Continuous extremal optimization for Lennard-Jones clusters

Tao Zhou,[1] Wen-Jie Bai,[2] Long-Jiu Cheng,[2] and Bing-Hong Wang[1,*]
[1]*Nonlinear Science Center and Department of Modern Physics, University of Science and Technology of China, Hefei Anhui, 230026, China*
[2]*Department of Chemistry, University of Science and Technology of China, Hefei Anhui, 230026, China*
(Received 17 November 2004; revised manuscript received 19 April 2005; published 6 July 2005)

We explore a general-purpose heuristic algorithm for finding high-quality solutions to continuous optimization problems. The method, called continuous extremal optimization (CEO), can be considered as an extension of extremal optimization and consists of two components, one which is responsible for global searching and the other which is responsible for local searching. The CEO's performance proves competitive with some more elaborate stochastic optimization procedures such as simulated annealing, genetic algorithms, and so on. We demonstrate it on a well-known continuous optimization problem: the Lennard-Jones cluster optimization problem.

DOI: 10.1103/PhysRevE.72.016702 PACS number(s): 02.60.Pn, 36.40.−c, 05.65.+b

I. INTRODUCTION

The optimization of a system with many degrees of freedom with respect to some cost function is a frequently encountered task in physics and beyond. One special class of algorithms used for finding the high-quality solutions to those NP-hard optimization problems is the so-called nature inspired algorithms, including simulated annealing (SA) [1,2], genetic algorithms (GA) [3–5], genetic programming (GP) [6], and so on.

In recent years, a nature-inspired algorithm named extremal optimization (EO) was proposed by Boettcher and Percus [7–11], which is very sententious and competitive compared with some well-known algorithms like SA, GA, GP, etc. To make the underlying mechanism of EO more concrete, let us focus on the natural selection of a biological system. In nature, highly specialized, complex structures often emerge when their most inefficient elements are selectively driven to extinction. For example, evolution progresses by selecting against the few most poorly adapted species, rather than by expressly breeding those species best adapted to their environment. The principle that the least-fit elements are progressively eliminated has been applied successfully in the Bak-Sneppen model [12,13], where each individual corresponding a certain species is characterized by a fitness value, and the least fit one with smallest fitness value and its closest dependent species are successively selected for adaptive changes. The extremal optimization algorithm draws upon the Bak-Sneppen mechanism, yielding a dynamic optimization procedure free of selection parameters.

Here we consider a general optimization problem, where the system consists of N elements, and we wish to minimize the cost function $C(S)$ depending on the system configuration S. The EO algorithm proceeds as follows.

(1) Choose an initial configuration S of the system at will; set $S_{\text{best}} := S$.

(2) Evaluate the fitness value f_i for each individual i and rank each individual according to its fitness value so that the least-fit one is in the top. Use k_i to denote the individual i's rank; clearly, the least-fit one is of rank 1. Choose one individual j that will be changed with the probability $P(k_j)$, and then, only randomly change the state of j and keep all other individuals' state unaltered. Accept the new configuration S' unconditionally $S := S'$, and if $C(S) < C(S_{\text{best}})$, then set $S_{\text{best}} := S$.

(3) Repeat step (2) as long as desired.

(4) Return to S_{best} and $C(S_{\text{best}})$.

The efficiency of the EO algorithm is sensitive to the probability function $P(k)$. In basic EO, $P(1)=1$, and for any $k(2 \leqslant k \leqslant N)$, $P(k)=0$. A more efficient algorithm, the so-called τ-EO, can be obtained through a slight modification from basic EO. In τ-EO, $P(k) \sim k^{-\tau}$ where $\tau > 0$. Of course, aiming at idiographic optimization problems, one can design various forms of $P(k)$ to improve the performance of basic EO. For example, Middleton has proposed the jaded extremal optimization (JEO) method for the Ising spin glass system by reducing the probability of flipping previously selected spins, which remarkably improved the efficiency of EO [14].

The previous studies indicate that the EO algorithm can often outperform some far more complicated or finely tuned algorithm, such as SA and GA, on some famous NP-hard [15] discrete optimization problems, including graph partitioning [7,8,16], traveling salesman problem [7], three-coloring problem [17,18], finding the lowest-energy configuration for the Ising spin glass system [14,17,19], and so on. However, many practical problems cannot be abstracted to discrete form. Thus to investigate EO's efficiency on continuous optimization problems [20] is not only of theoretic interest, but also of prominent practical worthiness.

In this paper, a so-called continuous extremal optimization (CEO) algorithm aiming at the continuous optimization problem will be introduced, which can be considered as a mixing algorithm consisting of two components: one is responsible for global searching and the other is responsible for local searching. The CEO's performance proves competitive with some more elaborate stochastic optimization proce-

*Electronic address: bhwang@ustc.edu.cn

dures such as simulated annealing, genetic algorithms, and so on. We demonstrate it on a well-known continuous optimization problem: the Lennard-Jones (LJ) cluster optimization problem.

This paper is organized as follows: in Sec. II, the LJ clusters optimization problem will be briefly introduced. In Sec. III, we will give the algorithm proceeds of CEO. Next, we give the computing results about the performance of CEO on the LJ cluster optimization problem. Finally, in Sec. V, the conclusion is drawn and the relevance of the CEO to the real-life problems is discussed.

II. LENNARD-JONES CLUSTER OPTIMIZATION PROBLEM

The continuous optimization problem is ubiquitous in materials science: many situations involve finding the structure of clusters and the dependence of structure on size is particularly complex and intriguing. In practice, we usually choose a potential function to take the most steady structure since it is considered to be in possession of the minimum energy. However, in all but the simplest cases, these problems are complicated due to the presence of many local minima. Such a problem is encountered in many areas of science and engineering—for example, the notorious protein folding problem [21].

As one of the simplest models that exhibits such behavior [22] one may consider the problem of finding the ground-state structure of a nanocluster of atoms interacting through a classical Lennard-Jones pair potential, in reduced units, given by

$$V(r) = \frac{1}{r^{12}} - \frac{1}{r^{6}}, \tag{1}$$

where r is the distance between two atoms. This potential has a single minimum at $r_e = \sqrt[6]{2}$, which is the equilibrium distance of two atoms. It can, of course, easily be reduced to an arbitrary LJ potential by a simple rescaling of length and energy units. The ith atom has energy

$$E_i = \frac{1}{2}\sum_{j \neq i} V(r_{ij}), \tag{2}$$

and the total energy for N atoms is

$$E = \sum_i E_i. \tag{3}$$

The optimization task is to find the configuration with minimum total potential energy of a system of N atoms, each pair interacting by potential of the form (1). Clearly, a trivial lower bound for the total energy is $-N(N-1)/2$, obtained when one assumes that all pairs are at their equilibrium separation. For N=2,3,4 the lower bound can actually be obtained in three-dimensional space, corresponding, respectively, to a dimer, equilateral triangle, and regular tetrahedron, with all interatomic distances equal to r_e. However, from N=5 onwards it is not possible to place all the atoms simultaneously at the potential minimum of all others and the ground-state energy is strictly larger than the trivial lower bound. This system has been studied intensely [23] and is known to have an exponential increasing number of local minima, growing roughly as $e^{0.36N+0.03N^2}$ near $N=13$, at which point there are already at least 988 minima [23]. If this scaling continues, more than 10^{140} local minima exist when N approaches 100.

III. CONTINUOUS EXTREMAL OPTIMIZATION

The continuous extremal optimization algorithm consists of two components: one is the classical EO algorithm responsible for global searching and the other is a certain local searching algorithm. We give the general form of CEO algorithm by way of the LJ cluster optimization problem as follows:

(1) Choose an initial state of the system, where all the atoms are placed within a spherical container with radius [24,25]

$$R = r_e\left[\frac{1}{2} + \left(\frac{3N}{4\pi\sqrt{2}}\right)^{1/3}\right], \tag{4}$$

where $r_e = \sqrt[6]{2}$ is the equilibrium distance and N denotes the number of atoms. Set the minimal energy $E_{\min}=0$.

(2) Use a certain local searching algorithm to find the local minimum from the current configuration of system. If the local minimal energy is lower than $E_{\min}$, then replace $E_{\min}$ by the present local minimum.

(3) Rank each atom according to its energy obtained by Eq. (2). Here, the atom that has highest energy is the least-fit one and is arranged in the top of the queue. Choose one atom j that will be changed with the probability $P(k_j)$ where k_j denotes the rank of atom j, and then, only randomly change the coordinates of j and keep all other atoms' positions unaltered. Accept the new configuration unconditionally. Here one should repeat step (3) several times to make the system configuration far away from last local minimum.

(4) Repeat steps (2) and (3) as long as desired.

(5) Return to the minimal energy $E_{\min}$ and the corresponding configuration.

For an idiographic problem, one can attempt various local searching algorithms and pitch on the best one. In this paper, for the LJ cluster optimization problem, we choose the limited-memory BFGS method (LBFGS) as the local searching algorithm. The BFGS method is an optimization technique based on a quasi-Newtonian method proposed by Broyden, Fletcher, Goldfard, and Shanno, which has become more and more popular and today is accepted as one of the best quasi-Newton methods [26], but of course, cannot escape the local minima. The LBFGS method proposed by Liu and Nocedal [25,27] is especially effective with problems involving a large number of variables. In this method, an approximation H_k to the inverse of the Hessian is obtained by applying M BFGS updates to a diagonal matrix H_0, using information from the previous M steps. The number M determines the amount of storage required by the routine, which is specified by the user, usually $3 \leqslant M \leqslant 7$, and in our computation M is fixed as 4.

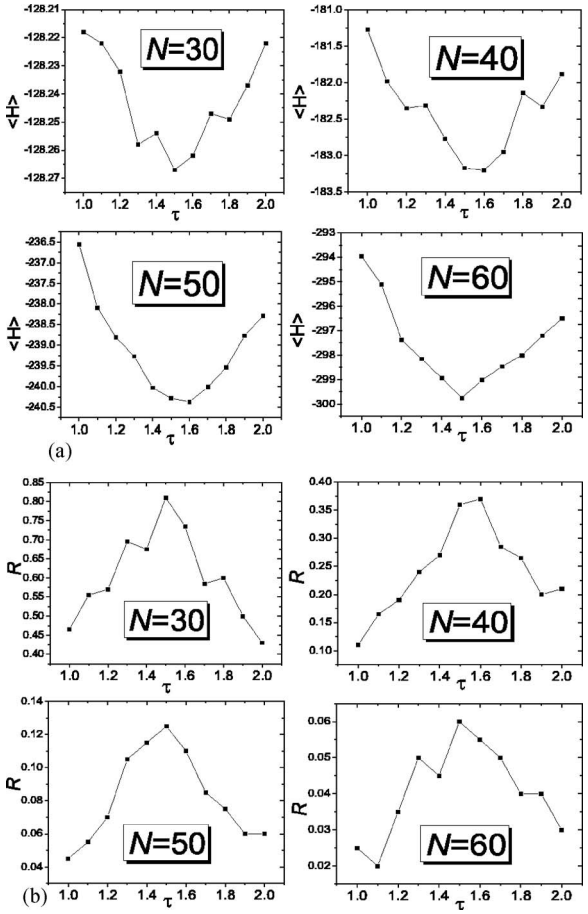

FIG. 1. The details of τ-CEO for $\tau \in [1,2]$. (a) shows the average energies obtained by CEO over 200 runs, and (b) exhibits the success rate of hitting the global minima in 200 runs [28]. For both (a) and (b), the four plots are the cases $N=30$, $N=40$, $N=50$, and $N=60$, respectively. One can find that the best τ corresponding lowest average energy and highest success rate is approximate to 1.5.

IV. COMPUTING RESULTS

Similar to τ-EO, we use τ-CEO algorithm for the LJ cluster optimization problem, where the probability function of CEO is $P(k) \sim k^{-\tau}$. Since there are $N^2/2$ pairs of interactional atoms in a LJ cluster of size N, we require αN^2 updates where α is a constant and fixed as 100 in the following computation. In order to avoid falling into the same local minimum too many times, before running the LBFGS algorithm, we should make the system configuration far away from last local minimum. Thus we run the LBFGS algorithm every 20 time steps. That is to say, for a LJ cluster of size N, the present algorithm runs EO $100N^2$ times and LBFGS $5N^2$ times in total.

We have carried out the τ-CEO algorithm so many times for different τ and N, and find that the algorithm performs better when τ is in the interval [1,2]. In Fig. 1, we report the details for $1 \leqslant \tau \leqslant 2$, where Fig. 1(a) shows the average energies obtained by CEO over 200 runs, and Fig. 1(b) exhibits the success rate R of hitting the global minima [28]. For both Figs. 1(a) and 1(b), the four plots are the cases $N=30$, $N=40$, $N=50$, and $N=60$, respectively. The readers should note that, although the difference of average energies between two different τ is great in the plot, it is very small in fact. One can find that, for most cases, the best τ corresponding lowest average energy and highest success rate is approximate to 1.5. Only when $N=40$ does $\tau=1.6$ perform better than $\tau=1.5$. Therefore, in the following computation, we set $\tau=1.5$. We have also compared the performance of CEO on larger LJ clusters for $\tau=1.5$ and $\tau=1.6$; the two cases are pretty much the same thing and $\tau=1.5$ is a little better.

It is clear that if τ is too small, the algorithm will be close to a random walk algorithm and the progress of the search becomes undirected. On the other hand, if τ is too large, the process approaches a deterministic local search with only the poorest atom being changed in each configuration; thus the results must be of poor quality. Some researchers have supposed that the optimal value of τ is closely related to a transition from ergodic to nonergodic behavior [7]. This is an interesting topic that may become one of our future works, but now we cannot say anything about it.

We demonstrate that for all the LJ clusters of size N not more than 100, the global minima can be obtained by using CEO algorithm. In Fig. 2, we report the performance of CEO on LJ cluster optimization problem according to 200 independent runs. In Fig. 2(a), the black squares, red circles, blue triangles, and green stars represent the global minima [28], the average energies obtained by CEO, the average energies obtained by the random walk (RW) algorithm, and the average energies obtained by the LBFGS algorithm only, respectively. In the RW algorithm, we replace the EO process with randomly exchanging the coordinates of a randomly selected atom and keep other processes unaltered. Since the LBFGS algorithm is not an active searching algorithm, it could not deal with the large-scale optimal problems. And when some active searching process is combined to the LBFGS algorithm, the performance will be sharply improved. One can see clearly that the CEO performs much better than the RW algorithm, which exhibits the value of the EO part. However, the deviation between the result of CEO and global minimum becomes greater and greater when the cluster size gets larger and larger, which indicates that for very large LJ cluster, CEO may be a poor algorithm. Figure 2(b) shows the success rate of hitting the global minima in 200 runs; the inset is the success rate for $N>50$, which may be unclear in the main plot. For both cases $N=95$ and $N=100$, the global optimal solution appears only once in 200 runs. The success rate decreases sharply for $N<45$ and then decreases gently. According to Wille's work [29], it is probably some crossover related to the finite run time.

Finally, we investigate the average CPU time over 200 runs versus the size of the LJ clusters. The computations were carried out in a single Pentium III processor (1 GHZ). From Fig. 3, in the log-log plot, the data can be well fitted by a straight line with slope 3.886 ± 0.008, which indicates that the increasing tendency of CPU time T versus cluster size is approximate to a power-law form as $T \sim N^{3.886}$. That means the CEO is a polynomial algorithm of order $O(N^4)$.

V. CONCLUSION AND DISCUSSION

In this paper, we explored a general-purpose heuristic algorithm for finding high-quality solutions to continuous op-

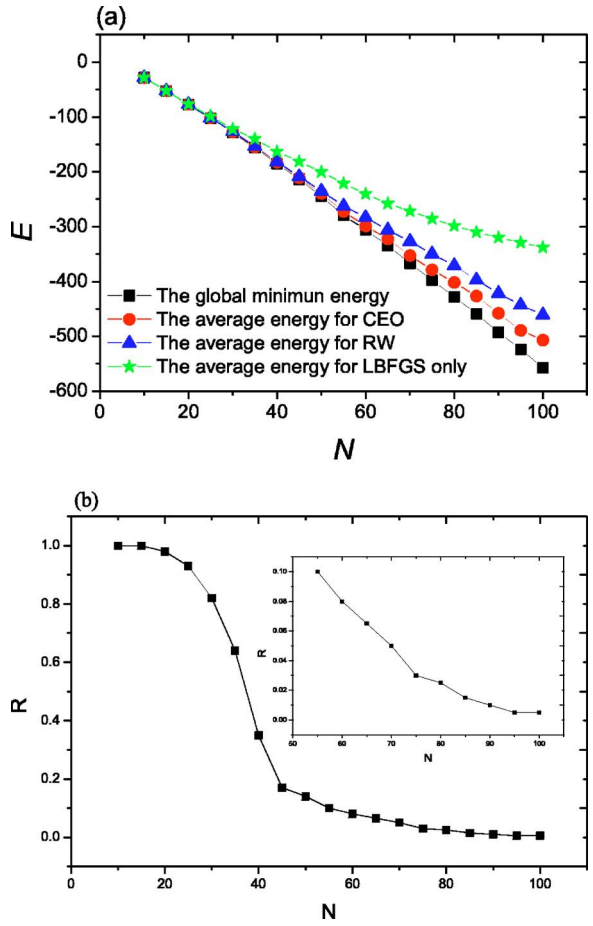

FIG. 2. (Color online) The performance of CEO algorithm on the LJ cluster optimization problem. In (a), the black squares represent the global minima, and the red circles, blue triangles, and green stars represent the average energies obtained by CEO, RW algorithm, and LBFGS only, respectively. (b) shows the success rate of hitting the global minima in 200 runs. The inset is the success rate for $N>50$, which may be unclear in the main plot.

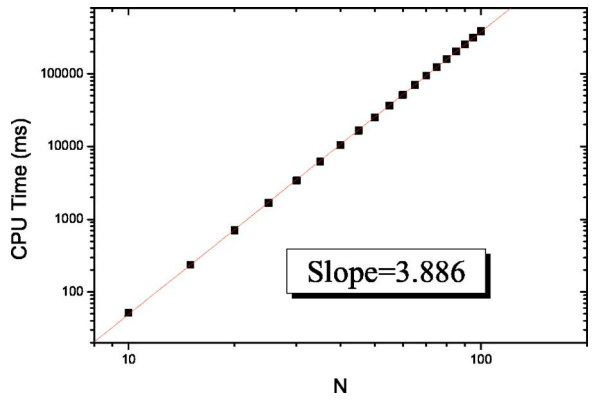

FIG. 3. (Color online) The average CPU time (ms) over 200 runs vs the size of LJ clusters. In the log-log plot, the data can be well fitted by a straight line with slope 3.886±0.008, which indicates that the increasing tendency of CPU time T vs cluster size is approximate to a power-law form as $T\sim N^{3.886}$.

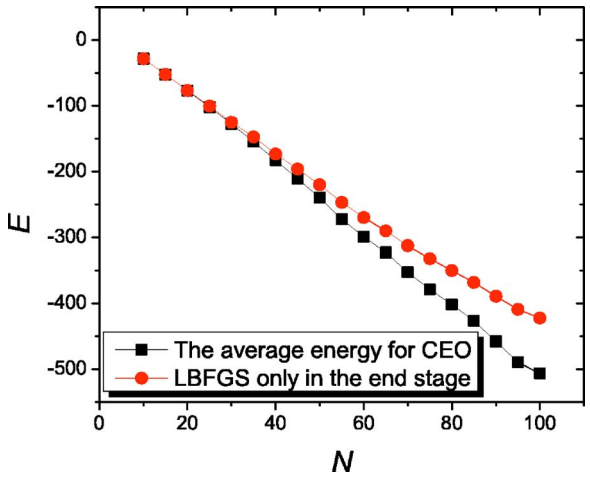

FIG. 4. (Color online) The performance of CEO algorithm on the LJ cluster optimization problem. The black squares and red circles represent the average energies obtained by CEO and using EO process attaching the LBFGS algorithm in the end stage only.

timization problems. The computing results indicate that this simple approach is competitive and sometimes can outperform some far more complicated or finely tuned nature-inspired algorithm on a well-known NP-hard continuous optimization problem for LJ clusters. For example, by combining the genetic algorithm and conjugate-gradient minimization, one can get global minima for $N<100$ [30,31]. Another example is to combine simulated annealing and gradient minimizer [29]. However, this method is valid only for very small N (e.g., $N=6$ and $N=13$) and the corresponding success rate is much smaller than CEO. According to EO's updating rule, it is clear that EO has very high ability in global searching; thus to combine EO and a strong local searching algorithm may produce a highly efficient algorithm for continuous optimization problems.

Recently, several algorithms aiming at the LJ cluster optimization problem have been proposed. Cai *et al.* have schemed out a so-called fast annealing evolutionary algorithm, which can obtain all the structure of known minimal energy until $N=116$ [25]. Lee *et al.* proposed the conformational-space annealing method, which finds all known lowest-energy configurations up to 201 atoms [32]. The adaptive immune optimization algorithm proposed by Shao *et al.* can find the optimal structure of $N<80$ with a very high efficiency [33]. And using the cluster similarity checking method, this upper limit can be pushed to $N=200$ [34]. These algorithms are more concerned with the special information about LJ clusters and perform better than CEO in some aspects. However, we have not found compelling evidence indicating that there exists a general-purpose algorithm like SA or GA entirely preponderate over CEO on the LJ cluster optimization problem. It is worthwhile to emphasize that, in this paper, we do not want to prove that the CEO is an all-powerful algorithm, even do not want to say that the CEO is a good choice for chemists on the LJ cluster optimization problem since a general-purpose method often performs poorer than some special methods aiming at an idiographic problem. The only thing we want to say is that the

CEO, an extension of the nature-inspired algorithm EO, is a competitive algorithm and needs more attention.

The previous studies of simulated annealing and the genetic algorithm indicate that in most cases local search techniques are most useful at the end stages of the search. However, in the current paper, the local search technique (LBFGS) is inserted in the process of a global search. It just because the LJ cluster of minimal energy always consists of some regular substructure, such as icosahedron, Marksdecahedron, face-centered-cubic (fcc) truncated octahedron, and so on. Using local searches several times during the process of a global search may be helpful to generate these regular modules. In Fig. 4, we report the performance using the LBFGS algorithm once only in the end stage. Clearly, it performs much more poorly than the present CEO algorithm.

Furthermore, to demonstrate the efficiency of CEO, much more experiments on various hard continuous optimization problems should be achieved, which are future works.

ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China under Grant Nos. 10472116, 70471033, and 70271070, the Specialized Research Fund for the Doctoral Program of Higher Education (SRFDP No. 20020358009), and the Foundation for Graduate Students of University of Science and Technology of China under Grant No. KD200408.

[1] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, Science **220**, 671 (1983).
[2] E. H. L. Aarts and J. H. M. Korst, *Simulated Annealing and Boltzmann Machines* (Wiley, New York, 1989).
[3] J. Holland, *Adaption in Natural and Artificial Systems* (The University of Michigan Press, Ann Arbor, MI, 1975).
[4] D. G. Bounds, Nature (London) **329**, 215 (1987).
[5] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning* (Addison-Wesly, Reading, MA, 1989).
[6] W. Banzhaf, P. Nordin, R. Keller, and F. Francone, *Genetic Programming—An Introduction* (Morgan Kaufmann, San Francisco, 1998).
[7] S. Boettcher and A. G. Percus, Artif. Intell. **199**, 275 (2000).
[8] S. Boettcher, J. Phys. A **32**, 5201 (1999).
[9] S. Boettcher, Comput. Sci. Eng. **2**, 6 (2000).
[10] S. Boettcher, Comput. Sci. Eng. **2**, 75 (2000).
[11] S. Boettcher, A. G. Percus, and M. Grigni, Lect. Notes Comput. Sci. **1917**, 447 (2000).
[12] P. Bak and K. Sneppen, Phys. Rev. Lett. **71**, 4083 (1993).
[13] K. Sneppen, P. Bak, H. Flyvbjerg, and M. H. Jensen, Proc. Natl. Acad. Sci. U.S.A. **92**, 5209 (1995).
[14] A. A. Middleton, Phys. Rev. E **69**, 055701(R) (2004).
[15] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-completeness* (Freeman, New York, 1979).
[16] S. Boettcher and A. G. Percus, Phys. Rev. E **64**, 026114 (2001).
[17] S. Boettcher and A. G. Percus, Phys. Rev. Lett. **86**, 5211 (2001).
[18] S. Boettcher and A. G. Percus, Phys. Rev. E **69**, 066703 (2004).
[19] S. Boettcher and M. Grigni, J. Phys. A **35**, 1109 (2002).
[20] In the discrete optimization problem, the set of all the states of the system is denumerable, while in the continuous optimization problem, the corresponding set is a continuum.
[21] T. P. Martin, T. Bergmann, H. Göhilich, and T. Lange, Chem. Phys. Lett. **172**, 209 (1990).
[22] L. T. Wille, in *Annual Reviews of Computational Physics VII*, edited by D. Stauffer (World Scientific, Singapore, 2000).
[23] M. R. Hoare, Adv. Chem. Phys. **40**, 49 (1979), and references therein.
[24] B. Xia, W. Cai, X. Shao, Q. Guo, B. Maigret, and Z. Pan, J. Mol. Struct.: THEOCHEM **546**, 33 (2001).
[25] W. Cai, Y. Feng, X. Shao, and Z. Pan, J. Mol. Struct.: THEOCHEM **579**, 229 (2002).
[26] W. F. Mascarenhas, Math. Program. **99**, 49 (2004).
[27] D. C. Liu and J. Nocedal, Math. Program. **45**, 503 (1989).
[28] The complete up-to-date list of the global minima of LJ clusters can be downloaded from the web http://brian.ch.cam.ac.uk/. It is worthwhile to emphasize that in this paper the term "global minimum" is only customary parlance for the already obtained minimal energy up to now. It may be not the truest global minimum especially for large *N*.
[29] L. T. Wille, Comput. Mater. Sci. **17**, 551 (2000).
[30] D. M. Deaven and K. M. Ho, Phys. Rev. Lett. **75**, 288 (1995).
[31] D. M. Deaven, N. Tit, J. R. Morris, and K. M. Ho, Chem. Phys. Lett. **256**, 195 (1996).
[32] J. Lee, I.-H. Lee and J. Lee, Phys. Rev. Lett. **91**, 080201 (2003).
[33] X. Shao, L. Cheng, and W. Cai, J. Chem. Phys. **120**, 11401 (2004).
[34] L. Cheng, W. Cai, and X. Shao, Chem. Phys. Lett. **389**, 309 (2004).