An Improved Genetic Algorithm for Global Optimization and Its Application to Sodium Chloride Clusters

H. Kabrede and R. Hentschke*

FB Physik, Bergische Universität, D-42097 Wuppertal, Germany

Received: May 15, 2002; In Final Form: August 2, 2002

We have developed an improved genetic algorithm with a self-guiding search strategy, using a combination of "traditional" and geometric genetic operators, whose relative weights are determined by the algorithm itself. Explicit mutation becomes unnecessary, because randomness is included in some of the geometric operators. This still quite simple algorithm is applied to the search for global minima on the potential energy hyper-surface of the most widely studied ionic clusters, sodium chloride clusters, with up to 100 ions. We present 32 previously unpublished apparent global minima of the Coulomb + Born−Meyer potential in the Tosi Fumi parametrization. Comparison with already published minima of the shell model due to Rittner shows no structural differences. We are able to reproduce the experimental sodium chloride bulk energy quite closely. Therefore, the Coulomb + Born−Meyer potential appears to be a good approximation for neutral clusters of all sizes.

1. Introduction

The structure of sodium chloride crystals is well-known. In a simple cubic lattice, sodium and chloride ions are placed at alternate positions. However, what is the structure of a sodium chloride cluster of given size, and how do clusters develop into the bulk structure with increasing size?

In general, it is a difficult task to find a cluster's global minimum structure. Even for a simple pair potential which only takes the two major interaction effects into account, i.e., Coulomb interaction and the repulsion of the ion cores due to the exclusion principle at short distances, the number of local minima on the potential energy hyper-surface should be growing exponentially with increasing cluster size.[1] Therefore, a systematic search is not feasible.

Different approaches have been applied to determine the global minima of sodium chloride clusters. The first publications dealing with small sodium chloride clusters[2,3] use intuitive guesses to generate structures likely to be in the vicinity of the global minimum. Subsequently, these structures are reformed by varying the ion coordinates. In a later approach by Phillips et al.,[4] ion configurations are generated using a build-up procedure. The ions are added one at a time using a Metropolis-like criterion deciding whether to accept or to reject the newly placed ion. Another alternative[5] is to generate cluster configurations with a similar build-up procedure for further minimization of the energy. One cluster ion is moved by a fraction of its nearest neighbor distance along the direction of the force acting on it. This procedure is applied to every cluster ion until no lower energy can be achieved. Subsequently, it is repeated with a reduced fraction of the nearest neighbor distance. Still other alternatives are molecular dynamics simulated annealing and adiabatic Gaussian density annealing, which have been used to determine the global minima of sodium chloride clusters consisting of no more than 27 ions.[6] In a recent reference, a global optimization technique called Basin Hopping or Monte Carlo minimization is applied with good success to $(NaCl)_N Cl^{\ominus}$ clusters.[7] Alternatively, a basic genetic algorithm was applied to this problem by Michaelian,[8] which yields the correct results for small clusters of up to 6 ion pairs.

In this work, we use an improved genetic algorithm utilizing techniques from the study of atomic clusters such as Lennard-Jones clusters.[9−11] A genetic algorithm is a search strategy applying methods based on our understanding of natural evolution.[12] It operates on the individual clusters which form the population. Each cluster in the population is represented usually by just one chromosome, which is an array containing the structural information necessary to construct this particular cluster. One problem is the suitable coding of this information. Suitable means that the so-called genetic operators can be applied to the chromosomes and thus will produce a new generation of clusters with improved overall fitness. The fitness function is based on the quantity which one intends to optimize, i.e., the potential energy in our case. There are certain "standard" operator categories such as crossover, mutation, and reproduction. However, operator design, their weighting, or the order in which they are applied are by no means straightforward, and this is why genetic algorithms are of interest by themselves.

A simple example of the genetic algorithm developed and tested in this work starts from a randomly generated initial population. We use the Cartesian coordinates of the ions as a natural coding, and each cluster's fitness is computed based on its potential energy. After the fitness has been assigned to each cluster, we mutate (cf. below) a small fraction of the population to allow the algorithm to explore the entire search space. The next step is to create a new generation. At this point, additional genetic operators are used to combine the information of the individual chromosomes, which are chosen for reproduction based on their fitness, to form new individual chromosomes representing other points in search space. This procedure is repeated for many generations until the global minimum or at least a very good local minimum has been found.

In the following sections II and III, we discuss the above algorithm in detail. In section IV, the algorithm is applied to clusters of sodium chloride with up to 100 ions.

* To whom correspondence should be addressed.

10.1021/jp026114+ CCC: $22.00
Published on Web 09/04/2002

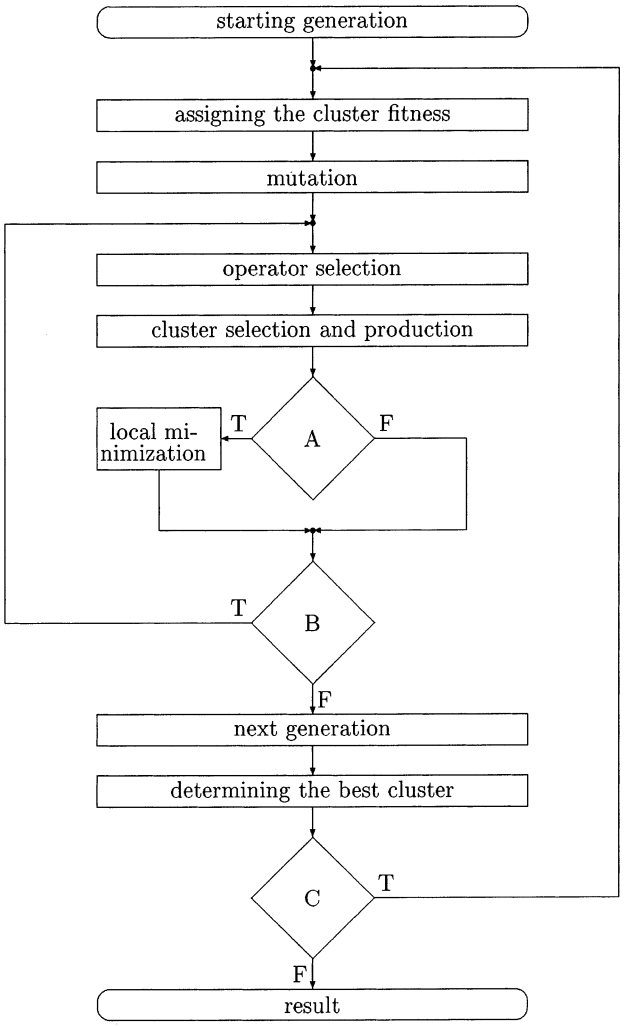

Figure 1. Flowchart of the genetic algorithm. A: $E < 0.85\ E_{opt}$. B: $N_{new} < N_{pop}$. C: $E_{opt,i} \neq E_{opt,i-500}$.

2. Method

A frequently used and sufficiently simple approximation of the alkali halide potential energy is the Coulomb + Born−Meyer potential:

$$V_{ij} = \frac{q_i q_j}{|\vec{r}_{ij}|} + A_{ij} \exp\left[-\frac{|\vec{r}_{ij}|}{\rho}\right] \tag{1}$$

Here q_i is the charge of ion i, and $|\vec{r}_{ij}|$ denotes the distance between ions i and j. Tosi and Fumi[13] fitted the parameters A_{ij} (A_{++}, A_{--}, $A_{+-} = A_{-+}$) and ρ to crystal data. We use their parameter values to calculate the potential energy of the neutral sodium chloride clusters. Note that taking the ion polarization into account, like in the shell model due to Rittner,[14] is computationally more expensive. Even though we do not use this model here, we do compare our results to corresponding results obtained with this model.

In the following, we discuss the genetic algorithm for finding the global minima of neutral sodium chloride clusters. Figure 1 shows a flowchart representation of the algorithm.

Starting Generation. A starting generation consisting of a certain number N_{pop} (10 in our case) of neutral clusters is generated. The Cartesian coordinates of each cluster ion are chosen randomly in a cubic box with length $L = 3(N)^{1/3}$, where N is the number of ions per cluster.

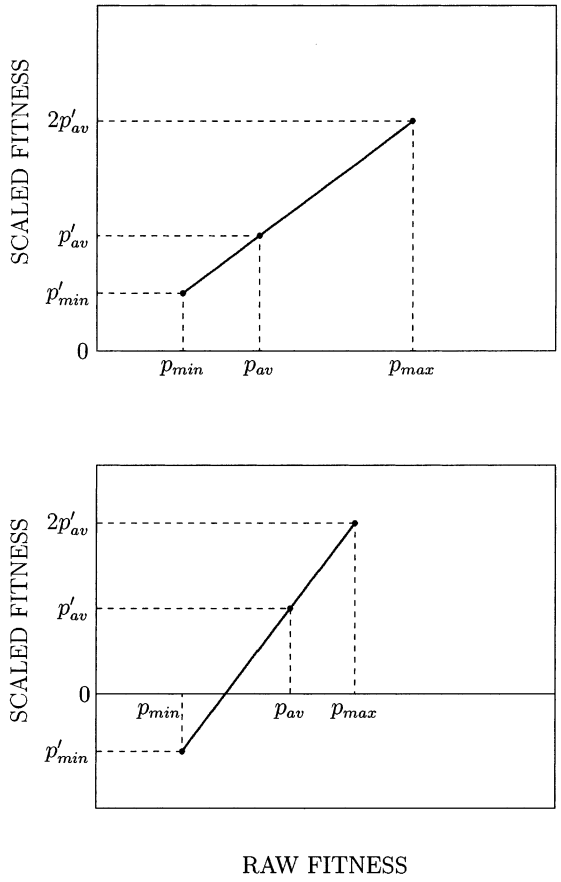

Figure 2. Top: Illustration of the normal scaling of the raw fitness p_i. Here the condition $p_{min} \geq (p_{max} - \lambda p_{av})/(1 - \lambda)$ is met ($\lambda = 2$ in this case). One can scale the raw fitness to the desired value λp_{av} without p'_{min} being negative. Bottom: Illustration of the scaling if the condition is not met. Here p'_{min} would be negative.

Assigning the Cluster Fitness. In this algorithm, fitness is the probability of being selected for the creation of a new cluster. Let f_i be the potential energy of cluster i if it is negative, zero if not. The raw fitness is then defined as

$$p_i := f_i / \sum_{i=1}^{N_{pop}} f_i$$

We apply linear scaling,[12] as illustrated in Figure 2, to this raw fitness, i.e.

$$p_i' = a p_i + b \tag{2}$$

Here p_i' is the scaled fitness, and a and b are parameters. Linear scaling of the raw fitness p_i is useful at the beginning, when one good individual dominates the entire population or when all clusters of the population have similar fitness. Scaling the fitness ensures constant working of the selection mechanism.

The parameters a and b in eq 2 are subject to the following conditions: the average of the raw fitness should be conserved

$$p_{av} = \frac{1}{N_{pop}} \sum_{i=1}^{N_{pop}} p_i$$

i.e., $p'_{av} = p_{av}$ and of course $p_i' \geq 0$. If the condition $p_{min} \geq (p_{max} - \lambda p_{av})/(1 - \lambda)$ is met, one can adjust a and b to yield $p'_{max} = \lambda p_{av}$. Here, λ is the average number of the best individual's copies per generation. For a small population below

TABLE 1: Final Selection Probabilities for the Seven Genetic Operators[a]

		genetic operators	
1	arithmetic mean:	$2 \rightarrow 1$	Each new ion coordinate is the arithmetic mean of two parent ion coordinates.
2	rotation I:	$1 \rightarrow 1$	The ions are sorted according to the angle between their position vector and the z axis. The first half of them is rotated by a random angle about the z axis.
3	rotation II:	$1 \rightarrow 1$	The ions are sorted according to their distance from the origin. The first half of them is rotated by a random angle with respect to the origin.
4	reflection:	$1 \rightarrow 2$	The ions are sorted according to their z coordinate. The first half of them is reflected at the $y-z$ plane for one cluster and at the $x-z$ plane for the other cluster.
5	crossover I:	$2 \rightarrow 2$	Two chromosomes, i.e., linear arrays containing the ion coordinates, are cut in their center and fused crosswise.
6	crossover II:	$2 \rightarrow 2$	The ions are sorted according to their distance to the origin. Subsequently, crossover I is applied.
7	crossover III:	$2 \rightarrow 2$	The cluster ions are sorted according to the angle between their position vector and the z axis. Subsequently, crossover I is applied.

[a] For every operation, each cluster's center of mass is the origin.

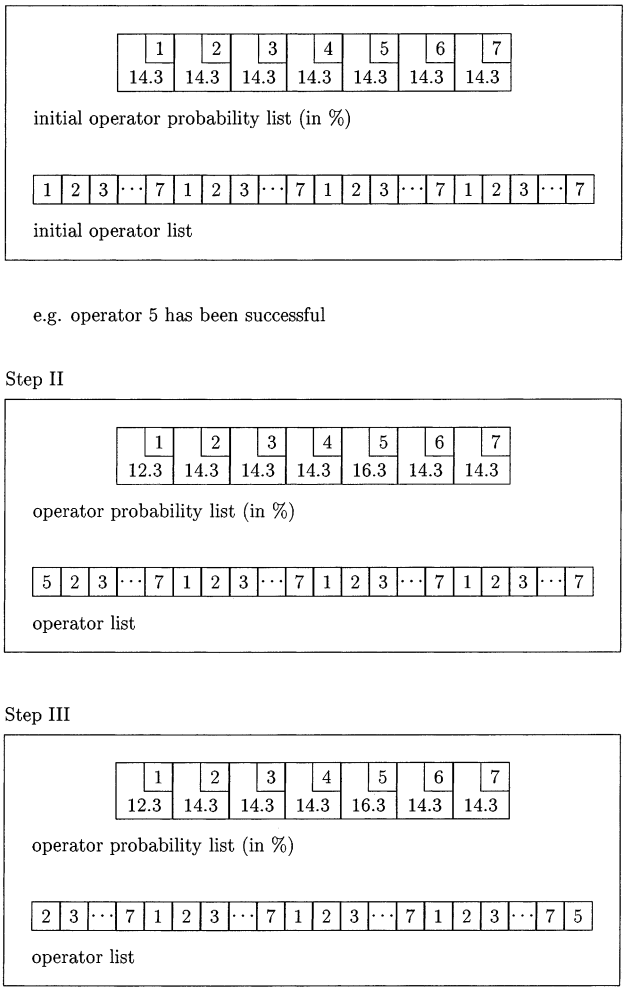

Figure 3. Illustration of the dynamical evolution of the operator selection probability explained in the text.

20 individuals, a value of λ between 1.2 and 2 has proven to be most advantageous[12] (in Figure 2 the value of λ is 2). Thus

$$a = (\lambda - 1)p_{av}/(p_{max} - p_{av})$$

$$b = (p_{max} - \lambda p_{av})p_{av}/(p_{max} - p_{av})$$

If the above condition is not met, one can only scale as much as possible with $p'_{min} = 0$ instead of $p'_{max} = \lambda p_{av}$. Thus

$$a = p_{av}/(p_{av} - p_{min})$$

$$b = -p_{min}p_{av}/(p_{av} - p_{min})$$

Mutation. A small fraction, given by the mutation rate (usually $< 10\%$), of the population's individuals is mutated. This means that every coordinate of the randomly selected clusters is translated either by a uniform or a Gaussian random number. As we explain below, this explicit mutation is optional in an algorithm using advanced geometric operators with inherent mutation, like this one.

Operator Selection. Initially, the probability of being selected for all 7 operators (cf. Table 1) is 1/7 (14.3%). During the calculation, however, this probability changes dynamically depending on an operator's success. An operator is called successful if it creates a low lying minimum with energy $E \leq 0.85E_{opt}$ (here 0.85 empirically has proven to be adequate) and $E_{new} \neq E_{old}$. Here E_{opt} is the best energy found so far, E_{new} is the energy of the new cluster(s), and E_{old} is the energy of the parent cluster(s). According to these criteria, the probabilities are updated. Figure 3 illustrates this updating process. Step I is the situation at the beginning of the algorithm. If an operator is successful, this operator replaces the first operator in the operator list. In the example of Figure 3, the operator 5 replaces operator 1. Subsequently, the probability of the successful operator, here operator 5, is raised by 2%, and the probability of the formally first operator, here operator 1, is lowered by 2%. This concludes step II. In the final step, step III, the list is rotated by one element. Note that because the original list consists of a 4-fold repetition of the operators 1−7, the lowest probability an operator can have before it vanishes from the list is 6.3% (i.e., $14.3\% - 4 \times 2\% = 6.3\%$). A vanished operator reappears in the list, after it has been successful. Note that because of the list, an operator will not end up at 0% and, thus, will always be considered. It therefore serves as a device which maintains genetic diversity.

Cluster Selection and Production. Depending on the type of the selected operator, one or two clusters are chosen. To select a cluster j, a random number ξ between 0 and 1 is computed. Subsequently, the cluster fitness p_i' is summed up until

$$\xi \leq \sum_{i=1}^{j} p_i'$$

This selection method is termed roulette wheel selection. One or two new clusters are produced using the operator selected in the previous step.

Local Minimization. If the potential energy of a new cluster is $\leq 0.85E_{opt}$ (here 0.85 is taken as a compromise between speed and effectiveness of the algorithm), local minimization by a conjugate gradient method[15] is applied to the cluster. This is the time-consuming step of this algorithm.

Next Generation. The number of clusters per generation is constant. As can be seen in the flowchart, the steps operator

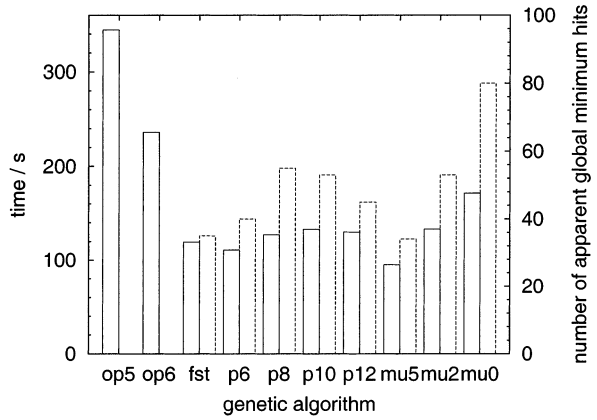

Figure 4. Comparison of 10 variations of the standard genetic algorithm proposed in this work. Average run time (left axis and solid columns) and the number of times the apparent global minimum of a cluster with 78 ions was found during 500 independent runs (right axis and dashed columns).

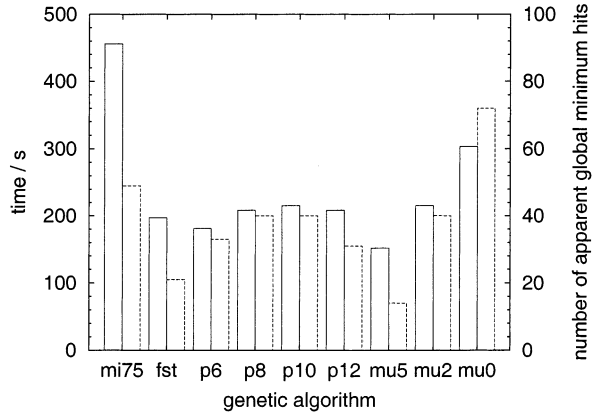

Figure 5. Comparison of nine variations of the standard genetic algorithm proposed in this work. Average run time (left axis and solid columns) and the number of times the apparent global minimum of a cluster with 96 ions was found during 1000 independent runs (right axis and dashed columns).

selection, cluster selection and production, and local minimization are repeated until at least N_{pop} clusters are produced. Initially, a new generation consists of those new clusters whose potential energy $E \leq 0.5E_{opt}$, the best cluster, and a random remainder of the previous generation.

Determining the Best Cluster. At this point, the best cluster found so far is updated. In addition, other useful statistical information and low lying minima are computed and stored as well.

Search Condition (C). Now that the new generation is known, the algorithm returns to the computation of the cluster fitness if the search condition is still met. The search terminates if E_{opt} remains unchanged for a fixed number of generations n_{max} (here 500−2000 generations). This E_{opt} defines what we call the apparent global minimum.

Figures 4 and 5 summarize various performance tests for our algorithm. The standard algorithm, which is applied to a cluster consisting of 78 ions (Figure 4) and to a cluster consisting of 96 ions (Figure 5), uses dynamical weighting of all seven operators, a population size of 10 clusters per generation, a mutation rate of 2%, and local minimization applied to all new clusters that have a potential energy $\leq 0.85E_{opt}$. This standard algorithm is compared to various modifications. We note that in Figure 4 the statistical error for the standard algorithm on the time scale is 2.7%, and for the number of apparent global minum hits, it is 10%. In Figure 4, the genetic algorithms called op5 and op6 use only operator 5 or 6 (cf. Table 1), respectively. The genetic algorithm fst uses all operators used by the standard algorithm but with fixed weights. These weights represent the final weight distribution obtained for the same cluster size with the standard algorithm (cf. Figure 6). The algorithms p6 to p12 correspond to the standard algorithm applied to populations of size 6−12 clusters per generation. mu5, mu2, and mu0 each correspond to the standard algorithm with a mutation rate set to 5%, 2%, and 0%, respectively. In Figure 5, the genetic algorithm mi75, which here replaces the poorly performing algorithms op5 and op6 in Figure 4, applies local minimization if a new cluster's potential energy is $\leq 0.75E_{opt}$. The standard genetic algorithm of course is equivalent to algorithms p10 and mu2 in the two figures.

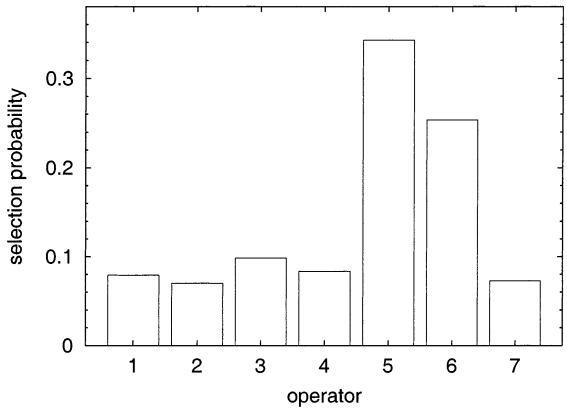

Figure 6. Mean operator selection probability at the end of a run for the operators in Table 1. The figure shows averages based on 100 independent trials for cluster size $N = 80$.

Obviously, the single operator algorithms op5 and op6 perform poorly, even though operators 5 and 6 do possess the highest selection probability in the standard algorithm (cf. Figure 6). In Figure 5, i.e., in the case of the 96-cluster, these are replaced by mi75. Here we see that lowering the threshold for local optimization does increase the average run-time considerably compared to the standard algorithm p10, but it does not increase the number of times the apparent global minimum is found in the same proportion. In Figures 4 and 5, we observe that lowering the mutation rate tends to increase the number of times the apparent global minimum is found. However, simultaneously, the average run time increases. This increase is more pronounced for the larger cluster size. Therefore, the small mutation rate of two percent used in the standard algorithm appears to be a good choice. Regarding the population size we note that there is little difference between 8 and 10 clusters, but there is a slight overall drop in the performance if 6 or 12 clusters are used. We also note that the algorithm fst, which uses a fixed operator selection distribution does not perform too badly. Nevertheless it uses information which has been acquired by the standard algorithm during a previous calculation.

Figure 6 shows the final selection probabilities for the seven genetic operators, which are explained in Table 1, averaged over 100 independent runs. These mean values differ only slightly for different cluster sizes. It is surprising that the simple operator 5 has such a high selection probability. Nevertheless, as mentioned above, using operator 5 or 6 only is not sufficient. Note that inherent mutation is present in operators 2 and 3.

Figure 7 shows a comparison of three different algorithms for global energy minimization of sodium chloride clusters. For $N = 5-20$, the first column represents our standard genetic algorithm as proposed in this work. The second column

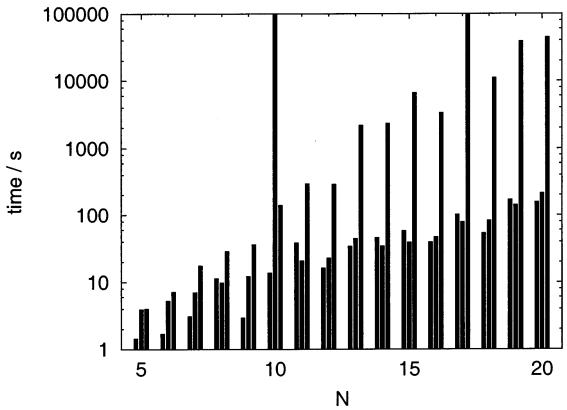

Figure 7. Comparison of three different algorithms for global energy minimization of NaCl clusters as explained in the text.

represents a Monte Carlo basin hopping algorithm as proposed in ref 7. The third column finally represents a naive algorithm, where randomly generated starting configurations are relaxed to their closest local minima. For each N in the displayed range the time is calculated by (total number of trials)×(average time per trial)/(successful trials). For all three algorithms, the first apparent global minima have been found with a probability of nearly 100% per trial. Our standard genetic algorithm and the Monte Carlo basin hopping algorithm succeed similarily well in finding the apparent global minima with just one exception. Monte Carlo basin hopping does not seem to be able to find the global minimum of the $N = 10$ cluster, even after extensive testing. In contrast to our algorithm and the Monte Carlo basin hopping, the naive algorithm performs quite poorly as N is increased.

3. Discussion of the Algorithm

A simple genetic algorithm that is widely used as a basis for improvement is the algorithm by Deaven et al.[10] This algorithm starts with a small randomly seeded generation. Every cluster of a generation is cut into halves by a random plane. These halves are recombined in every possible combination, and the resulting cluster is relaxed to its closest local minimum. The new clusters are allowed to replace inferior parent clusters if their energy is lower than their parent's energy by some amount δE. Occasionally, a cluster is mutated or a cluster, found with a different minimization strategy, replaces one cluster of this new generation.

Our algorithm also uses the above algorithm as a basis, but it is different in three important points. In contrast to the basic algorithm, we use seven different genetic operators to produce new clusters. We combine "traditional" operators acting on strings of real numbers, the geometric operator by Deaven (operator 6 in Figure 6), and four new geometric operators. This is advantageous as Figure 4 shows. The algorithm therefore can produce a much more diverse population in each generation. We adhere to the evolutionary principle that fitness decides which clusters are used for reproduction. Our new generation consists of those clusters that are better than a certain limit and the best individual found so far. Therefore, the average potential energy does not decrease monotonic in our algorithm. Because local minimization is the time-consuming step, we minimize only those clusters that are likely to yield a good structure, i.e., the potential energy fulfills $E \leq 0.85E_{\mathrm{opt}}$. Even though it seems that we might miss a number of good structures, experience shows that we do not, and we gain a significant overall increase in speed.

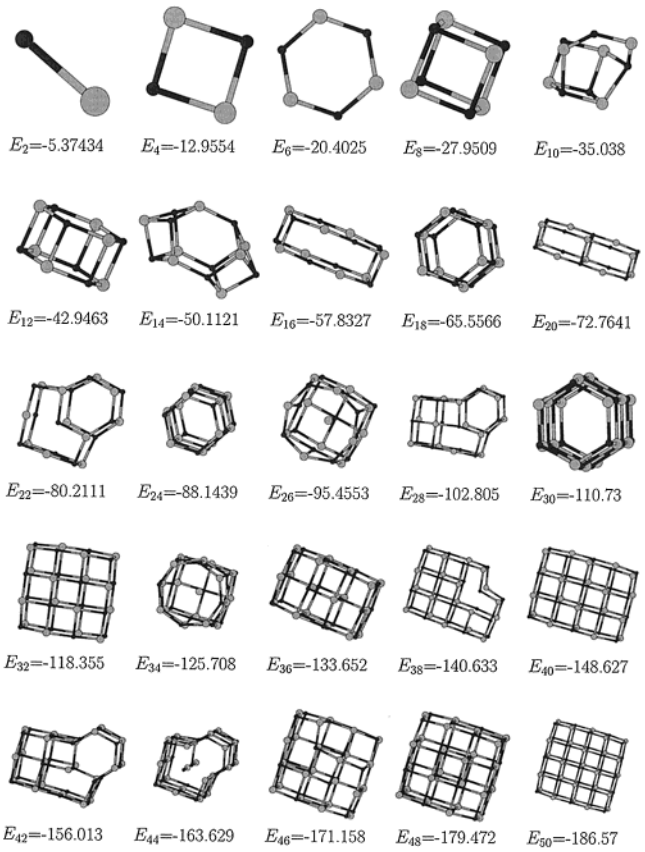

Figure 8. Apparent global minimum structures for the neutral clusters consisting of 2−50 ions. The energies are given in eV.

In contrast to other authors, we do change mainly the production step of the genetic algorithm. Wolf and Landman[16] for example improve the basic algorithm in the mutation step, which we find of no relevance in our algorithm. They use seeded initial configurations and speed up the local minimization in a different way than we do. Hartke[17] improves the basic algorithm by a new selection method. He uses a structure parameter as a second selection criterion to ensure different geometric structures in every new generation. However, our algorithm is self-guiding in the sense that it changes its search strategy while searching, which is found to be very effective (cf. Figures 4 and 5). A similar concept has been used by Zeiri.[11] The algorithm is kept quite simple to make it easy to implement for different applications in contrast to other new genetic algorithms that use specialized and more complicated methods.

4. Sodium Chloride Cluster Structures

We have computed the apparent global minima of neutral sodium chloride clusters consisting of 2 to 100 ions (cf. Figures 8 and 9). The four- and six-ion clusters illustrate the basic structures which form the larger clusters. Among the 50 described clusters, 18 are cuboids (two of them being cubes). Five apparent global minima are cylinders with a hexagonal base. The closest minimum to each of them is a cuboid except for $N = 6$. There is no cuboidal local minimum structure of this size. A mixture of squares and hexagons in the base of a cylinder can be found in seven apparent global minima. Such a mixture is competitive even for clusters consisting of a large number of ions, like $N = 88$.

There is a close relation between stable neutral clusters consisting of an uneven number of ions and those which contain an additional ion.[7] Removing an ion from the face center of

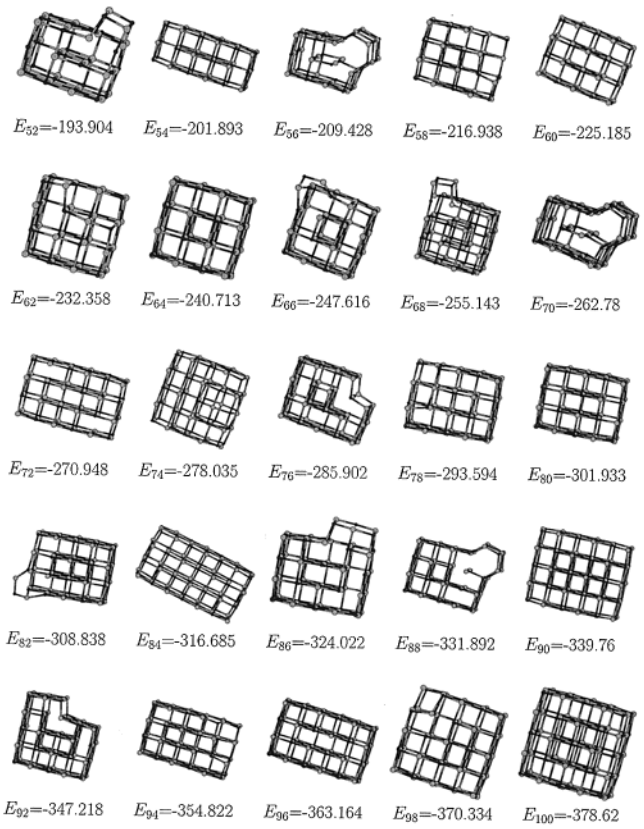

Figure 9. Apparent global minimum structures for the neutral clusters consisting of 52−100 ions. The energies are given in eV.

the 3 × 3 × 3- or stable 27-cluster one obtains the apparent global minimum of the cluster consisting of 26 ions. The next cuboid with an uneven number of ions is a 5 × 3 × 3 structure. If one removes its central ion, one obtains the second lowest minimum of the 44-cluster. The apparent global minimum of the cluster consisting of 74 ions also belongs to this series.

The ions in the corners of the clusters with this vacancy are $Na^{\oplus}$ ions. Exchanging $Na^{\oplus}$ with $Cl^{\ominus}$ in these three clusters, the one consisting of 34 ions and the incomplete and extended cuboids yield different clusters with different energies.

Clusters whose ion number can be written as the product of three quite unequal factors only, e.g., $N = 52$ (2 × 2 × 13), exhibit incomplete or extended cuboidal apparent global minimum structures. This is also true for those clusters which do not fit into another scheme, like the clusters with $N = 22$ and 38 ions. Three clusters remain whose apparent global minima cannot be represented by a simple scheme. These are the clusters with $N = 10$, 14, and 34 ions.

For some cluster sizes, adding an ion pair to their apparent global minimum leads to the succeeding apparent global minimum structure, for example, from $N = 46$ to 48 and from $N = 62$ to 64. In both cases, the latter cluster is especially stable (cf. below). Also adding an ion pair to an especially stable apparent global minimum can lead to the succeeding apparent global minimum structure, e.g., from $N = 64$ to 66. Overall, however, we do not find a simple scheme leading from one especially stable apparent global minimum structure to the next. The corresponding structures are low lying local minima but not the apparent global minima.

The different ionic radii of $Na^{\oplus}$ and $Cl^{\ominus}$ are taken into account by the Born−Meyer parameters (A_{++}, A_{--}, and A_{+-}) of the potential. The angles in the $N = 4$ cluster are not 90° but 88.3° and 91.7° respectively. The angles in the six-ion cluster are not

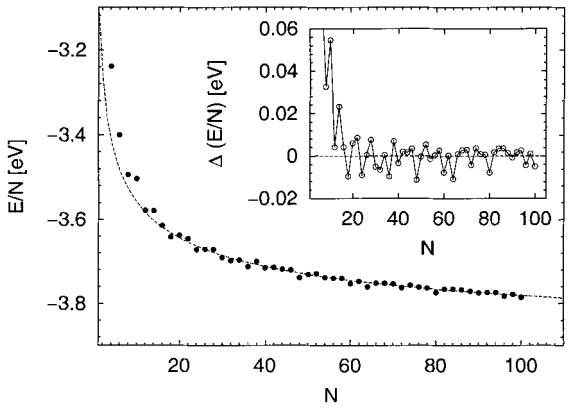

Figure 10. Apparent global minimum energy per ion vs N. The dashed line is a fit using eq 3 in the range of $50 \leq N \leq 100$. The inset shows the difference between the apparent global minimum energy per ion and the fitted curve.

120° but slightly smaller (<1°) in case of the Na−Cl−Na angle and slightly larger in case of the Cl−Na−Cl angle. Although the nearest neighbor distances in these simple clusters are the same in all cases, this is not true for the larger clusters, in particular not for the cuboids. In agreement with ab initio calculations,[20] the distance of the corner ions to their nearest neighbors is smaller than the average nearest neighbor distance. The distance of the ions in the cluster faces to their nearest neighbor ion inside the cluster is larger than the corresponding mean value.

Assuming that the cohesive energy E of a cluster can be written as the difference between a volume term and a surface term, we may write

$$\frac{E}{N} = e_{\mathrm{vol}} - \frac{e_{\mathrm{surf}}}{N^{1/3}} \tag{3}$$

Here e_{vol} and e_{surf} are adjustable parameters. Note that the volume term should be proportional to N and the surface term to $N^{2/3}$. In Figure 10, E/N is plotted versus N. The dashed line is a fit of eq 3 to the cluster data in the range $50 \leq N \leq 100$. The bulk energy per ion obtained from this fit is −3.97 eV. This is in very good agreement with the experimental value of −3.96 eV.[18] Especially stable global minimum structures lie below the fitted curve. In the case of small clusters such as $N = 18$ and 24, this is a cylinder with a hexagonal base. This changes to complete cuboids with a good ratio between volume and surface, i.e., a not too large aspect ratio, in the case of bigger clusters ($N = 30$, 32, 36, 40, 48, 60, 64, 72, 80, and 96).

5. Discussion of the Cluster Structures

Sodium chloride is a commonly used material and the most prominent example of ionic clusters. In 1977, T. P. Martin calculated the structure of small sodium chloride clusters with up to seven ion pairs.[2] His cohesive energy included all Coulomb interactions, whereas in the case of the short ranged Born−Meyer term, he compared two approximations: (a) only nearest neighbor interactions were included; b) in addition to the nearest neighbor interactions, nearest neighbor interactions of the same ion type were included also. He did not find significant differences between the global minima obtained with the two approximations. By variation of the ion coordinates, he got similar results as we do except for the 12-cluster, where he obtained a lower energy for a cuboid structure in comparison to the two-layer ring structure.

Together with J. Diefenbach he extended his former work in 1985[3] and calculated minima of neutral and singly positive charged alkali halides CsF, CsI, NaCl, and NaI using the Coulomb + Born−Meyer potential and the polarizable model due to Rittner. The global minimum structures of the clusters consisting of 2−36 ions agree with our structures except for the 22- and the 28-cluster. However, their method of calculation is based on intuitive guesses of the global minimum structures, and thus, they do not consider the full energy hyper-surface. They find one example for which the sodium chloride apparent global minimum structures of the two potentials are not the same: $N = 22$.

In 1990, Phillips and co-workers computed the global minima of sodium chloride clusters with up to 15 ion pairs,[4] both for neutral clusters and clusters with an excess sodium or chloride ion. Their computed structures agree with ours even though they use the polarizable model due to Rittner. The cohesive energies of their clusters are slightly (≈1−2%) lower than ours, because of the additionally considered dipole moments.[3]

Differences of the two models used for the alkali halides seem to occur for clusters with excess ions only, as can be seen for example in the work of Doye et al.[7] In 10 cases, they find different global minimum structures for the two potentials for $(NaCl)_N Cl^{\ominus}$ in the range $N \leq 35$.

It is not possible to observe small clusters directly with an electron microscope. Experiments can only provide data concerning the clusters relative stabilities and their vibrational frequencies.[19] Ab initio calculations are another means to check the validity of the global minimum structures computed with pair potentials. Note however that the computational effort makes it necessary to restrict the configuration space to the vicinity of a few guessed structures. Ochsenfeld and Ahlrichs report such calculations[20] for clusters consisting of 2−6, 8, 9, 12, 15, 18, and 32 ion pairs. They largely obtain very similar results. For the smallest clusters, the different approaches yield different local minima but have the same global minima. The differences in energy between the local minima are not the same for some cluster sizes. The clusters whose global minimum structures are cylinders of six-ion rings in our calculations favor cuboids in the ab initio calculations of Ochsenfeld and Ahlrichs. These cuboids are the apparent second lowest minima of the pair potential used here.

Ayuela et al.[21] carry out ab initio calculations using two different basis sets. One basis modeling neutral atoms yields good agreement with the results of Ochsenfeld and Ahlrichs. The results of the other basis set modeling free ions are in better agreement with the pair potential results especially for the cylinder of six-ion ring global minimum structures. The clusters missing in Ochsenfeld and Ahlrichs work are computed in this work, too. They are in good accord with our work except for the cases $N = 28$ and 36. These two cluster structures are local minima of the Coulomb + Born−Meyer potential but not the apparent global minima.

6. Conclusion

In this work, we have developed an improved genetic algorithm for global optimization of atomic and ionic clusters. Instead of using just a single genetic operator, we use a variety of traditional and geometric genetic operators. Their relative importance, i.e., their selection probabilities, is variable and determined by a self-guiding strategy based on the success of each operator. We apply local minimization only to those clusters which already have a low potential energy. In addition, we use geometric genetic operators which include the element of mutation. The resulting algorithm performs well for the NaCl clusters studied here. Comparison with ab initio results shows a convincing consistency with the cluster structures obtained in this work. In addition, we report apparent global minima for $(NaCl)_N$ for $N > 18$ that have not been studied previously. The algorithm is still sufficiently simple to be very easily generalized to molecular systems. Currently, we use a slightly modified algorithm to investigate the global minimum structures of water molecules up to $N = 23$. Simultaneously, we compare three simple but widely used empirical point-charge water models.

References and Notes

(1) Stillinger, F. H. *Phys. Rev. E* **1999**, *59*, 48.
(2) Martin, T. P. *J. Chem. Phys.* **1977**, *67*, 5207.
(3) Diefenbach, J.; Martin, T. P. *J. Chem. Phys.* **1985**, *83*, 4585.
(4) Phillips, N. G.; Conover, C. W. S.; Bloomfield, L. A. *J. Chem. Phys.* **1991**, *94*, 4980.
(5) Tang, E. C. M.; Tang, T. B. *Z. Phys. D* **1993**, *28*, 61.
(6) Amara, P.; Straub, J. E. *Phys. Rev. B* **1996**, *53*, 13857.
(7) Doye, J. P. K.; Wales, D. J. *Phys. Rev. B* **1999**, *59*, 2292.
(8) Michaelian, K. *Am. J. Phys.* **1998**, *66*, 231.
(9) Niesse, J. A.; Mayne, H. R. *J. Comput. Chem.* **1997**, *18*, 1233.
(10) Deaven, D. M.; Tit, N.; Morris, J. R.; Ho, K. M. *Chem. Phys. Lett.* **1996**, *256*, 195.
(11) (a) Zeiri, Y. *Comput. Phys. Comm.* **1997**, *103*, 28. (b) Zeiri, Y. *J. Chem. Phys.* **1997**, *102*, 2785.
(12) Goldberg, D. E. *Genetic algorithms in search, optimization, and machine learning*; Addison-Wesley: Reading, PA, 1989.
(13) Tosi, M. P.; Fumi, F. G. *J. Phys. Chem. Solids* **1964**, *25*, 45.
(14) Rittner, E. S. *J. Chem. Phys.* **1951**, *19*, 1030.
(15) Press, W.; Flannery, B. P.; Teukolowsky, S.; Vetterling, W. *Numerical Recipes in C*: *The Art of Scientific Computing*; Cambridge University Press: Cambridge, U.K., 1988.
(16) Wolf, M. D.; Landman, U. *J. Phys. Chem. A* **1998**, *102*, 6129.
(17) Hartke, B. *J. Comput. Chem.* **1999**, *20*, 1752.
(18) Ashcroft, N. W.; Mermin, N. D. *Solid State Physics*; Saunders College Publishing: Fort Worth, TX, 1976.
(19) Martin, T. P. *Phys. Rep.* **1983**, *95*, 167.
(20) Ochsenfeld, C.; Ahlrichs, R. *J. Chem. Phys.* **1992**, *97*, 3487.
(21) Ayuela, A.; López, J. M.; Alonso, J. A.; Luaña, V. *Physica B* **1995**, *212*, 329.