PAPER

Lower-Energy Structure Optimization of $(C_{60})_N$ Clusters Using an Improved Genetic Algorithm

Guifang SHAO^{†a)}, Wupeng HONG[†], *Nonmembers*, Tingna WANG[†], *Member*, and Yuhua WEN^{††}, *Nonmember*

SUMMARY An improved genetic algorithm is employed to optimize the structure of $(C_{60})_N$ ($N \leq 25$) fullerene clusters with the lowest energy. First, crossover with variable precision, realized by introducing the hamming distance, is developed to provide a faster search mechanism. Second, the bit string mutation and feedback mutation are incorporated to maintain the diversity in the population. The interaction between C_{60} molecules is described by the Pacheco and Ramalho potential derived from first-principles calculations. We compare the performance of the Improved GA (IGA) with that of the Standard GA (SGA). The numerical and graphical results verify that the proposed approach is faster and more robust than the SGA. The second finite differential of the total energy shows that the $(C_{60})_N$ clusters with $N = 7, 13, 22$ are particularly stable. Performance with the lowest energy is achieved in this work.

key words: $(C_{60})_N$ clusters, genetic algorithm, population diversity, PR potential

1. Introduction

Clusters, which are aggregates of atoms and/or molecules held together by different types of bonding or nonbonding interactions, constitute a new type of material which may have properties that are distinct from those of discrete molecules or bulk materials. Intense interest in clusters arises from the size-dependent evolution of their geometric and electronic structures, thereby, influencing the clusters' chemical and physical properties. Because the relationship between the structure and properties of clusters is of crucial importance in the exploration of new materials, the investigation of the clusters' geometric structures is an important task in cluster research. However, the structure investigation of clusters, both experimentally and theoretically, is a challenge to scientists [1].

Since the break-through in the macroscopic production and characterization of C_{60} in 1990, fullerenes have attracted considerable attention from scientists in various fields. Extensive experimental and theoretical investigations, including structure studies, of C_{60} and related clusters have been performed in recent decades [2], [3]. The interaction between C_{60} clusters and their practical applications has attracted much attention [4]. Fullerene C_{60} , whose molecules are nearly spherical and attracted to each other by a van der Waals force, consists of 60 C carbon

atoms held together by covalent bonds on the surface of a truncated icosahedral structure [5]. The stable structure of $(C_{60})_N$ clusters has been extensively studied from many perspectives [6]. However, the following three major challenging issues remain in determining the optimal cluster structure: (1) How can we model the interaction between C_{60} molecules for better determining the lower-energy structures? (2) How to make a good choice with regard to the quality and efficiency of the optimal structural method? and (3) How to arrive at valid conclusions according to the optimal result? Whether one is using empirical potentials or ab initio theory to describe the bonding in clusters, one of the principal objectives is to find, for a given cluster size, the arrangement of atoms (or ions or molecules) corresponding to the lowest potential energy—i.e., the global minimum (GM) on the potential-energy hypersurface [7]. However, as the number of minima rises quasi-exponentially with increasing cluster size, finding the GM becomes increasingly difficult.

The structural stability problem of $(C_{60})_N$ clusters, therefore, can be transformed into a mathematical matter, i.e., when the value of N is known, how to place the C_{60} molecules in 3D space so that the total potential energy is lowest. In this study, the Improved Genetic Algorithm (IGA) was used in an attempt to find the global lowest energy structures for $(C_{60})_N$ clusters with $N \leq 25$. Crossover, based on hamming distance, is developed to achieve better results more rapidly. Feedback mutation is combined with bit string mutation to guarantee diversity in population. Experimental results verified the performance of the IGA. Also, based on the results of 100 independent runs, conformational analysis was conducted.

The rest of the paper is organized as follows: Section 2 presents a brief summary of prior approaches relevant to structure optimization of $(C_{60})_N$ clusters. In Sect. 3, the algorithmic details are introduced. Extensive experimental results are reported in Sect. 4. Finally Sect. 5 concludes the paper.

2. Related Work

The determination for the ground-state structure of an atomic, or molecular, cluster is one of the most fundamental and challenging problems in cluster physics. The structure of the system is governed by the high dimensionality of the energy hypersurfaces, which gives rise to many local minima. In order to search the global minimum, without being trapped in a local minimum, many methods are proposed.

Manuscript received February 15, 2013.

Manuscript revised June 5, 2013.

[†]The authors are with the Department of Automation, Xiamen University, Xiamen, 361-005 China.

^{††}The author is with the Department of Physics, Xiamen University, Xiamen, 361-005 China.

a) E-mail: gfshao@xmu.edu.cn

DOI: 10.1587/transinf.E96.D.2726

2.1 Cluster Structure Optimization with Traditional Methods

The most commonly used strategies for cluster structure optimization are (1) the simulated annealing scheme based on Monte Carlo or (2) molecular dynamics (MD) simulation.

Doye and Wales have presented likely candidates (modeled by Girifalco's potential) for the lowest-energy minima of (C₆₀)_N clusters with $N < 80$ [8]. Rey et al., using Girifalco's intermolecular potential, have performed the MD simulation for (C₆₀)_N clusters up to $N = 25$ [9]. Doye and Wales located global minima for (C₆₀)_N clusters (modeled by the potential of Pacheco and Ramalho) up to $N = 105$, and used the Monte Carlo plus minimization method to locate the global minima [10].

However, such traditional methods yield results that easily fall within the local minima.

2.2 Cluster Structure Optimization Using a Genetic Algorithm

To date, a small number of theoretical studies, using a genetic algorithm, have been performed on (C₆₀)_N clusters [5]. Y.H. Luo et al. predicted a magic number structure of neutral (C₆₀)_N clusters up to $N = 25$ in combination with genetic algorithms (GA's) and PR potential [11]. W. Zhang, using a global optimization procedure that combines the genetic algorithm and database conjugate-gradient method, optimized the lowest-energy structures of fullerene clusters (C₆₀)_N ($N < 57$) described by PR potential [12]. W.S. Cai employed a parallel fast annealing evolutionary algorithm (PFAEA) to optimize the structures of (C₆₀)_N molecular clusters up to $N = 80$ with the lowest energy based on Girifalco's intermolecular potential [13]. GA, especially SGA, is very time-consuming and totally random. Therefore, L.J. Cheng et al., using the Girifalco potential, applied the dynamic lattice searching (DLS) method to the global optimization of (C₆₀)_N clusters up to $N = 150$ [13]. GA samples the configuration space more efficiently. In the standard GA, however, there are some structural problems, such as premature convergence.

2.3 The Standard Genetic Algorithm

The genetic algorithm is an optimization technique based on the principles of natural evolution [14]. The idea of the SGA is to select the best performing, or "fittest," individuals for breeding, and in this way encourage good performing individuals [15]. Crossover and mutation, as two operators, are used to create offspring, then populate the next generation of the population by replacing the least fittest individuals. By an iteration of these operations, the population evolves towards containing more and more fit individuals. Usually a chromosome representation of the individuals is provided to simplify the crossover operations. The fitness of an individual is a measure of how well the individual performs

when applied to the problem at hand and is calculated by the fitness function. The representation mapping and fitness function are always problem dependent.

Execution of the SGA is as follows:

- (1) Candidate solutions are encoded as fixed length vectors, and from among these vectors, an initial one is chosen at random.
- (2) A group of individuals is randomly selected from the population with a probability determined by their fitness.
- (3) The mating operator is based on single-point crossover at a random point.
- (4) The mutation operator may randomly alter one individual in the population according to a probability.

Even though the SGA is a powerful optimization tool, it does have some problems. The basic problems of SGA are its lack of local search and its risk of premature convergence. Furthermore SGA requires long execution-times when applied to large and complex problems.

3. Computational Details

3.1 The PR Potential

In this paper, the interaction between C₆₀ molecules is modeled by the Pacheco and Ramalho potential (PRP), but only the PRP's two-body part is included [16]. Compared to Girifalco's potential, PRP has a relative softness. According to PRP, the fullerene is treated as a spherical molecule, and the interaction depends only on the distance between the centers of the fullerene molecules [17].

The parameter-free expression of the interaction between fullerenes can be denoted as:

$$V(r) = F(r) \times M(r) + [1 - F(r)] \times W(r) \quad (1)$$

Where r is the distance between two molecules and is computed by the following formula:

$$r = 3.0634 \times \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \quad (2)$$

x_i, y_i, z_i ($i = 1, 2$) is the coordinate of each molecule. $M(r)$ is a Morse potential

$$M(r) = M_0 \exp \left[\tau \left(1 - \frac{r}{d_0} \right) \right] \left\{ \exp \left[\tau \left(1 - \frac{r}{d_0} \right) \right] - 2 \right\} \quad (3)$$

A van der Waals interaction potential is expanded as follows:

$$W(r) = -\frac{C_6}{r^6} - \frac{C_8}{r^8} - \frac{C_{10}}{r^{10}} - \frac{C_{12}}{r^{12}} \quad (4)$$

A function, $F(r)$, of the Fermi type describes the crossover of these two regimes and is expressed as $F(r) = [1 + \exp((r - \mu)/\delta)]^{-1}$.

The PR parameters used in this study are listed in Table 1.

3.2 The Optimization Method

A flowchart representing the operation of our cluster geometry optimization IGA program is shown in Fig. 1. For

Table 1 Parameters defining the PR potential for $(C_{60})_N$ clusters.

Parameter	M_0 / eV	$d_0 / \text{\AA}$	τ	$\delta / \text{\AA}$	$\mu / \text{\AA}$
Value	0.3	10.3	9.75	1.04	10.05
Parameter	$C_6 / \text{eV}\text{\AA}^6$	$C_8 / \text{eV}\text{\AA}^6$	$C_{10} / \text{eV}\text{\AA}^8$	$C_{12} / \text{eV}\text{\AA}^8$	
Value	$21N_c^2$	$2534N_c^2$	2.09×10^8	7.78×10^{10}	

N_c is the number of atoms in a single fullerene, according to C_{60} $N_c = 60$. [16]

Fig. 1 The flow chart of IGA.

a given cluster size, the practical operation of IGA used here is basically that shown in Fig. 1.

A string, representing a solution to the problem, is encoded as a vector of random binary numbers. Each solution string is composed of N genes, where N is the number of the points to be distributed within the unit sphere. The PR potential energy is used as an evaluation function associated with every string. Some new children are bred by performing crossovers and mutations. The new children and parents are combined and sorted according to fitness. Then, M fitter chromosomes are selected to create a new population for the next generation. The terminal conditions are defined as when either (1) the maximum number of generations is reached, or (2) the highest ranking solution's fitness has reached a plateau such that successive iterations no longer produce better results. To guarantee the population's diversity and avoid "premature convergence", the SGA is improved by introducing two additional criteria.

3.3 The Initial Population

Since it is convenient to use integer values for GA implementation, the lowest-energy problem of $(C_{60})_N$ clusters is encoded as follows:

$$\begin{cases} c = \{c_1, c_2, \dots, c_N\} \\ c_i = (x_i, y_i, z_i) \\ \{x_i, y_i, z_i\} = 0.01 * (\beta_1 \beta_2 \dots \beta_{11}) \quad \beta \in \{0, 1\} \end{cases}$$

where c is the chromosome which corresponds to a kind of structure of $(C_{60})_N$ clusters. c_i denotes the i^{th} gene, x_i, y_i, z_i represent the spatial coordinates of the i^{th} C_{60} molecule, respectively.

Then, the initial population is created as follows:

Step 1. A binary number $\beta, \beta \in \{0, 1\}$, is randomly generated and this process is repeated 11 times until one of the spatial coordinates of the i^{th} C_{60} molecule is generated, e.g. $x_i = 0.01 * (\beta_1 \beta_2 \dots \beta_{11})$.

Step 2. Step 1 is repeated three times to obtain the gene c_i , and then repeated until N different genes are generated, i.e. a member of an initial population is produced.

Step 3. Step 2 is repeated M times to create the initial population.

3.4 Crossover with Variable Precision

To search for a feasible solution from lower precision space to higher precision space and enhance the population diversity, we propose a crossover with variable precision based on convex set theory. To ensure the validity of crossover, we introduce hamming distance. The hamming distance of two chromosomes is denoted as:

$$D_{Hg}(x, y) = \sum_{i=1}^N d_i \quad d_i = \begin{cases} 1 & x_i \neq y_i \\ 0 & \text{else} \end{cases} \quad (5)$$

where x and y are two randomly chosen chromosomes, d_i describes the different gene between x and y .

When $D_{Hg}(x, y) = 0$, one of the two chromosomes is replaced by a randomly generated chromosome.

If $D_{Hg}(x, y) = 1$, it is assumed that the i^{th} gene is different, i.e. $x_i \neq y_i$, another gene, j ($j \neq i$), is selected as the new cut point and, by mutation, two new genes, x'_j, y'_j , are produced.

Suppose that $D_{Hg}(x, y) = n$, then the sequence number of the different gene is composed of the vector, Dif_gen , and crossover is accomplished as follows:

Step1. Randomly generate a vector Vec consisting of $1 \sim n$ with the members in the Vec differing from one another;

Step2. Produce a natural number r , where $r < n$;

Step3. Select several crossover points and swap the data in these points as follows:

$$\begin{cases} x'_{Dif_gen(Vec(m))} = y_{Dif_gen(Vec(m))} \\ y'_{Dif_gen(Vec(m))} = x_{Dif_gen(Vec(m))} \end{cases} \quad m \in [1, r] \quad (6)$$

For example, considering the following parent population

$$\begin{cases} x = \{1, 2, 3, 4, 5, 6, 7, 8\} \\ y = \{1, 1, 3, 2, 5, 3, 1, 8\} \end{cases}$$

$D_{H_g}(x, y) = 4$ and $Dif_gen = \{2, 4, 6, 7\}$. Suppose that $Vec = \{3, 2, 4, 1\}$ and $r = 3$, then the following two new children are produced $\begin{cases} x' = \{1, 2, 3, 2, 5, 3, 1, 8\} \\ y' = \{1, 1, 3, 4, 5, 6, 7, 8\} \end{cases}$.

3.5 Large Scale Mutation

During the evolution, some good individuals will generate more and more children, resulting in a decrease of the gene variance. Especially in the later stages of evolution, gene diversity becomes so poor that the population evolves slowly or comes to a complete stop. To avoid this problem, two types of mutation operators are considered, and this, to some extent, helps keep the population diversity high by preventing premature convergence within poor solutions.

(i) *Bit string mutation*. Bit string mutation inverts the bits at random positions.

(ii) *Feedback mutation*. Feedback mutation is realized as follows:

Step1. The fitness ratio $fit_{population}$ of the whole population is computed as

$$fit_{population} = \left(\sum_{i=1}^K f_i \right) / (K \times f_{optimal}) \quad (7)$$

where $f_{optimal}$ is the fitness of the optimal chromosome, f_i is the fitness of the i^{th} individual, and K is the population size.

$fit_{population}$ is used as a yardstick to measure premature convergence. Lower values of $fit_{population}$ help to maintain a high population diversity. When $fit_{population} = 1$, individuals that are very close to each other are apparently generated. **Step2.** If $fit_{population} > \alpha$ then the worst chromosome will be replaced by a new randomly generated one. Here $\alpha = 0.999$.

4. Results and Discussion

In our tests, we compared the IGA with SGA to evaluate its performance and implemented the programs in Matlab 6.5. The parameter settings, optimized by extensive experiments, are shown in Table 2.

4.1 Convergence Characteristic of IGA

Table 3 summarizes the comparison between IGA and SGA under some cluster sizes. We conclude that the IGA finds a lower energy than SGA with fewer generations. Furthermore, IGA successfully maintains population diversity in general.

Figure 2 portrays the evolution process of both methods. The IGA performs better in regard to convergence speed and global searching than the SGA. The SGA evolves faster at first, but slower later owing to a lack of population diversity, i.e. premature convergence. Our new algorithm converges more quickly than a SGA as the evolution progresses. Consequently, a lower energy is found in a shorter

Table 2 Parameters for IGA and SGA.

Parameter	Value	Parameter	Value
Population size	8	Coordinate encoding	11 bits
Mutation rate	0.3	Crossover rate	0.25
Maximum number of generations			10000

Table 3 Results using SGA and IGA for different sizes of $(C_{60})_N$ clusters.

Cluster size	3			4		
	Fitness	Aver	Gene	Fitness	Aver	Gene
SGA	-0.83369	0.8152	218	-1.66737	0.92065	741
IGA	-0.83369	0.62651	192	-1.66737	0.84698	624
Cluster size	7			9		
	Fitness	Aver	Gene	Fitness	Aver	Gene
SGA	-3.95579	0.90984	1448	-5.40202	0.98139	3439
IGA	-4.7446	0.83961	936	-6.48295	0.84115	1785
Cluster size	13			18		
	Fitness	Aver	Gene	Fitness	Aver	Gene
SGA	-8.58984	0.99016	4814	-12.9422	0.97641	5730
IGA	-10.7601	0.83362	2687	-17.1365	0.83588	3592
Cluster size	22			25		
	Fitness	Aver	Gene	Fitness	Aver	Gene
SGA	-17.2095	0.9951	7366	-20.6511	0.9981	9262
IGA	-21.9874	0.82921	5098	-25.145	0.84579	6445

Fig. 2 Comparison of IGA and SGA with the convergence plot of optimal fitness. (a) IGA uses 26 fewer generations than SGA to obtaining the same lowest-energy. (b) To find the same energy, -3.95 , IGA needs 267 generations, but SGA uses 978 generations. (c) IGA finds the lower energy first after 3,073 generations, whereas, SGA finds it after 4,618 generations. (d) IGA requires only 70 percent of the generations that SGA needs to find lower energy.

period of time. As the cluster size increases, the number of generations needed to locate the optimum also increases. For example, as shown in Fig. 2 (a), to find the same optimum value, -0.83369 , SGA and IGA use 192 and 218 generations, respectively. Figs. 2 (b), 2 (c) and 2 (d) also illustrate that IGA, with a faster speed, searches for a lower energy structure than SGA.

For instance, the IGA first locates the lower energy at the 553rd generation, but the SGA locates it at the 978th generation, as shown in Fig. 2 (b). Figure 2 (c) shows that the SGA uses 1.6 times more generations than the IGA uses. From the convergence curves of Fig. 2 (d), we see that the

Fig. 3 The fitness ratio curves of IGA and SGA under different cluster sizes. (a) The IGA has higher values than the SGA from 7 to 162 generations. (b) The fitness ratio value is higher in the IGA than the SGA from 7 to 30 generations. (c) All values in the IGA are lower than those in the SGA. (d) There are higher values in the IGA than in the SGA from 4 to 12 generations. The fitness ratio is defined by Eq. (7).

IGA finds an energy, -13.9337 , after 322 generations, but the SGA requires 1,456 generations to find a similar energy: -13.93194 .

4.2 Population Diversity Performance of IGA

The decreasing of population diversity leads directly to premature convergence. Higher population diversity also prevents the algorithm from converging prematurely to local optima. To illustrate here the performance of the IGA, we exploit the fitness ratio to evaluate the population diversity. As shown in Fig. 3, the population diversity becomes worse and worse as the population evolves. The IGA has lower fitness ratio values than the SGA on the whole, indicating that the IGA is well suited to maintain diversity in the population. The SGA performs well for the small values of generation compared to the IGA, however when the generation increases, the SGA's performance decreases. The reason for this decreased performance is that when the generation increases, the diversity of the population becomes very small. For the SGA, there is an undistinguishable difference between the maximal and average fitness values.

However, the IGA preserves the population diversity successfully, and sustains the convergence capacity. For example, the average fitness ratio for the IGA is 0.84698 and 0.92065 for the SGA in Fig. 3 (a). Figure 3 (b) indicates that the average fitness ratios of the SGA before and after 200 generations, are 0.91812 and 0.98531, respectively; for the IGA, they are 0.76183 and 0.85121. Figures 3 (c) and 3 (d) also show that the IGA has a lower fitness ratio than the SGA, indicating that the IGA can maintain population diversity and finally achieve global optimization.

Table 4 The lowest energies found for $(C_{60})_N$ clusters ($N \leq 25$).

N	E(eV)	N	E(eV)	N	E(eV)	N	E(eV)
2	-0.2779	8	-4.78772	14	-11.0266	20	-18.7645
3	-0.83369	9	-6.48295	15	-12.847	21	-20.0853
4	-1.66737	10	-7.63607	16	-15.0583	22	-21.9874
5	-2.50106	11	-8.48586	17	-16.418	23	-22.6695
6	-3.61264	12	-9.34378	18	-17.8365	24	-24.5016
7	-4.7446	13	-10.7601	19	-18.3284	25	-25.145

Fig. 4 Lowest-energy structures of $(C_{60})_N$ clusters ($N = 3-25$), calculated by first-principles PR potential.

Table 5 Lowest-energy structures of $(C_{60})_N$ clusters ($N \leq 25$).

N	structure	N	structure	N	structure	N	structure
2	line	8	layer	14	layer	20	layer
3	triangle	9	layer	15	layer	21	shell
4	tetrahedron	10	layer	16	shell	22	shell
5	pentahedron	11	layer	17	shell	23	shell
6	octahedron	12	layer	18	shell	24	shell
7	decahedron	13	layer	19	layer	25	shell

4.3 Lower-Energy Structure of $(C_{60})_N$ Clusters

The lowest energies of the $(C_{60})_N$ clusters searched by the IGA are tabulated in Table 4, and the structures are illustrated in Fig. 4. Compared to the SGA the IGA can find lower energy structures; for example, the SGA locates the energies -2.25569 and -6.57964 under cluster sizes 5 and 11.

In summary, we draw the following conclusions:

- (1) For $N \leq 15$, the lowest-energy structures are body structure or layer structure.
- (2) $N = 16$ is the first shell structure with an irregular hendecahedron.
- (3) For $16 \leq N \leq 25$, the lowest-energy structures are shell structures.
- (4) The structural transition from shell to layer occurs at $N = 19$ and $N = 20$.

Analysis of the structures with different N values (Fig. 4) revealed the following:

- (1) There are obvious body structures when $N < 8$ (Table 5);
- (2) We obtain the structure for $N = 8$ when there is a trihedron generated from one side of the structure for $N = 7$;
- (3) The structure for $N = 9$ is gained while two trihedrons are produced from two neighboring sides of the structure at $N = 7$, respectively;
- (4) We obtain the structure for $N = 10$ if a tetrahedron grows from the structure of $N = 9$, and a new tetrahedron continues to grow, resulting in the structure for $N = 11$;

- (5) There is a small-scale reconstruction for $N = 12$, with a pentahedron emerging from the structure at $N = 11$ simultaneously.
- (6) The structure for $N = 13$ is achieved with a new pentahedron generated from the structure at $N = 12$;
- (7) The structure for $N = 14$ can be achieved if a trihedron emerges from the structure for $N = 13$; then, a new tetrahedron continues to grow so we obtain the structure for $N = 15$;
- (8) Reconstruction occurs when $N = 16$ with the appearance of two quadrilateral surfaces;
- (9) We obtain the structure for $N = 17$ with a tetrahedron emerging from the lower surface (quadrilateral) of the structure at $N = 16$; moreover, the structure for $N = 18$ is achieved if a new tetrahedron continues to grow from another quadrilateral surface;
- (10) Another reconstruction, with layer structure characteristics, occurs for $N = 19$ and $N = 20$;
- (11) The structure reverts to a shell structure when $N = 21$;
- (12) The structure for $N = 22$ is obtained when a tetrahedron arises from a quadrilateral surface of the structure at $N = 21$;
- (13) The structure for $N = 24$ is obtained with a tetrahedron generated from the structure at $N = 23$, and we obtain the structure for $N = 25$ with a new tetrahedron continuing to emerge.

4.4 Magic Number Analysis

The relative stability of a cluster with N molecules is commonly expressed by the second differential of the energy,

$$\Delta_2 E_N = E_{N+1} + E_{N-1} - 2E_N \quad (8)$$

Where E_{N+1} , E_{N-1} , E_N are the energies of the $N + 1$ molecules, $N - 1$ molecules and N molecules clusters, respectively.

The higher $\Delta_2 E_N$, the more stable the structure of the

Fig. 5 The second energy difference of the lowest-energy $(C_{60})_N$ clusters. Peaks indicate a lower energy cluster which is more stable compared to adjacent size and may correspond to magic number.

cluster. A “magic number” cluster has a large peak in the plot of $\Delta_2 E_N$ because a magic number cluster is very stable. Figure 5 is the $\Delta_2 E$ plot of $(C_{60})_N$ clusters. The plot demonstrates that the prominent $\Delta_2 E_N$ peaks occur at $N = 7, 13, 22$, and that there are three weak peaks at $N = 9, 16, 18$; thereby showing that, especially at $N = 13$ and 22 , these clusters are particularly stable. The numbers $N = 7, 13, 22$ are magic numbers.

5. Conclusions

The drawback of the SGA is the high number of generations required to locate the best fitness value. In this work, we developed an improved GA (with new crossover and mutation operations described above) that is effective to solve lower-energy structures of $(C_{60})_N$ clusters. These operations, taken together, can traverse the search space and produce good results. Experiments showed that the obtained prominent peaks in $\Delta_2 E$ are the same as in the work of Rey *et al.* and Doye *et al.*

The IGA was improved in terms of both accuracy and efficiency. In particular, the time cost of the genetic algorithm was further reduced by carefully design, e.g., utilizing orthogonal testing or dynamic coding. In the future we plan to analyze the lower energy structure of metal mixed clusters, i.e., nanoalloy clusters with $N > 10000$.

Acknowledgments

This work was supported by NSFC under Grant No. 51271156. It was also financially supported by the National Defense Basic Scientific Research program of China under Grant No.B1420110155 and the Natural Science Foundation of Fujian Province of China (No.2012J05125 and No.2011J01373), the Foundation of Key Laboratory of System Control and Information Processing, Ministry of Education, P.R. China under Grant No. SCIP2011004.

References

- [1] L. Yufen, Z. Jun, et al., “Structures of odd-numbered pseudo-fullerene clusters: Genetic algorithm and density functional studie,” J. Fudan University (Natural Science), vol.42, no.6, pp.833–842, 2003.
- [2] S.C. Jung and M.H. Kang, “Cluster structure of the C_{60} hexagonal monolayer: Density-functional theory calculations,” J. Korean Physical Society, vol.53, no.2, pp.986–989, 2008.
- [3] S. Kurosu, T. Fukuda, Y. Shibuya, and T. Maekawa, “Formation and reinforcement of clusters composed of C_{60} molecules,” Nanoscale Res. Lett., vol.6, no.1, pp.80–85, 2011.
- [4] D. Sung and N. Park, “Effects of encapsulated metallofullerene on the fermi level alignment at the metal-nanotube interface,” J. Korean Phys. Soc., vol.50, no.3, pp.708–711, 2007.
- [5] C. Wensheng, F. Yan, et al., “Global optimization of $(C_{60})_N$ molecular clusters,” Chemical Physics Letters, 359, pp.27–34, 2002.
- [6] L. Ren and L.J. Cheng, “Structural prediction of $(Au_{20})_N$ ($N = 2-40$) clusters and their building-up principle,” Computational and Theoretical Chemistry, vol.984, pp.142–147, 2012.
- [7] S. Darby, et al., “Theoretical study of Cu-Au nanoalloy clusters using a genetic algorithm,” J. Chem. Phys., vol.116, no.4, pp.1536–1550, 2002.

- [8] J.P.K. Doye and D.J. Wales, "The structure of $(C_{60})_N$ clusters," Chem. Phys. Lett., vol.262, no.1, pp.167–174, 1996.
- [9] C. Rey, J. García-Rodeja, and L.J. Gallego, "Clusters and layers of C_{60} molecules supported on a graphite substrate," Phys. Rev. B, vol.55, pp.7190–7197, 1997.
- [10] J.P.K. Doye, D.J. Wales, W. Branz, and F. Calvo, "Modeling the structure of clusters of C_{60} molecules," Phys Rev. B, vol.64, 235409, 2001.
- [11] Y.H. Luo, J.J. Zhao, and G.H. Wang, "Genetic-algorithm prediction of the magic-number structure of $(C_{60})_N$ clusters with a first principles interaction potential," Phys. Rev., B59, 14903, 1999.
- [12] Z. Wingham, L. Lei, et al., "Lowest-energy structure of $(C_{60})_n$ clusters and thermal effects," Phys. Rev. B, vol.62, no.12, pp.8276–8280, 2000.
- [13] L.J. Cheng, W.S. Cai, and X.G. Shao, "Geometry optimization and conformational analysis of $(C_{60})_N$ clusters using a dynamic lattice searching method," Chem Phys Chem, vol.6, pp.261–266, 2005.
- [14] J.H. Holland, Adaptation in natural and artificial systems, MIT Press, 1975.
- [15] A. Hassani and J. Treijs, "An overview of standard and parallel genetic algorithms," IDT Workshop on Interesting Results in Computer Science and Engineering, Mälardalen University, pp.1–7, Sweden, 2009.
- [16] J.M. Pacheco and J.P.P. Ramalho, "First principles determination of the dispersion interaction between fullerenes and their intermolecular potential," Phys. Rev. Lett., vol.79, pp.3873–3876, 1997.
- [17] M. Hasegawa and K. Nishidate, "Semiempirical approach to the energetics of interlayer binding in graphite," Phys. Rev. B, vol.70, 205431, 2004.

Tingna Wang is currently pursuing her Master's degree at the Institute for Pattern Recognition and Intelligence System, Xiamen University, Xiamen City, China. Her research interests involve pattern recognition, image processing, and evolutionary computation.

Yuhua Wen received the B.Sc. degree from Department of Hydraulics & Pneumatics Engineering, Lanzhou University of Technology, Lanzhou, China in 1996. He received the M.Sc. degree from Department of Space Engineering and Mechanics, Harbin Institute of Technology, Harbin, China in 1998. He received the Ph.D degrees from Institute of Mechanics, Chinese Academy of Science, Beijing, PRC in 2001. He researched as a Postdoctoral Research Fellow from Department of Physics, Tsinghua University, PRC in 2003. He is currently a professor with the Physics Department of Xiamen University, PRC. His research interests include Nanostructures, Molecular dynamics simulations & first-principles calculations.

Guifang Shao received the B. Sc., M. Sc., and Ph. D. degrees from Chongqing University, PRC in 2000, 2003, and 2007, respectively, all in control theory and control engineering. She is currently an associate professor with the Institute of Pattern Recognition and Intelligence System, Xiamen University, PRC. Her research interests include robot control, DNA microarray image processing, and pattern recognition.

Wupeng Hong is currently pursuing his Master's degree at the Institute for Pattern Recognition and Intelligence System, Xiamen University, Xiamen City, China. His research interests involve pattern recognition, image processing, and evolutionary computation.