Global Optimization of Lennard-Jones Clusters by a Parallel Fast Annealing Evolutionary Algorithm

Wensheng Cai, Haiyan Jiang, and Xueguang Shao*

Department of Chemistry, University of Science and Technology of China, Hefei 230026, Anhui, P. R. China

Received January 26, 2002

A parallel fast annealing evolutionary algorithm (PFAEA) was presented and applied to optimize Lennard-Jones (LJ) clusters. All the lowest known minima up to LJ_{116} with both icosahedral and nonicosahedral structure, including the truncated octahedron of LJ_{38}, central fcc tetrahedron of LJ_{98}, the Marks' decahedron of LJ_{75-77}, and $LJ_{102-104}$, were located successfully by the unbiased algorithm. PFAEA is a parallel version of fast annealing evolutionary algorithm (FAEA) that combines the aspect of population in genetic algorithm and annealing algorithm with a very fast annealing schedule. A master−slave paradigm is used to parallelize FAEA to improve the efficiency. The performance of PFAEA is studied, and the scaling of execution time with the cluster size is approximately cubic, which is important for larger scale energy minimization systems.

1. INTRODUCTION

Global optimization problem is a subject of intense current interest.[1] In chemistry and biochemistry, the knowledge of molecular structure is very important for scientists to understand the formation of macromolecular system and their reaction mechanism. The native configuration of a macromolecular system is structurally related to the global minimum of its potential energy surface (PES). The number of minima on the PES usually increases exponentially with the size of the system making it very hard to optimize a macromolecular system such as protein using a global optimization algorithm. For example, in much simpler systems the PES of Lennard-Jones (LJ) cluster with 98 atoms has been estimated to possess 10^{40} minima.

The LJ clusters have been investigated intensively, and they provide a useful testing ground for global optimization algorithms. The LJ potential is given by

$$E = 4\epsilon \sum_{1 \le i < j \le N} \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^{6} \right] \qquad (1)$$

where ϵ and $2^{1/6}\sigma$ are the pair well depth and the equilibrium pair separation, respectively. Many methods have been applied to the LJ problem such as simulated annealing (SA),[2−5] genetic algorithm (GA),[6,7] basin-hopping,[8,9] lattice methods,[10,11] smoothing and hyper-surface deformation techniques.[12] Most of the global minima for $N < 147$, where N is the number of atoms, were first found by Northby using a lattice-based search of icosahedral structures.[10] Whereas, unbiased algorithms that do not use specific candidates considering cluster geometry are of the most interest since these methods are superior in generalization to be applied to more complex potentials.

Most global minima for LJ clusters containing fewer than 100 atoms are based on icosahedral packing.[1] Several exceptional clusters, the truncated octahedron LJ_{38} and the Marks' decahedron LJ_{75-77}, are hurdles in the structural optimization of LJ clusters for unbiased algorithms.[1,8] At these sizes, the lowest energy minimum based on icosahedron acts as a trap and is widely separated from the true global minimum. Analysis of the PESs of LJ_{38} and LJ_{75} using disconnectivity graphs has shown that the icosahedral "funnel"[13] is wide, whereas the funnel leading to the global minima is much narrower.[14,15]

The two hurdles of LJ_{38} and LJ_{75} were first beat by Wales et al. using a pure unbiased algorithm named basin-hopping.[8] In Wales' work, a seeding technique is used to find the global minima of most icosahedral clusters and those decahedral clusters including LJ_{76-77} and $LJ_{103-104}$. This seeding technique is very effective because the start configuration is based on the similar structure of the cluster containing one more atom or one less atom. Wolf and Landman also successfully located the LJ global minima for LJ_2 to LJ_{110} using local minimization and a seeded genetic algorithm.[7]

Compared with the methods mentioned above, usual simulated annealing (SA) algorithm and more sophisticated variants of SA appeared to fail at small sizes.[2−4] Xue found new global minima for LJ_{65} and LJ_{66} using a two-level simulated annealing, which basically corresponds to the basin-hopping without resetting the coordinates to those of the current minimum.[5] In fast annealing evolutionary algorithm (FAEA), the method to generate a new solution, the annealing schedule, and the population made it more efficient than the usual SAs.[16] The comparison of FAEA with some other algorithms was given in ref 16.

In this study, a parallel fast annealing evolutionary algorithm (PFAEA) is proposed and applied to the LJ cluster problem. PFAEA is a parallel version of FAEA,[16−18] and a master−slave paradigm is implemented to parallelize the FAEA. Several additional techniques such as similarity checking, limited memory quasi-Newton algorithm (L-BFGS), and moving are applied.[17] L-BFGS is an optimization technique based on quasi-Newton method that was proposed by Broyden, Fletcher, Goldfard, and Shanno and was named for that. To illustrate the performance of the unbiased

* Corresponding author e-mail: xshao@ustc.edu.cn.

10.1021/ci020004i CCC: $22.00
Published on Web 08/10/2002

PFAEA, seeding technique was not used. The results showed that all the known lowest energy structures up to $N = 116$ have been located successfully. Furthermore, it is shown that the time complexity of PFAEA with cluster size is cubic. The same scaling is exhibited in the phenotype algorithm[19] and is regarded as a remarkable improvement over the exponential scaling of basin hopping.[1]

2. METHOD

2.1. Very Fast Annealing. A very fast annealing technique of Very Fast Simulated Reannealing (VFSR) proposed by Ingber[20,21] is adopted in PFAEA. In annealing-time step k, current temperature $T(k)$ is calculated according to eq 2

$$T(k) = T_0 \exp(-ck^{1/D})$$

$$c = m \exp(-n/D) \tag{2}$$

where T_0 is the initial temperature, D is the number of parameters to be optimized, and m and n are parameters to tune annealing schedule.

A new solution for an individual x^i_{k+1} is generated by eq 3 using the current temperature T:

$$x^i_{k+1} = x^i_k + y^i(B_i - A_i),\ i = 1, D \tag{3}$$

The random variable y^i is generated from u^i, a random number with uniform distribution, as follows:

$$u^i \in U[0, 1]$$

$$y^i = \mathrm{sgn}\left(u^i - \frac{1}{2}\right)T\left[\left(1 + \frac{1}{T}\right)^{|2u^i - 1|} - 1\right] \tag{4}$$

The annealing process will stop when the current temperature T reaches the threshold temperature T_{exit}.

$$T_{exit} = T_0 \exp(-m) \tag{5}$$

2.2. Parallelization of FAEA. The parallel fast annealing evolutionary algorithm is implemented by a coarse-grained master−slave paradigm. There are one master task and a number of slave tasks in the PFAEA. In general, each slave has a serial FAEA task, whereas the master controls the whole process. In this study, the master also has a serial FAEA task to make the sufficient use of CPU times.

In PFAEA, the population is divided to N_p (number of processors) subpopulations. Starting with a randomly generated subpopulation, master and slaves will run the serial fast annealing evolutionary procedure. After each generation, the slaves send the lowest potential of subpopulations to master. Based on the results received, master decides if the goal, i.e., the global minimum, has been reached. The master will broadcast a stop signal to all slaves if the global minimum has been found. Furthermore, the master will gather all the subpopulations from the slaves after a number of generations and check for similarity to avoid duplications among the subpopulations. After that, the subpopulations are sent back to the slaves. At the end of the annealing schedule, each processor selects the best individual of its subpopulation and optimizes it further. Finally, the master task will select the best individual as the optimal result from the best solutions of all tasks.

In the present work, PFAEA program is performed on a cluster of workstations (COW). The cluster is a distributed memory computer system containing eight computers with two Pentium III 866 CPUs each. Parallel code is written according to the Message Passing Interface (MPI) library, and the standard MPI software used here is MPICH-1.2.1.

2.3. Program Structure and Implementation. The flowchart of the PFAEA program is shown in Figure 1, which includes the following steps:

1. The algorithm starts with initializing the parameters, including the initial temperature T_0, the parameters to control the annealing schedule m and n in eq 2, the length of Markov chain *ns*, the population size *npop* for evolution, and the cluster size N.

2. Subpopulations are generated randomly and evaluated on master and slaves. Spherical coordinates are used for encoding the position of each atom by three values: r, θ, and φ. Therefore, each individual has $3N$ genes or parameters. The value of every gene is generated by $u(A_i - B_i)$, where $u \in U[0, 1]$ is a random number with uniform distribution, and A_i and B_i are the low and up limits of each parameter.

3. At each generation of the annealing schedule, the similarity checking is performed on master and slaves to keep the diversity of subpopulation. A generation means the population evolved by annealing technique at a temperature.

4. Every 20 generations, a limited memory BFGS (L-BFGS) procedure is applied on master and slaves to optimize each structure of subpopulations created in the annealing procedure to a local minimum. The L-BFGS code written by D. C. Liu and J. Nocedal[22] comes from Netlib (http://www.netlib.org) and was proven to converge very fast in low iteration cost.

5. After the local minimization, all subpopulations are sent to master by slaves for checking similarity of the whole population. All the slaves are idle when master checks the similarity. After that, subpopulations are sent back to slaves by master.

6. The subpopulations of master and slaves are evolved by a very fast annealing procedure. A new solution for each individual is generated according to the very fast annealing schedule, and the Metropolis criterion of simulated annealing is used for determining whether the new solution is acceptable or not. Then the subpopulations are replaced with the new generation.

7. After each generation, the lowest energy of subpopulation on each slave is sent to the master. All the known lowest minima (E_{lowest}) were stored in the program. If the known lowest minimum has been found ($E - E_{lowest} < 5 \times 10^{-7}$), the master will broadcast a signal to all slaves, and the annealing process will stop. Otherwise, the annealing process will continue until the annealing temperature T reaches T_{exit}.

8. At the end of PFAEA, an angular move of outer atoms is performed on the best individual of each subpopulation, and the L-BFGS method is used for further optimization. Finally, the best configurations of slaves are sent to the master, and the configuration with the lowest energy is selected and output as the final result.

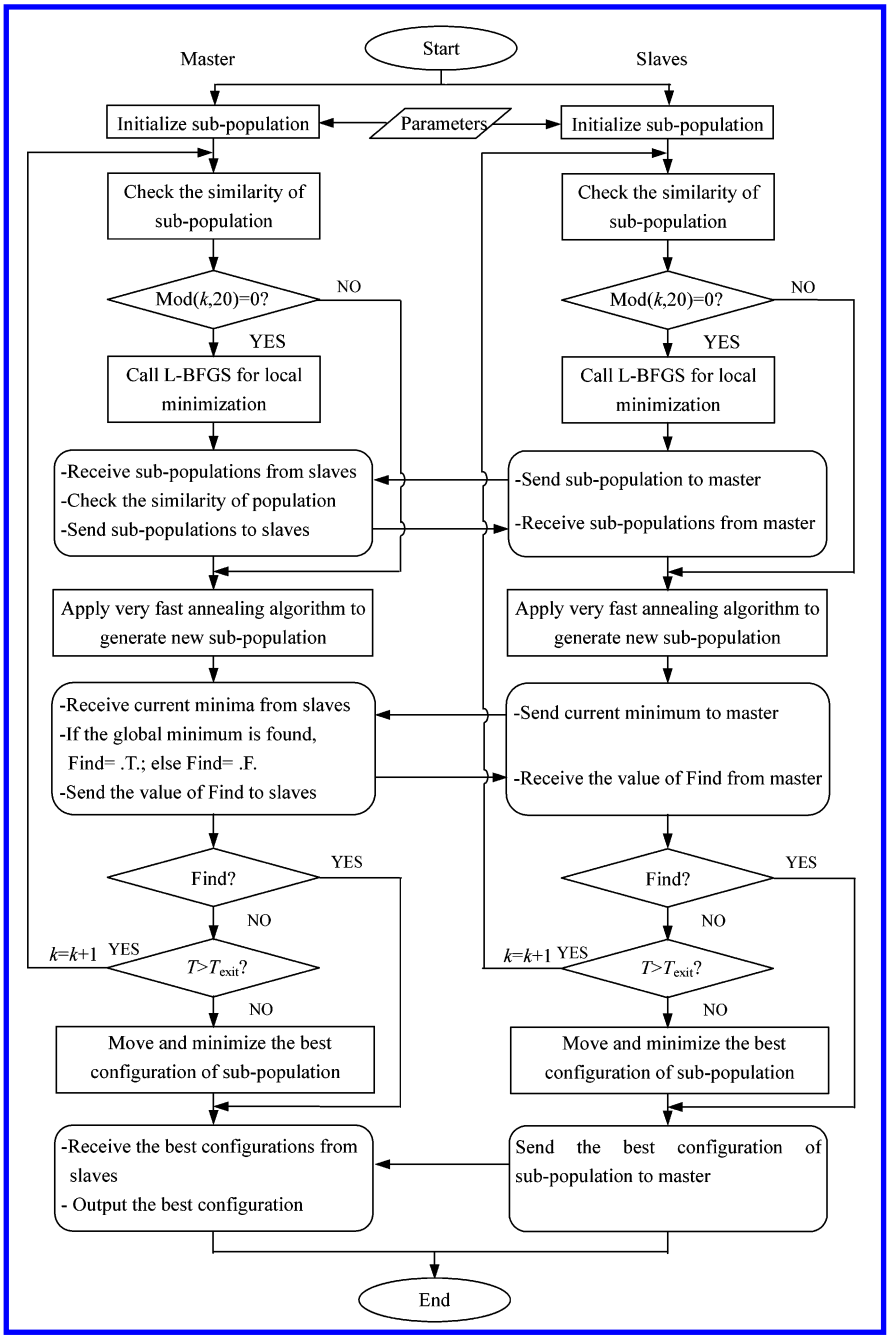

Figure 1. Flowchart of the PFAEA main program.

3. RESULTS AND DISCUSSION

3.1. The Optimization of LJ Clusters. The PFAEA has successfully located all the lowest known minima for systems up to LJ_{116} including all the well-known difficult cases, the global minima for LJ_{38}, LJ_{75-77}, $LJ_{102-104}$, and LJ_{98}. Each of these structures was found from the initial population with randomly generated coordinates. Seeding technique that has bias in the starting configuration has not been used in PFAEA. Therefore, it is a purely unbiased algorithm.

The optimization results are listed in Table 1. It is shown that most of the global minima of icosahedral clusters are located easily compared with the structures mentioned above, and the optimization of this kind of clusters has a high success ratio especially when the atom number is smaller than 75.

In this study, the tetrahedral global minimum of LJ_{98} is located successfully in 4 out of 100 runs by PFAEA. The new structure of LJ_{98} with the energy of -543.665361ϵ was first reported by Leary in 1999.[9] The result of present work is a great improvement because this new global minimum is regarded particularly difficult to find according to the former work of Leary in which the success ratio is only 6 out of 1000 by a variant of the basin-hopping algorithm.[9]

The four different configurations of LJ clusters with $N \leq 116$ are presented in Figure 2, including the icosahedron and all the famous nonicosahedral structures (the truncated octahedron of LJ_{38}, the Marks' decahedron of LJ_{75-77} and $LJ_{102-104}$, and the central fcc tetrahedron of LJ_{98}). From the motifs for the lowest energy structures listed in Table 1, it is clear that most of them are based on icosahedron, and as the size of clusters increases, the proportion of decahedral structures increases. To show the stability of the lowest energy clusters compared with their neighbors, the second finite differences of energy,[23] $\Delta_2 E(N) = E(N+1) +$

Table 1. Optimization of LJ_N for $N \leq 116$ by PFAEA

N	motif[a]	success/ total runs	N	motif[a]	success/ total runs	N	motif[a]	success/ total runs
38	(1)	39/100	65	(2)	3/10	92	(2)	1/10
39	(2)	9/10	66	(2)	8/10	93	(2)	2/10
40	(2)	10/10	67	(2)	5/10	94	(2)	2/10
41	(2)	10/10	68	(2)	8/10	95	(2)	2/10
42	(2)	10/10	69	(4)	8/10	96	(2)	1/10
43	(2)	10/10	70	(2)	8/10	97	(2)	1/10
44	(2)	10/10	71	(2)	7/10	98	(4)	4/100
45	(2)	10/10	72	(2)	7/10	99	(2)	2/10
46	(2)	9/10	73	(2)	10/10	100	(2)	1/10
47	(2)	10/10	74	(2)	8/10	101	(2)	2/10
48	(2)	10/10	75	(5)	2/200	102	(5)	9/100
49	(2)	10/10	76	(5)	2/50	103	(5)	3/30
50	(2)	9/10	77	(5)	1/50	104	(5)	2/30
51	(2)	10/10	78	(4)	2/10	105	(2)	1/10
52	(2)	10/10	79	(2)	5/10	106	(2)	2/5
53	(2)	10/10	80	(2)	6/10	107	(4)	3/30
54	(2)	10/10	81	(2)	7/10	108	(2)	4/20
55	(3)	10/10	82	(2)	2/10	109	(2)	3/20
56	(2)	10/10	83	(2)	2/10	110	(2)	1/20
57	(2)	10/10	84	(2)	1/10	111	(2)	1/10
58	(2)	10/10	85	(2)	2/10	112	(2)	1/20
59	(2)	10/10	86	(2)	3/10	113	(4)	1/20
60	(2)	10/10	87	(2)	2/10	114	(2)	3/20
61	(2)	10/10	88	(4)	5/10	115	(4)	1/20
62	(2)	10/10	89	(2)	2/10	116	(2)	1/20
63	(2)	10/10	90	(2)	4/10			
64	(2)	10/10	91	(2)	3/10			

[a] Structural motifs: (1) face-centered-cubic (fcc) truncated octahedron, (2) Mackay icosahedron at the core, (3) complete Mackay icosahedron, (4) icosahedral structure with incomplete core, and (5) Marks decahedral structure.

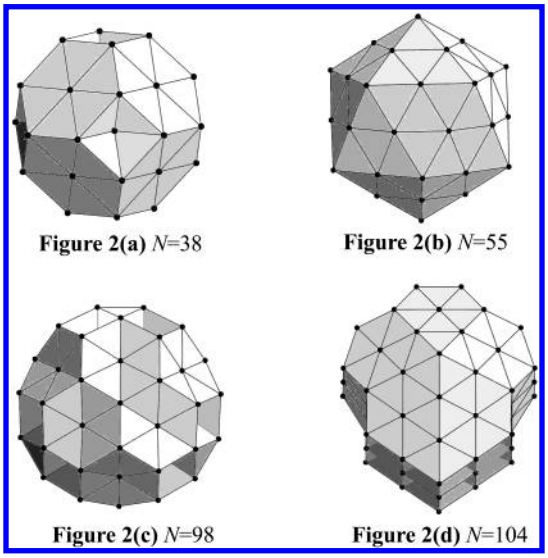

Figure 2. Global minima of the Lennard-Jones clusters with $N = 38, 55, 98, 104$ by PFAEA.

$E(N-1) - 2E(N)$, where $E(N)$ is the minimum energy of the cluster with N atoms obtained by PFAEA, were calculated and shown in Figure 3, in which particularly stable structures can be indicated by peaks. Obviously there are peaks at $N = 13$ and 55 corresponding to complete Mackay icosahedra.

3.2. Performance Study of PFAEA. The scalability of the PFAEA is studied thoroughly by optimizing LJ_{38}, LJ_{55}, and LJ_{75} using both serial FAEA and PFAEA, and the results are shown in Table 2. A serial FAEA can be regarded as the program running on one processor with whole population. It can be seen from the speedup in Table 2 that the parallel

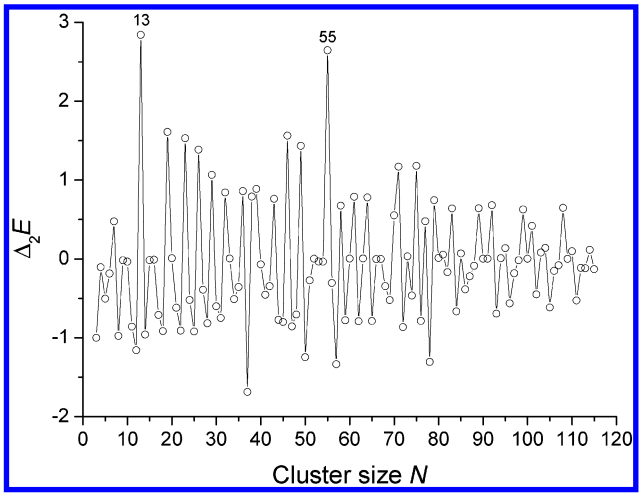

Figure 3. Plot the second finite differences, $\Delta_2 E(N) = E(N+1) + E(N-1) - 2E(N)$, versus the size of cluster N. The peaks correspond to the particularly stable structures.

Table 2. Performance of PFAEA[d]

N	version	no. of processors	T_{comm}[a]	T_{simi}[b]	T_{total}[c]	speedup
38	serial	1	0	76.13	354.22	
	parallel	2	2.65	28.41	178.32	1.99
	parallel	4	2.80	15.95	103.99	3.41
	parallel	8	3.13	12.74	71.55	4.95
	parallel	12	2.92	12.03	58.81	6.02
	parallel	16	2.98	1.72	41.03	8.63
55	serial	1	0	397.35	1035.8	
	parallel	2	6.23	114.21	458.65	2.26
	parallel	4	5.71	43.57	259.37	3.99
	parallel	8	5.44	24.85	169.89	6.09
	parallel	12	4.88	21.07	146.59	7.06
	parallel	16	4.76	19.70	118.66	8.72
75	serial	1	0	1990.36	3297.15	
	parallel	2	10.21	510.93	1266.22	2.60
	parallel	4	9.55	146.64	663.17	4.97
	parallel	8	9.04	55.24	449.19	7.34
	parallel	12	7.646	38.42	377.48	8.73
	parallel	16	5.21	32.62	336.38	9.81

[a] The communication time between master and slaves. [b] The time for checking similarity. [c] The total time required. [d] All data of time are the average values over five runs. Parameters used are *npop*: 240, total generation: 149 ($n = 5$), *ns*:1.

program has a good scalability, especially when the number of processors was lower than 8 for the larger problem size LJ_{55} and LJ_{75}. For $N = 38, 55, 75$, the communication time takes 2−4% of the total CPU time when $N_p = 8$, and the time for checking similarity of the whole population, during which slaves are idle (see step 5 of section 2.3), takes approximately 5%.

It is notable that the time for similarity checking decreases sharply while the number of processor increases. There are two reasons for that. One is the good diversity of subpopulation generated on different nodes. The second is the small size of subpopulation distributed by the parallel algorithm. The sharply decreasing of the time for checking similarity results in the near-liner or super-liner speedups shown in Table 2 for $N_p = 2, 4, 8$.

Furthermore, the complexity of PFAEA has been studied, and the results are shown in Figure 4. The PFAEA program was run with eight processors and $ns = 4$ that were the values in practical optimization; other parameters were the same

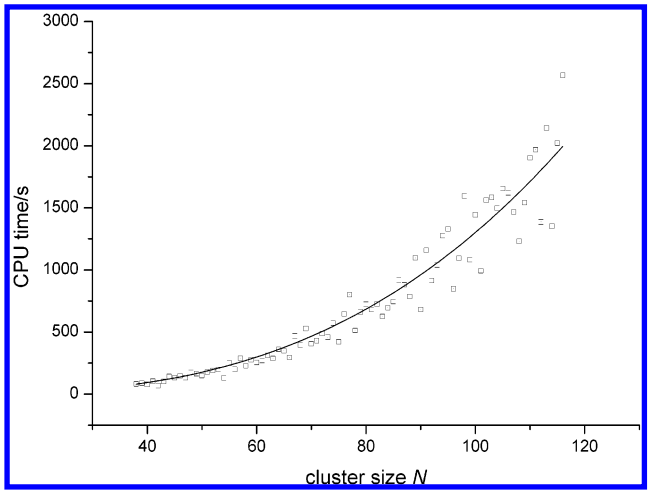

Figure 4. The scaling of CPU time with cluster size ($38 \leq N \leq 116$) by PFAEA. (Parameters used are *npop*: 240, N_p: 8, total generation: 149, and *ns*: 1).

as in Table 2. The data were analyzed by the nonlinear curve fitting with the functional form $f(N) = a \times N^b$. As a result, the curve $0.02 \times N^{2.9}$ is obtained and displayed in Figure 4. Consequently, the scaling of the execution time with cluster size N is approximately cubic.

4. CONCLUSIONS

A parallel fast annealing evolutionary algorithm is presented and applied to global optimization of Lennard-Jones atomic clusters up to 116 atoms. All the known lowest minima were located successfully, including the eight nonicosahedral structures, i.e., the truncated octahedron of LJ_{38}, the central fcc tetrahedron of LJ_{98}, and the Marks' decahedron of LJ_{75-77} and $LJ_{102-104}$, which are considered to be very difficult to be optimized. The results indicated that as an unbiased algorithm, PFAEA could be very efficient for the optimization of atomic or molecular clusters. Additionally, the complexity of PFAEA is in cubic with the cluster size N, which is the best among the algorithms published. As a result, PFAEA may have the potential to be applied to large-scale energy minimization problem.

ACKNOWLEDGMENT

This study is supported by the National Natural Science Foundation of China (No. 29975027 and No. 20172048) and the Teaching and Research Award Program for Outstanding Young Teachers (TRAPOYT) in higher education institutions of the Ministry of Education (MOE), P. R. China.

REFERENCES AND NOTES

(1) Wales, D. J.; Scheraga, H. A. Global Optimization of Clusters, Crystals and Biomolecules. *Science* **1999**, *285*, 1368−1372.
(2) Wille, L. T. Minimum-energy Configuration of Atomic Clusters: New Results Obtained by Simulated Annealing. *Chem. Phys. Lett.* **1987**, *133*, 405−410.
(3) Schelstraete, S.; Verschelde, H. Finding Minimum-Energy Configurations of Lennard-Jones Clusters using an Effect Potential. *J. Phys. Chem. A* **1997**, *101*, 310−315.
(4) Ma, J.; Straub, J. E. Simulated Annealing using the Classical Density Distribution. *J. Chem. Phys.* **1994**, *101*, 533−541.
(5) Xue, G. Molecular Conformation on the CM-5 by Parallel Two-Level Simulated Annealing. *J. Glob. Opt.* **1994**, *4*, 187−208.
(6) Deaven, D. M.; Tit, N.; Morris, J. R.; Ho, K. M. Structural Optimization of Lennard-Jones Clusters by a Genetic Algorithm. *Chem. Phys. Lett.* **1996**, *256*, 195−200.
(7) Wolf, M. D.; Landman, U. Genetic Algorithms for Structural Cluster Optimization *J. Phys. Chem. A* **1998**, *102*, 6129−6137.
(8) Wales, D. J.; Doye J. P. K. Global Optimization by Basin-Hopping and the Lowest Energy Structures of Lennard-Jones Clusters Containing up to 110 Atoms. *J. Phys. Chem. A* **1997**, *101*, 5111−5116.
(9) Leary, R. H.; Doye, J. P. K. Tetrahedral Global Minimum for the 98-atom Lennard-Jones Cluster. *Phys. Rev. E* **1999**, *60*, 6320−6322.
(10) Northby, J. A. Structure and Binding of Lennard-Jones Clusters: $13 \leq N \leq 147$. *J. Chem. Phys.* **1987**, *87*, 6166−6177.
(11) Romero, D.; Barrón, C.; Gómez, S. The Optimal Geometry of Lennard-Jones Clusters: 148−309. *Comput. Phys. Commun.* **1999**, *123*, 87−96.
(12) Pillardy, J.; Piela, L. Molecular Dynamics on Deformed Potential Energy Hypersurfaces. *J. Phys. Chem.* **1995**, *99*, 11805−11812.
(13) Leopold, P. E.; Montal, M.; Onuchic, J. N.A Kinetic Approach to the Sequence-structure Relationship. *Proc. Natl. Acad. Sci. U.S.A.* **1992**, *89*, 8721−8725.
(14) Doye, J. P. K.; Miller M. A.; Wales, D. J. The Double-Funnel Energy Landscape of the 38-atom Lennard-Jones Cluster. *J. Chem. Phys.* **1999**, *110*, 6896−6906.
(15) Doye, J. P. K.; Miller M. A.; Wales, D. J. Evolution of the Potential Energy Surface with Size for Lennard-Jones Clusters. *J. Chem. Phys.* **1999**, *111*, 8417−8428.
(16) Cai, W. S.; Shao, X. G. A Fast Annealing Evolutionary Algorithm for Global Optimization. *J. Comput. Chem.* **2002**, *23*, 427−435.
(17) Cai, W. S.; Feng, Y.; Shao, X. G. et al. Optimization of Lennard-Jones Atomic Clusters. *J. Mol. Struct.* (*Theochem*) **2002**, *579*, 229−234.
(18) Xia, B.; Cai, W. S.; Shao, X. G. et al. Chiral Recognition Study for the Inclusion Complexes of Amino Acids with α-cyclodextrin using a Fast Annealing Evolutionary Algorithm *J. Mol. Struct.* (*Theochem*) **2001**, *546*, 33−38.
(19) Hartke, B. Global Cluster Geometry Optimization by a Phenotype Algorithm with Niches: Location of Elusive Minima, and Low-Order Scaling with Cluster Size. *J. Comput. Chem.* **1999**, *20*, 1752−1759.
(20) Ingber, L.; Rosen, B. Genetic Algorithms and Very Fast Simulated Reannealing: A Comparison. *Mathmatical Computer Modeling* **1992**, *16*, 87−100.
(21) Ingber, L. Very Fast Simulated Re-Annealing. *Mathmatical Computer Modeling* **1989**, *12*, 967−973.
(22) Liu, D. C.; Nocedal, J. On the Limited Memory BFGS Method for Large Scale Optimization. *Mathematical Programming B* **1989**, *45*, 503−528.
(23) Doye, J. P. K.; Wales, D. J. The Structure of $(C60)_N$ Clusters. *Chem. Phys. Lett.* **1996**, *262*, 167−174.

CI020004I