ($q = 2$) and a temperature of 0.01.

To empirically determine which of the various possible values of q might work best for our optimization method, we must explicitly test the various distributions against an established set of test functions.

III. TEST FUNCTIONS

As a preliminary test for the use of the generalized q-distribution, we begin by trying to find the optimal value of q that minimized the number of function calls necessary in the minimization procedure. We have used the Tsallis distribution for all test functions. We have compared the two approaches (LEP and NNP) with one another, both with the Gaussian distribution function as well as NNP with the generalized q-distribution function. The goal is to determine the optimal value of q for each method and to demonstrate how the two approaches compare with one another. To do so, we have run each approach on several well-established functions. These functions included the Goldstein–Price (GP) equation, the Branin (BR), the Hartman three- and six-dimensional functions (H3 and H6), and the Shubert function (SH).[31] Each of these functions is explicitly stated in the Appendix. All of these functions were tested using both the LEP and NNP methods, and each method was tested using the Gaussian distribution and the NNP with generalized q-distribution for the relocation of the probes that are moved. To compare these in a machine-independent fashion, we report the number of function calls for both the Gaussian and the q-distribution function of the NNP approach, and the Gaussian distribution function of the LEP method in Table I. For comparison, we included in Table I the results of pure random search (PRS), simulated annealing types 1 and 2 (SA1 and SA2, respectively) and tabu search (these results are taken directly from Ref. 17). Several comments should be made about these results.

The number of function calls for the LEP approach are different than reported in an earlier publication.[19] In the pre-

TABLE I. Average number of function evaluations in the global optimization of five test functions.

Method[a]	GP	BR	H3	H6	SH
PRS	5,125	4,850	5,280	18,090	6,700
SA1	5,439	2,700	3,416	3,975	241,215
SA2	563	505	1,459	4,648	780
TS	486	492	508	2,845	727
LEP (Gaussian)	112	144	122	1536	281
NNP (Gaussian)	575	358	60	411	360
NNP ($q = 2.5$)	153	68	52	237	114

[a]The methods are the pure random search (PRS), simulated annealing types 1 and 2 (SA1 and SA2, respectively), tabu search (TS), lowest energy pivot (LEP), and nearest neighbor pivot (NNP). The references for these methods and results can be found in Ref. 17.

vious publication, we tried to keep all parameters of the problem identical for all test functions. In the current publication, we have used four of these parameters (number of initial probes, number of probes moved, number of search iterations per standard deviation, σ, and rate of contraction, R) as free parameters. These parameters were varied to minimize the number of function calls for each of the test functions. As a specific example, for the GP function, the best values were found to be 15 initial probes, 5 probes move at each step, 4 steps were taken at any given value of σ, and a rate of contraction of 0.385. It can be clearly seen from Table I that the LEP method, when used with our optimal parameters, yields a general improvement of a factor of approximately 4 over previous methods. This represents the bare method. As we shall see, when modifications such as the use of nearest neighbor selection or a generalized q-distribution are added, additional improvements are gained.

Similarly, the parameters for the NNP approach have been optimized as well. We are therefore comparing the best results in Table I. Initially we used a Gaussian distribution with the nearest neighbor pivot variation, and improvement was only seen for the H3 and H6 methods, and this suggested to us that perhaps varying the distribution may yield some more improvement. Tests were done to determine what the best value of q might be for various functions, and the results of this can be seen in Figs. 6–8. One can readily see that a value of roughly $q = 2.5$ yields the lowest number of function calls for nearly all of the test functions. This appears to be the best ''compromise'' between searching locally and nonlocally. The results of these test functions indicate that the NNP method is superior over previous methods, with a four- to 12-fold improvement over the tabu search method,[17] for all functions tested.

IV. LENNARD-JONES CLUSTERS

Lennard-Jones clusters are excellent for testing the efficiency of global optimization algorithms. Homogeneous Lennard-Jones clusters have well-established minimas and regular minimum-energy structures for very large clusters.[32,33] However, the number of local minima apparently grows as $\exp(N^2)$ (Ref. 34). Wilke and Vennik have shown that to find the global minimum in Lennard-Jones clusters is a NP-hard problem.[35] Several global optimization methods have been applied to the energy function of Lennard-Jones clusters. These include simulated annealing,[36] genetic algorithm,[37] diffusion equation methods,[38] quantum annealing,[39] J-walking,[40] and others.[41] The total energy for a Lennard-Jones cluster of N particles is

$$E_N = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} V_{\mathrm{LJ}}(r_{ij}), \tag{7}$$

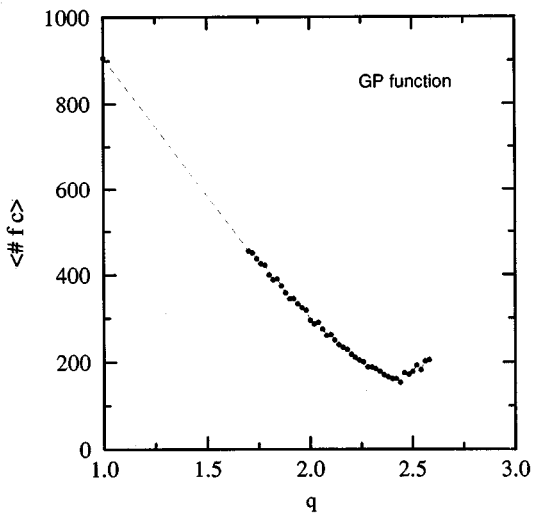

FIG. 6. Average number of function calls $\langle \#fc \rangle$ vs q for the Goldstein–Price (GP) function. This represents the average over the successful runs with a minimum convergence of 95%. One thousand runs were done for each value of q plotted.

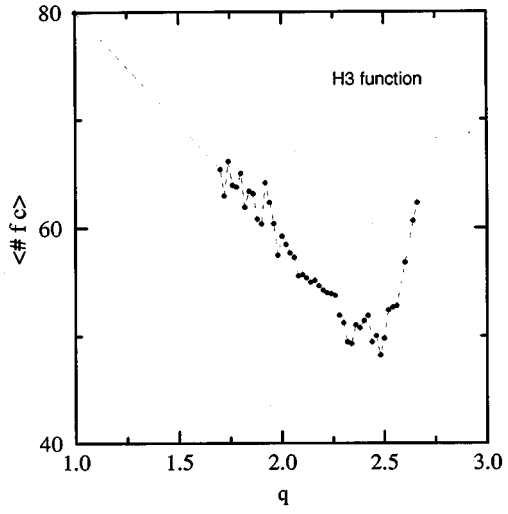

FIG. 7. Average number of function calls $\langle \#fc \rangle$ vs q for the three-dimensional Hartman (H_3) function. This represents the average over the successful runs with a minimum convergence of 95%. One thousand runs were done for each value of q plotted.

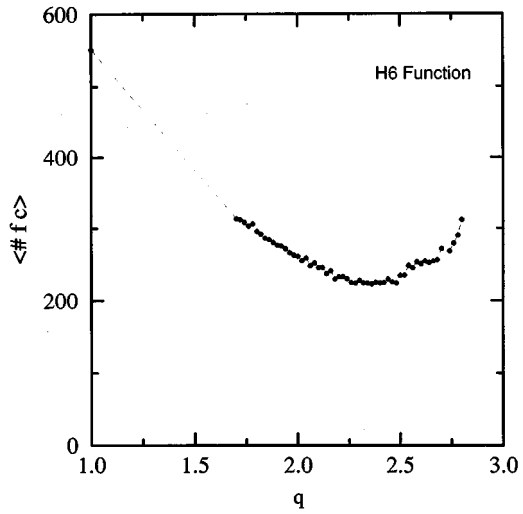

FIG. 8. Average number of function calls $\langle \#fc \rangle$ vs q for the six-dimensional Hartman (H_6) function. This represents the average over the successful runs with a minimum convergence of 95%. One thousand runs were done for each value of q plotted.

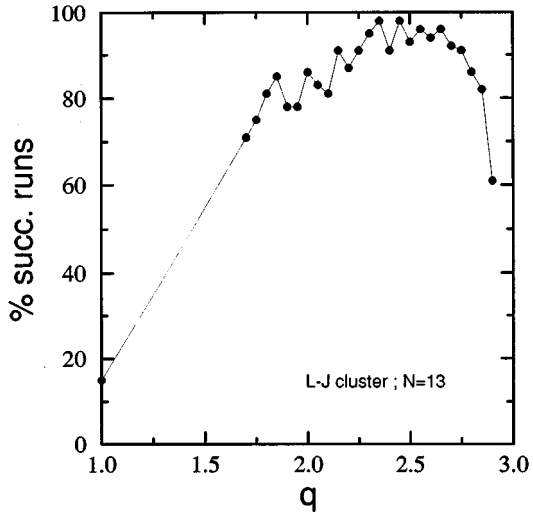

FIG. 9. Percentage of successful runs vs q for $N = 13$ Lennard-Jones function. One hundred runs were done for each value of q plotted.

where r_{ij} is the distance between the ith and the jth particles and $V_{\mathrm{LJ}}(r)$ is the Lennard-Jones two-body potential

$$V_{\mathrm{LJ}}(r) = \frac{1}{r^{12}} - \frac{2}{r^{6}}. \tag{8}$$

For small Lennard-Jones clusters ($N \leqslant 6$), the global energy minimum was located very quickly (less than 1 CPU second on an IBM RS/6000). One of the powerful features of this algorithm is that previous knowledge of the system can be built into the initialization of the probes. For larger clusters, we incorporated the information already gained by starting with the structure of the smaller $(N - k)$ clusters and adding k additional particles at random. In any ''growing'' problem, such as minimum energy configuration of clusters, self-avoiding walks, protein folding, etc.,[32] this systematic approach to solving the structure of large clusters can be incorporated. For an N-cluster, we begin with $m \times k$ initial pivot probes chosen as follows: There are m clusters of size $(N-1)+1$ random atom, m clusters of size $(N-2)+2$ random atoms,..., m clusters of size $(N-k+1)+(k-1)$ random atoms, and finally m completely random pivot probes (in our calculations we set $k = N/2$). With this set of the initial pivot probes, if the N-cluster has a similar structure with a smaller cluster, the algorithm converges faster than purely random initial points. If the N-cluster has a much different structure than the $(N - k)$ structure, or has one or more local minima near the global minimum, as in the $N = 18$ Lennard-Jones cluster, then the method works no less efficiently than it would with initial pivot probe locations chosen completely at random. In order to reach the exact minimum for the Lennard-Jones clusters, a gradient descent minimization was used once our method met its convergence criteria.

As was mentioned, we tested varying values of q to make certain that $q = 2.5$ was still a good value for Lennard-Jones clusters of varying sizes. Figure 9 represents the rate of convergence for different values of q for a cluster of 13 particles, and it can be seen that this convergence reaches a maximum at or very near to $q = 2.5$. Each point plotted represents an average over 100 runs. Figure 10 is a similar illustration for 18 particles in a Lennard-Jones cluster, and again it can be seen that there is a maximal value near $q = 2.5$. This indicates fairly strongly that a value of q near 2.5 seems to be a somewhat universal property of the generalized distribution function for our method of optimization.

Figure 11 illustrates how our method scales with the number of Lennard-Jones particles to be minimized. Using a log–log scale we show that our method scales approximately as $N^{2.9}$, compared to the recently reported modified genetic

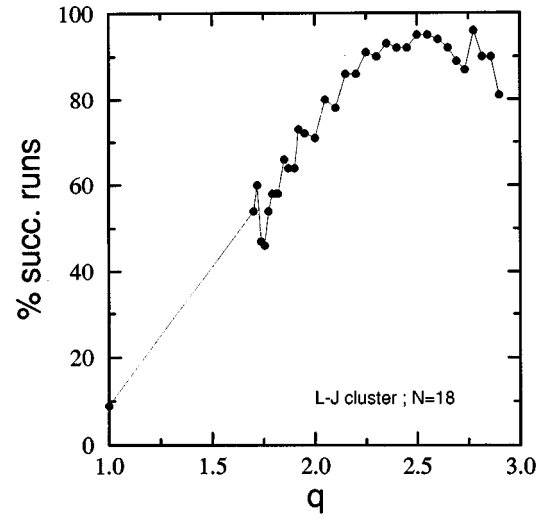

FIG. 10. Percentage of successful runs vs q for $N = 18$ Lennard-Jones function. One hundred runs were done for each value of q plotted.

TABLE II. Parameters for the Hartman function, H_n, for $n = 3$.

i	α_{i1}	α_{i2}	α_{i3}	c_i	p_{i1}	p_{i2}	p_{i3}
1	3	10	30	1	0.3689	0.1170	0.2673
2	0.1	10	35	1.2	0.4699	0.4387	0.7470
3	3	10	30	3	0.1091	0.8732	0.5547
4	0.1	10	35	3.2	0.03815	0.5743	0.8828

TABLE III. Parameters for the Hartman function, H_n, for $n = 6$.

i	α_{i1}	α_{i2}	α_{i3}	α_{i4}	α_{i5}	α_{i6}	c_i
1	10	3	17	3.5	1.7	8	1
2	0.05	10	17	0.1	8	14	1.2
3	3	3.5	1.7	10	17	8	3
4	17	8	0.05	10	0.1	14	3.2

i	p_{i1}	p_{i2}	p_{i3}	p_{i4}	p_{i5}	p_{i6}
1	0.1312	0.1696	0.5569	0.0124	0.8283	0.5886
2	0.2329	0.4135	0.8307	0.3736	0.1004	0.9991
3	0.2348	0.1451	0.3522	0.2883	0.3047	0.6650
4	0.4047	0.8828	0.8732	0.5743	0.1091	0.0381

algorithm, which scales as $N^{4.7}$.[37] One should note that the probability of finding the global energy minimum was strongly correlated with the size of the cluster. It was relatively difficult to obtain the global minimum for clusters $N = 6$ and $N = 18$. This implies that the difficulty (cpu time) does not scale in a simple way with N, but depends on the characteristics of the potential energy hypersurface.[42] Our cpu time given includes this gradient minimization. The cpu time on this chart has been experimentally determined on an IBM RS/6000-580 for our method, and the genetic algorithm[37] was included for comparison of scaling. The exact cpu time was not determined. Clearly, from this example one can see that substantial savings in time should be realized for using the NNP method over the genetic algorithm for very large clusters.

V. DISCUSSION

In this paper we have presented two different pivot methods and compared them, the lowest energy pivot method (LEP) and the nearest neighbor method (NNP). Using a Gaussian distribution it appears that the lowest energy pivot is good for low dimensional functions, while for higher order potentials the nearest neighbor method shows its strength. Once a generalized q-distribution is used in place of a Gaussian distribution, the nearest neighbor pivot method shows even more substantial improvement, especially for the high-dimensional functions. Having tested this with a range of values for q indicates that the value of q approximately equal to 2.5 gives the best results for this method. We concluded that when using the generalized q-distribution the NNP method is superior to previous methods.

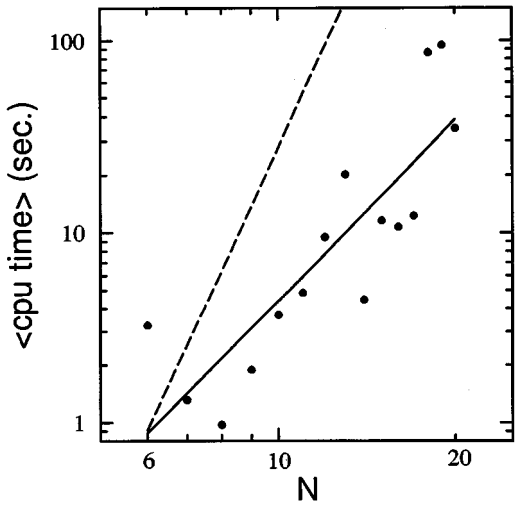

FIG. 11. Log–log graph of cpu time in seconds vs number of Lennard-Jones particles. The dashed line represents the scaling of the genetic algorithm of Ref. 37, $N^{4.7}$. The points represent experimentally determined times for each of the clusters, and the solid line represents the scaling of this method, $N^{2.9}$.

As an explicit demonstration of the improvements our method can yield over previous methods, Table I shows that for the simple LEP method using a Gaussian distribution, one can obtain roughly a fourfold improvement for most potentials. More interesting than that, however, is the roughly 12-fold improvement one sees by comparing the NNP method with a q-distribution using $q = 2.5$.

Continuing in our exploration of the pivot methods for optimizing functions, we have investigated the minimization of Lennard-Jones clusters of particles and found our method to be highly satisfactory. Figure 11 illustrates this by comparing our method to a modified genetic algorithm.[37] In this reference, a cpu-time scaling as $N^{4.7}$ was reported for Lennard-Jones clusters. We have shown that our method scales as $N^{2.9}$ for the same clusters using the same initialization and convergence criteria. A difference such as this does not make a large savings in time for small clusters, but when one wishes to optimize a very large system, this scaling makes a big difference in computer time.

In short, we have presented a new method of optimizing functions that has proven itself to be quite efficient, very flexible, rarely gets trapped in local minima, does not require computationally expensive derivatives, and is quite easy to implement. Flexibility is further enhanced by the ability to incorporate any previous knowledge of the potential under investigation into the optimization. As a specific example, we used smaller Lennard-Jones clusters as the starting point for the larger ones. For very large systems, one could use a crystalline structure as a starting point and optimize from there. Although there are established methods[32] for large Lennard-Jones clusters, they require a homogeneous regular structure as a basis, and our method has no such restriction. We are currently investigating a mathematical basis for this method, which should yield additional improvements to the method. We are also using this method for noble gas clusters, oxides and halides, polymer configurations, and small clusters of water molecules.

ACKNOWLEDGMENTS

P.S. would like to acknowledge C. Tsallis for stimulating discussions about applications of generalized statistical mechanics to global optimization, and D. Stariolo, who was kind enough to send us Ref. 23 prior to publication. S.K. is grateful to the donors of the Petroleum Research Fund, administered by the ACS, for support of this research.

APPENDIX

The test functions used in the text included the Goldstein–Price (GP) equation, given by[31]

$$f(x_1,x_2)=[1+(x_1+x_2+1)^2(19-14x_1+3x_1^2-14x_2+6x_1x_2+3x_2^2)][30+(2x_1-3x_2)^2(18-32x_1+12x_1^2+48x_2-36x_1x_2+27x_2^2)], \quad \text{(A1)}$$

where $-2 \leqslant x_i \leqslant 2$. The global minimum is equal to 3 and the minimum point is located at $(0,-1)$. There are four local minima in the minimization region.

The second test function is the Branin (BR) function,

$$f(x_1,x_2)=\left(x_2-\frac{5.1}{4\pi^2}x_1^2+\frac{5}{\pi}x_1-6\right)^2+10\left(1-\frac{1}{8\pi}\right)\times\cos x_1+10,\ -5\leqslant x_1\leqslant 10,\ 0\leqslant x_2\leqslant 15. \quad \text{(A2)}$$

The global minimum is approximately 0.398 and it is reached at three points $(-3.142, 12.275)$, $(3.142, 2.275)$, and $(9.425, 2.425)$.

The H_n Hartman test function has dimensionalities of 3 and 6 ($n = 3,6$) and is given as

$$f(x_i)=-\sum_{i=1}^{4} c_i \exp\left[-\sum_{j=1}^{n}\alpha_{ij}(x_j-p_{ij})^2\right], \quad 0\leqslant x_i\leqslant 1, \quad \text{(A3)}$$

where the parameters, α_{ij}, p_{ij}, and c_i are given in Tables II and Tables III. For $n = 3$ the global minimum is equal to -3.86 and it is reached at the point (0.114, 0.556, 0.852). For $n = 6$, the minimum is -3.32 at the point (0.201, 0.150, 0.477, 0.275, 0.311, 0.657).

The last test function is the Shubert (SH) function,

$$f(x_1,x_2)=\left[\sum_{i=1}^{i=5} i\cos[(i+1)x_1+i]\right]\times\left[\sum_{i=1}^{i=5} i\cos[(i+1)x_2+i]\right], -10\leqslant x_i\leqslant 10, \quad \text{(A4)}$$

and has 760 local minima in this region, 18 of which are global with $f(x_1,x_2) = -186.7309$.

[1] *Reviews in Computational Chemistry* edited by K. B. Lipkowitz and D. B. Boyd, (VCH, New York, 1992), Vol. III.
[2] L. C. W. Dixon and G. P. Szego, *Towards Global Optimization* (North–Holland, New York, 1975), Vols. 1 and 2.
[3] D. R. Herschbach, J. Avery, and O. Goscinski, *Dimensional Scaling in Chemical Physics* (Kluwer Academic, Dordrecht, 1993).
[4] T. L. Blundell, B. L. Sibanda, M. J. E. Sternberg, and J. M. Thornton, Nature **326**, 347 (1987).
[5] K. A. Dill, S. Bromberg, K. Yue, K. M. Fiebig, D. P. Yee, P. D. Thomas, and H. S. Chan, Protein Sci. **4**, 561 (1995).
[6] M. Oresic and D. Shalloway, J. Chem. Phys. **101**, 9844 (1994).
[7] L. Piela, J. Kostrowicki, and H. A. Scheraga, J. Phys. Chem. **93**, 3339 (1989).
[8] R. S. Berry, Int. J. Quant. Chem. **58**, 657 (1996); R. S. Berry, J. Jellinek, and G. Natanson, Phys. Rev. A **30**, 919 (1984); D. J. Wales and R. S. Berry, J. Chem. Phys. **92**, 4283 (1990).
[9] H. B. Schlegel, in *Ab Initio Methods in Quantum Chemistry-I*, edited by K. P. Lawley (Wiley, New York, 1987).
[10] J. E. Dennis and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations* (Prentice–Hall, Englewood Cliffs, NJ, 1983).
[11] J. C. Gilbert and C. Lemarechal, Math. Prog. **45**, 407 (1989); X. Zou, I. M. Navon, F. X. Le Dimet, A. Nouailler, and T. Schlick, SIAM J. Opt. **3**, 582 (1993).
[12] A. V. Levy and A. Montalvo, SIAM J. Sci. Stat. Comput. **6**, 15 (1985).
[13] D. Shalloway, J. Global Opt. **2**, 281 (1992).
[14] K. S. Kirkpatrick, C. D. Gellat, and M. P. Vecchi, Science **220**, 671 (1983); K. S. Kirkpatrick, J. Stat. Phys. **34**, 975 (1984).
[15] A. B. Finnila, M. A. Gomez, C. Sebenik, C. Stenson, and J. D. Doll, Chem. Phys. Lett. **219**, 343 (1994).
[16] D. D. Frantz, D. L. Freeman, and J. D. Doll, J. Chem. Phys. **93**, 2769 (1990).
[17] D. Cvijovic and J. Klinowski, Science **267**, 664 (1995).
[18] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning* (Addison Wesley, Reading, MA 1989).
[19] A. F. Stanton, R. E. Bleil, and S. Kais, J. Comp. Chem. **18**, 594 (1997).
[20] P. Serra, A. F. Stanton, and S. Kais, Phys. Rev. E **55**, 1162 (1997).
[21] C. Tsallis, J. Stat. Phys. **52**, 479 (1988).
[22] D. A. Stariolo and C. Tsallis, in *Annual Reviews of Computational Physics II*, edited by D. Stauffer (World Scientific, Singapore, 1995), p. 343.
[23] C. Tsallis and D. A. Stariolo, *Generalized Simulated Annealing*, Physica A (in press).
[24] T. J. P. Penna, Phys. Rev. E **51**, R1 (1995).
[25] I. Andricioaei and J. E. Straub, Phys. Rev. E **53**, R3055 (1996).
[26] J. Schulte, Phys. Rev. E **53**, R1348 (1996).
[27] K. C. Mundim and C. Tsallis, Int. J. Quantum Chem. **58**, 373 (1996).
[28] The FORTRAN code for the random number generator of the q-distribution is given in Ref. 23.
[29] H. Szu and R. Hartley, Phys. Lett. A **122**, 157 (1987).
[30] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, J. Chem. Phys. **21**, 1087 (1953).
[31] A. Torn and A. Zilniskas, *Global Optimization* (Princeton University Press, Princeton, NJ, 1991).
[32] J. Gu and B. Du, in *Global Minimization of Nonconvex Energy Functions: Molecular Conformation and Protein Folding*, edited by P. M. Pardalos, D. Shalloway, and G. Xue (AMS, 1996), DIMACS **23**, p. 65.
[33] R. S. Berry, T. L. Beck, H. L. Davis, and J. Jellinek, in *Advances in Chemical Physics*, edited by I. Prigogine and S. A. Rice (Wiley, New York, 1988), Vol. 70B, p. 75.
[34] M. R. Hoare, Adv. Chem. Phys. **40**, 49 (1979).
[35] L. T. Wille and J. Vennik, J. Phys. A 18, L419 (1985).
[36] L. T. Wille, Chem. Phys. Lett. **133**, 405 (1987).
[37] S. K. Gregurick, M. H. Alexander, and B. Hartke, J. Chem. Phys. **104**, 2684 (1996).
[38] J. Kostrowicki, L. Piela, B. J. Cherayil, and H. A. Scheraga, J. Phys. Chem. **95**, 4113 (1991).
[39] A. B. Finnila, M. A. Gomez, C. Sebenik, C. Stenson, and J. D. Doll, Chem. Phys. Lett. **219**, 343 (1994).
[40] D. D. Frantz, J. Chem. Phys. **102**, 3747 (1995).
[41] P. M. Pardalos and G. L. Xue, J. Global Opt. **4**, 117 (1994).
[42] J. Ma and J. E. Straub, J. Chem. Phys. **101**, 533 (1994).