The role of local optimizations in evolutionary process of atomic clusters modeling

Artur Stasikowski[*], Jerzy Czerbniak, Tomasz Gwizdałła

Department of Solid State Physics, University of Łódź,
Pomorska 149/153, 90-236 Łódź, Poland

Abstract

The application of genetic algorithms in a physical problem of modeling of isolated atomic clusters is the topic of our research. Evolutionary algorithms are a mechanism of the global optimization learning about the solution of the search space. This mechanism plays the role of giving the candidates to global optima. We can use the local optimization in the evolutionary process to improve the efficiency of our algorithm. The goal of our work is evaluation of the influence of the local optimization methods (type of simple gradient) on the growing of the efficiency and accuracy of the evolutionary process in optimization of atomic clusters modeling.

1. Introduction

The global optimization (GO) is a very important problem in various fields of science and engineering. For this reason, many scientists work on creating and improving the methods of GO. Among other problems in physics we deal with global optimization when it comes to modeling of atomic clusters. The cluster is the isolated atomic or molecular system containing up to several thousands of atoms. Modeling of clusters consists of forming the cluster shape in order to find the high symmetry and stability. In order to get the minimal bonding energy, the modelling process is reduced to appropriate arrangement of atoms in the cubic cell.

The task of minimizing energy of atomic structures is very difficult because the potential energy surface (PES) has many deep and short peaks and above this, the number of local structures increases exponentially along with the increase of cluster size. For this reason finding global minima becomes not trivial (NP-hard) problem especially for quite large systems. There are three most popular and most efficient methods of GO which are used to model atomic clusters: basin-hopping algorithm [1] (which uses energy minimization and

---

[*]Corresponding author: *e-mail address*: a.s@poczta.fm

Monte-Carlo simulation), conformational space annealing [2] (which is extension of simulated annealing) and evolutionary algorithms [3] which are one of the methods of artificial intelligence and which we used in our work.

2. Method

The evolutionary algorithm (EA) is the process based on the mechanism of biological evolution: natural selection and heredity. The algorithm works in discreet time. In each time unit, in some environment there exists population of the individuals from same species which compete with each other and can cross freely in the whole population.

The basic idea of GA is using the evolutionary rule concerning the survival of the fittest individuals. That means that better individuals have a greater chance to survive and give offspring. The worse individuals can also survive and give offspring, but probability of this is significantly smaller. This mechanism is carried out in the reproduction method.

The crossover comes after the reproduction phase, this process is equivalent to natural exchange of genetic material, where descendant inherits part of genetic material from the first, and remaining part from the second parent. The second process is mute, which can occur simultaneously to crossover, where a change of gene is random. Reproduction, crossover and mutation are evolutionary operators which are the main part of elementary evolutionary algorithm.

The evolutionary algorithm method is used by us in the physical problem of modeling of atomic cluster. The first problem is the choice of the suitable fitness function. In our problem we selected the fitness function in the exponential form [4]:

$$f\left(U\right)=e^{-b\frac{U-U_{\min}}{U_{\max}-U_{\min}}},\tag{1}$$

where: U is the binding energy of cluster, U_{min} and U_{max} are the lowest and the highest total energy in population, and parameter b determines the selectivity of the fitness function.

The simplest way to determine binding energy is the calculate ion of the sum of two-body interaction energies. We selected the 2-body Lennard-Jones(L-J) [5] potential which was created to describe the interaction between noble gas atoms. This potential has a sharp shape of its function, for this reason L-J is a very popular benchmark for the optimization method.

$$\varphi\left(r\right)=4\varepsilon\left(\left(\frac{\sigma}{r}\right)^{12}-\left(\frac{\sigma}{r}\right)^{6}\right),\tag{2}$$

where: ε is the depth of the potential energy minimum and $\sigma*2^{1/6}$ is the equilibrium pair separation.

In previous work [6] we carried out a test of efficiency of elementary genetic algorithm in the atomic systems described by L-J potential. The system of six atoms has been chosen for the tests, each atom described by three cartesian coordinates, so our EA optimizes the 18-parameter function. The optimal cluster for the potential has used the octahedron structure. EA reaches global minimum in almost 4.6% of independent runs. In other cases calculations stop at the local value of energy corresponding to the atom arrangement in the bicapped tetrahedron. This problem is connected with the shape of L-J potential. Sharper interaction between pairs of atoms determine more deeps(local solutions). At this point we deal with the so-called deceptive problem, that means a premature convergence. We used a few modifications in genetic algorithm to overcome a problem of premature convergence [3,6,7].

After many tests of modified EA, which was used in optimization of 6 atom system described by the Lennard-Jones potential, we observed higher efficiency in comparison to the initial, elementary form of EA [6]. After appropriate combination of different modification methods, we obtained significant increase of efficiency over 10%.

We also used methods based on the knowledge of the problem. In this case it could be sorting of coordinates in each cluster in the crossover operation. Sorting operation is based on arranging the elements of code sequence (coordinate of the position of atoms) according to one of the coordinates in the space of solution. We also used rotation of clusters according to 3 random Euler angles. This operator is used in relation to one individual from each pair taking part in crossing.

Sorting as well as rotation combined with the previous genetic algorithm modifications increase efficiency of finding GM even up to 50%.

3. The basin of attraction of global optimum

The evolutionary algorithms are the methods of global optimization which were created in order to find candidates of global optimum. Determining exact value of global optimum of fitness function in EA is difficult and the time-consuming process. Hence methods of local optimization (LO) should take part in the process of evolution which increases precision of the calculation. We can see that the LO methods in the evolutionary process are important to find the global optimum.

In the evolutionary process the population of solutions often includes individuals located the in so-called basin of attraction of global optimum, but due to a small value of fitness often does not go to the next generation. Consequently, that evolutionary process loses good candidates on the global optimum. This situation is clearly shown in the Figs. 1,2,3,4.
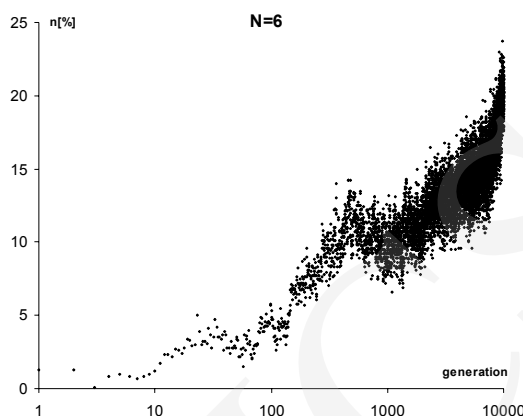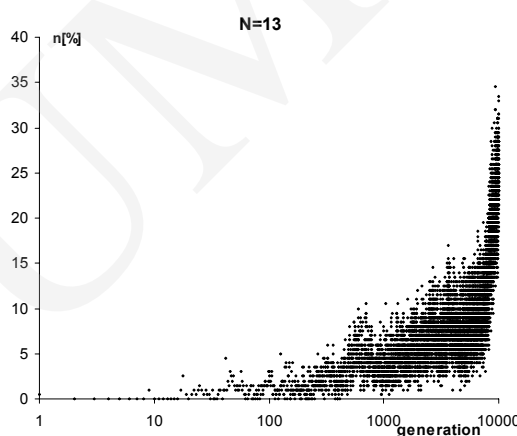
Fig. 1. Evolutionary process of modeling cluster size N=6

Fig. 2. Evolutionary process of modeling cluster size N=13

There are presented the plots of percentage quantity *n* of solution located in the basin of attraction of global optimum in the whole process of evolution, for several atomic systems. We chose typical octahedron structure (N=6, Fig. 1), tested in the previous work [2]. We also carried out our test on two very symmetric and stable structures: N=13 (Fig. 2) and N=19 (Fig. 3) (so called magic numbers [8]). The last one is the large system of atoms (N=26, Fig. 4). That is non symmetric cluster where finding of global optima is very difficult.
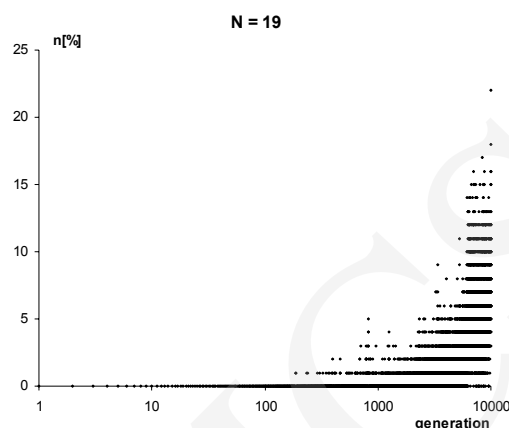
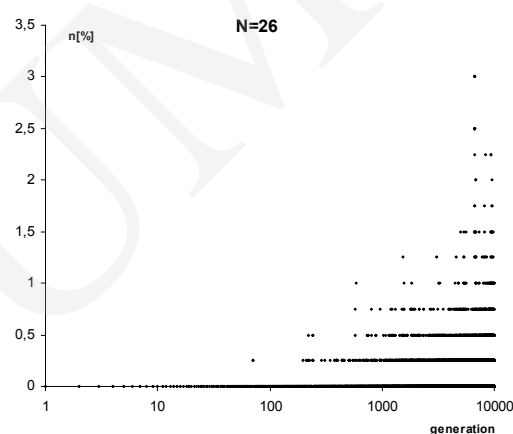Fig. 3. Evolutionary process of modeling cluster size N=19

Fig. 4. Evolutionary process of modeling cluster size N=26

As follows from this diagram the candidates for the global optima appear very early and the number of these individuals increases during the evolutionary process. In the next section the evolutionary modeling of atomic systems with local optimization is discussed.

4. Local optimization in evolutionary process

In the paper the method of Steepest Descent (SD) [9] in the evolutionary process is used. This is the algorithm employed to find the nearest local minimum of a function which assumes that the gradient of the function can be computed. The method of the steepest descent called the gradient descent method, starts at a point P_0 (in search space) and, as many times as needed,

moves from P_{i-1} to P_i by minimizing the local downhill gradient along the line extending from P_i in the direction of $-\nabla f(P_i)$.

SD is one of the oldest and simplest methods. It is rarely used in present-day calculations. However, it is the theoretical basis of modern methods (e.g., Conjugate Gradient, Newton, qNewton). We chose the SD method for the time computing reason.

In our computation the SD finds the global minima of the fitness function but individuals are changed on the genotype level. In the process of evolution we introduce the possibility of Lamarckian evolution type [10]. This evolution provides the changes of features of individual in a course of his life.

The SD calculations may be used in the evolutionary process in a few different ways depending which part of current population is taken to local optimization. The first one is when we choose randomly 50% of 100% of all individuals. In other cases the number of chosen genotypes is equal to 5% or 10% of the whole population size (pop-size), but they are chosen from those individuals which underwent either crossing or mutation earlier. The specific case is here the choice made among those genotypes which did not cross or mute. Let us now denote the index of population (which can be understood as its age) in which the evolutionary algorithm supported by the Steepest Descent local optimization found the global optimum for the first time as k. These values are gathered in Tab.1.

Table 1. Average values of generation *k* with global optimum and time *t* of generation run

	N=6		N=13		N=19		N=26	
	k	*t[s]*	*k*	*t[s]*	*k*	*t[s]*	*k*	*t[s]*
Only local optimization	128	-	70	-	476	-	5012	-
5% pop-size (after crossover)	26	0.25	31	0.73	120	14.2	580	261
5% pop-size (after mute)	27	0.29	40	1.83	146	15.5	644	279
5% pop-size (others)	50	0.29	55.12	1.13	204	15.3	1021	291
10% pop-size (after crossover)	16	0.13	22.3	1.08	86	19	328	199
10% pop-size (after mute)	13	0.09	24	1.7	96	19.8	344	263.2
10% pop-size (others)	32	0.36	40	1.39	144	19.92	950	480
50% pop-size	5.54	0.11	8.61	1.07	25	7.47	85	209
100% pop-size	4.31	0.10	5.34	0.67	17.63	8.74	43.7	63.3

Next to them the time of optimization obtained on the *CPU Athlon 1,66 MHz* unit is presented. Only the first row of table where the results of SD calculations in the independent runs are shown, is of special importance. The number here indicates that every k-th of run the local optimization gives the global optimum when the initial atoms positions are randomly sampled in the whole cell volume.

It means that the inverse of this number is the probability of finding the global minimum in random search and the quotient of search space volume and this number are the size of basin of attraction of global minimum.

5. Conclusions and discussion

As follows from Table 1 random search of the global optimum using only a local method is possible for small systems. With the increase of the number of atoms, the number of local deeps also increases rapidly. Therefore for the bigger systems such an attempt is not only ineffective but even impossible.

In the evolutionary process the local optimization (LO) has been used to individuals as well to take part in recombination of the other ones. In the first case LO leads to the effect that the global optimum may be found in the earlier generations. It turns out however that the time of performing calculations is comparable. It comes from the different time needed to make SD minimization. This is obvious because after recombination all genotypes "jump" into other regions of search space and again need relaxation through the methods of local optimization. For the individuals which did not take part in recombination, it is possible that they have been already optimized in earlier generations.

With the increasing number of individuals undergoing the SD procedure, the index of generation k when the global optimum is found decreases. It is distinctly seen especially for large atomic systems. It is obvious because increasing the number of locally optimized genotypes, the chance to find the globally best solution is increased. However, the average time needed to find this solution decreases with the increasing number of genotypes optimized locally.

Concluding we can say that in the evolutionary modelling of atomic clusters the use of local optimization to all individuals in population is the most effective way. It is a little bit surprising because it is not compatible with widely spread theories [10] which assume that if someone wants to keep the population sufficiently diverse one has to use LO to approximately 5% of individuals in population.

References

[1] Wales D.J., Doye J.P.K., *Global optimization by basin-hopping and the lowest energy structures of Lennard-Jones clusters containing up to 100 Atoms*, J. Phys. Chem. A101 (1997) 5111.
[2] Lee J., Lee I.H., Lee J., *Unbiased global optimization of Lennard-Jones clusters for N < or =201 using the conformational space annealing method*, Phys Rev Lett. 22;91(8):080201 (2003).
[3] Michalewicz Z., *Genetics Algorithms + Data Structures = Evolution*, Programs Springer-Verlag Berlin Heidelberg, (1996).
[4] Roberts Ch., Johnston R.L., Wilson N.T., *A genetic algorithm for the structural optimization of Morse clusters*, Theor. Chem. Acc. (2000) 104, 123.
[5] Jones J.E., Ingham A.E., Proc. R. R. Soc. London, Ser., A107 (1925) 636.

[6] Stasikowski A., Czerbniak J., Gwizdałła T., *The techniques of evolutionary optimization improvement – application to atomic clusters*, III Sympozjum Modelowanie i Symulacja Komputerowa w Technice, Łódź, (2004).

[7] Goldberg D.E., Genetic *Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading MA, (1989).

[8] Doye J.P.K., Wales D.J., *Magic numbers and growth sequences of small face-centered-cubic and decahedral clusters*, Chemical Physics Letters, 247 (1995) 339.

[9] Arfken, G., *The Method of Steepest Descents*, §7.4 in *Mathematical Methods for Physicists*, 3rd ed. Orlando, FL: Academic Press, (1985) 428.

[10] Whitley D., Gordon V.S., Mathias K., *Lamarckian Evolution, the Baldwin Effect and Function optimization*, PPSN'94, (1994) 46.