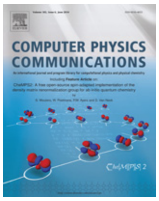
Computer Physics Communications 185 (2014) 1639–1646

Contents lists available at ScienceDirect

Computer Physics Communications

journal homepage: www.elsevier.com/locate/cpc



Efficient implementation and application of the artificial bee colony algorithm to low-dimensional optimization problems

Guido Falk von Rudorff[a], Christoph Wehmeyer[a,b], Daniel Sebastiani[a,b,*]

[a] *Dahlem Center for Complex Quantum Systems, Freie Universität Berlin, Arnimallee 14, 14195 Berlin, Germany*

[b] *Institute of Chemistry, Martin-Luther-Universität Halle-Wittenberg, von-Danckelmann-Platz 4, 06120 Halle, Germany*

ARTICLE INFO

Article history:
Received 23 October 2013
Received in revised form
6 January 2014
Accepted 7 March 2014
Available online 18 March 2014

Keywords:
Swarm-intelligence
Molecular clusters
Parallelization

ABSTRACT

We adapt a swarm-intelligence-based optimization method (the artificial bee colony algorithm, ABC) to enhance its parallel scaling properties and to improve the escaping behavior from deep local minima. Specifically, we apply the approach to the geometry optimization of Lennard-Jones clusters. We illustrate the performance and the scaling properties of the parallelization scheme for several system sizes (5–20 particles). Our main findings are specific recommendations for ranges of the parameters of the ABC algorithm which yield maximal performance for Lennard-Jones clusters and Morse clusters. The suggested parameter ranges for these different interaction potentials turn out to be very similar; thus, we believe that our reported values are fairly general for the ABC algorithm applied to chemical optimization problems.

1. Introduction

Structure optimization has been a matter of interest for chemical physics and physical chemistry [1–7], as the equilibrium structure of a system is most important for describing the physical and chemical properties [2]. In particular, molecular clusters have been studied in detail [3,8–18]. Even for the most simple interaction potentials, the potential energy surface (PES) for clusters with more than four atoms is highly complicated and exhibits multiple non-connected minima. As the global minima of these cluster types cannot be found analytically, various methods have been developed and applied: artificial bee colony [19–22], basin-hopping [23,12,24], dynamic lattice search [25], genetic algorithm [26,27], Monte Carlo search [28], parallel tempering [29], particle swarm optimization [21,30], simulated annealing [21,31–33], swarm intelligence [34], tabu search [35–37] and minima hopping [38], which can be considered to be the most common approach for cluster optimization.

The artificial bee colony (ABC) algorithm has been used for various optimization problems from chemical physics [39,40], engineering [41,42], and computer science [43–46]. Recently, we presented a modification of the artificial bee colony (ABC) algorithm for the optimization of molecular geometries [47].

The main idea of the ABC algorithm is to search the PES using a swarm intelligence approach. Several replications of a random conformation, called foragers, are placed on the PES and search for local minima. The computational resources are distributed dynamically among these foragers based on their relative energies, such that foragers in energetically favorable regions get more computational resources. This stochastic element is crucial for the performance and efficiency of the ABC approach.

When applied to clusters, the scaling of the ABC algorithm is exponential w.r.t. to the cluster size. As large clusters are more interesting, we want to push the limit of the computationally accessible search space. With increasing cluster size, the optimization is only feasible with nearly optimal values for the internal parameters of the algorithm, so an understanding of the parameters of the ABC algorithm is vitally important. Therefore, we analyze the reaction of the algorithm on parameter changes and derive a set of valid parameter ranges that both helps ensuring convergence in the first place and yields good performance. For the parameter determination, we used the Lennard-Jones potential

$$E_{\mathrm{LJ}} = 4\varepsilon \sum_{i<j} \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^{6} \right] \tag{1}$$

in reduced units, that is distances in σ and energies in ε. For evaluation, we optimized cluster structures for both the Lennard-Jones potential and the Morse potential with its range-determining parameter a

$$E_{\mathrm{M}} = \varepsilon \sum_{i<j} [\exp(2a(1 - r_{ij}/r_0)) - 2\exp(a(1 - r_{ij}/r_0))]. \tag{2}$$

[*] Corresponding author at: Dahlem Center for Complex Quantum Systems, Freie Universität Berlin, Arnimallee 14, 14195 Berlin, Germany. Tel.: +49 3037447445.
E-mail address: daniel.sebastiani@chemie.uni-halle.de (D. Sebastiani).

http://dx.doi.org/10.1016/j.cpc.2014.03.006
0010-4655/© 2014 Elsevier B.V. All rights reserved.

As the global minima are well-known for the Lennard-Jones and Morse potentials, it is easily possible to define a convergence criterion based on known reference solutions. For applying the ABC algorithm to problems without any known solution, we derive two intrinsic convergence criteria from extensive statistics.

For larger clusters or more complex problems, the computational costs increase quickly, so an efficient parallelization strategy is mandatory. As neither of the previous versions [47,19] of the ABC algorithm is parallelized, this manuscript reports two parallelization strategies and evaluate their impact on the convergence behavior.

The potential energy surface often exhibits narrow funnels that are difficult to find by MD at finite temperatures. These funnels may be of particular interest like, for example, misfolded non-equilibrium structures in protein folding. For finite temperatures, the Maxwell–Boltzmann-distribution in principle requires the knowledge of all geometries, especially those with a minimum energy such that the according Maxwell–Boltzmann factors can be calculated. Therefore, having different methods at hand for finding minimal energy conformations is desirable. In this paper, we use molecular clusters with phenomenological potentials in order to evaluate and tune the artificial bee colony algorithm.

The paper is divided into three main sections: Methods, Results and Discussion. Each of which itself is divided into three subsections on the targets of this paper: the suggested parameter ranges, convergence criteria, and parallelization strategies. Finally, the information on these separate parts is evaluated together with respect to both limitations and capabilities.

2. Methods

2.1. Working principles

The ABC algorithm basically tries to find a point with lower energy on the PES of a cluster by deriving candidate conformations from the point with the lowest energy found so far, with an additional stochastic component. Up to now, the ABC algorithm has been tested on both purely mathematical test functions [48] and cluster structures [47]. In both cases, the costs for the evaluation of each candidate conformation are very small. In this paper, we measure the computational costs by counting the number of energy calculations for a given conformation, that is, the number of single point calculations (SPC).

The total optimization process is divided into cycles [20]. Each cycle begins with an employee bee phase and ends with an onlooker bee phase, optionally followed by a scout bee phase. The employee bees hold a candidate conformation and know about the best solution of all candidate conformation they tested so far, whereas the O onlooker bees have access to the information of all employed bees. In the employee phase, each employee bee derives a new candidate conformation either by generating a totally random one or by choosing a point in the search space that is inside a hypersphere centered around the current conformation of the employed bee. In general, only if the new conformation has a lower energy than the previous one, the employed bee adapts the new position. If the employed bee discards a new position multiple times in a row, the radius of the hypersphere, the current size level, is halved. In the case an employed bee found no better position for a certain period, it becomes a scout bee and is forcibly set to a random point on the PES, where it becomes an employed bee again. This scout step is important for the sampling efficiency of the algorithm. After all employed bees have finished, the onlooker phase starts: each onlooker bee selects one employed bee based on a probability distribution defined by the energies of the current worker conformations. In this way, employed bees with a lower energy are preferred over those with higher energies. Each selected employed bee searches for a new position again. One employed bee can be selected by multiple onlooker bees. After all onlookers have completed, the cycle is finished.

In our implementation, we use implicit onlookers. This means that the onlooker bees are modeled by allowing the employed bees to perform another random local search step. In order to stress this difference, we use the expression *worker* for the foragers, disregarding whether they are currently used as employed bee, onlooker bee or scout bee during the optimization process. In other words, the number of workers W denotes the number of different positions on the PES that is kept in memory at any time.

During the onlooker phase, the algorithm allows each worker i of the W workers to perform another $S(W_i)$ steps. The distribution among the threads in the parallelized versions is given by $S(W_i, T_j)$ where T_j denotes the jth thread from t threads in total, so that

$$S(W_i) = \sum_{j=1}^{t} S(W_i, T_j) \tag{3}$$

$$S(T_i) = \sum_{j=1}^{O} S(W_j, T_i). \tag{4}$$

We assume t to be less than or equal to the number of available parallel compute nodes.

2.2. Suggested parameter ranges

In principle, an optimal set of values for the free parameters of the algorithm can be obtained for any cluster type (for example Lennard-Jones [49], Morse [50], Tersoff [51,52], and TIP5P [53]) and cluster size [54,55]. However, these settings would only be of interest for this particular application and would not be usable in order to cope with other optimization problems. Therefore, we focus on describing ranges for the algorithmic parameters. These ranges are further referred to as suggested parameter ranges (SPR) and are defined by two criteria: at least 90% of all N runs converge and the average costs are not higher than the threshold of 110% of the minimal average costs for this cluster. Although this tight criterion introduces some special values like the data for $e = 17$ in Fig. 3, it is helpful to determine the actual optimal parameter values and supports the transferability of the suggested parameter ranges.

Determining these SPR is an optimization problem itself. Therefore, we have chosen to apply the ABC algorithm to find these parameter ranges. For the target quantity that is to be optimized, we average the computational cost for reproducing the known optimal conformation over $N \simeq 500$ runs with a specific parameter set. Regarding the computational costs, Lennard-Jones clusters with five to twenty atoms are used as reference solutions [56]. Using the reduced unit representation of the Lennard-Jones potential guarantees validity of the results for all parametrizations of the Lennard-Jones potential.

Starting from the best parameter set for each cluster size e, five out of the six parameters are kept constant and the sixth one is used for the rasterization of the local minimum in parameter space. Again, each data point is calculated by averaging over N optimization runs. The resulting data is smoothed in order to be able to define a minimum from the noised information.

Fig. 1 illustrates the procedure of determining the SPR. First of all, a parameter set has to be found by the master optimization runs. Varying the number of onlookers leads to the raw data. As each of the data points may vary in spite of the averaging, the raw data is smoothed by calculating a Bézier curve, which in turn is used for SPR determination.

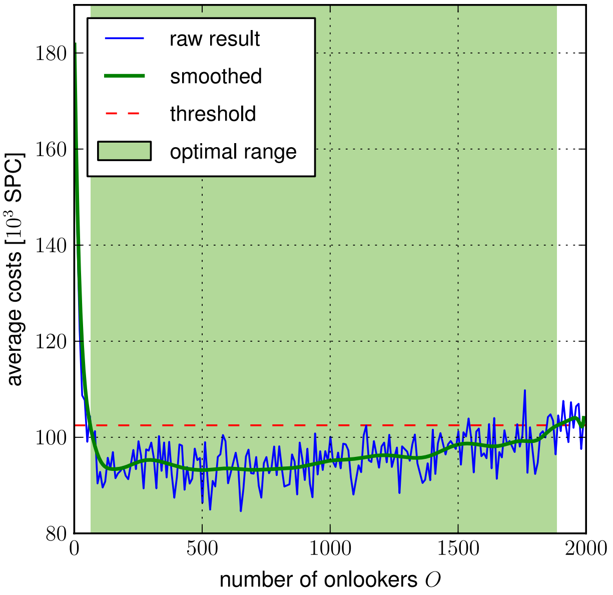


Fig. 1. Suggested parameter range for the number of onlookers for the Lennard-Jones cluster with 15 atoms: raw rasterization results and Bézier-smoothed curve.

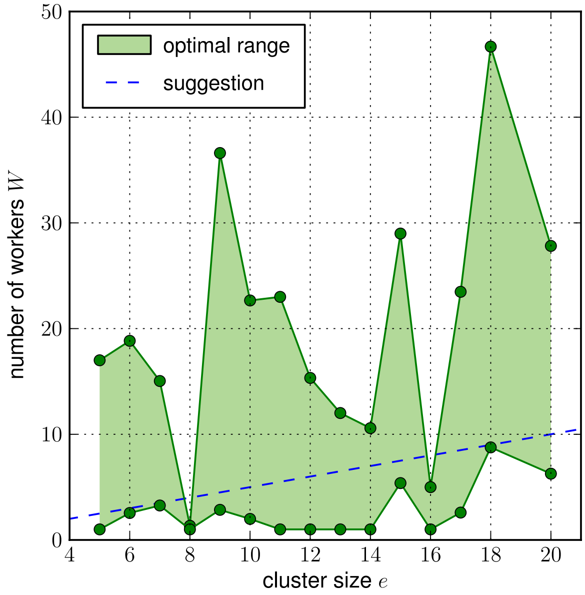


Fig. 2. Suggested parameter range for the number of workers depending on the cluster size for the Lennard-Jones potential; a suggested dependency (dashed).

2.3. Convergence criterion

For unsolved problems, there are no data for comparing to. Therefore, defining a stop criterion immanent to the algorithm is necessary. It is common to use the convergence history for the definition, that is, assuming an optimization run to be finished in the case there were no improvements for a certain number of cycles or a gradient is below a given threshold. As the ABC algorithm itself does not require the computation of gradients, it seems favorable to evaluate whether the cycle based threshold is reasonable. We analyze the convergence behavior of 500 optimization runs for three different cluster sizes and two different cluster types each.

2.4. Parallelization

We analyze two parallelization strategies, whose difference consists of the onlooker phase handling. The worker phase is what is called the embarrassingly parallel. Hence, the onlooker phase is the crucial part, as the selection of the next onlooker depends on the result of the previous selected onlooker. During this serial computation, most of the computational resources remained unused, so a parallelized version is desired. For tasks of unpredictable duration, one could use a pool structure. [57] In our approach, we exploit our knowledge about the relative duration of the tasks during the onlooker phase.

The parallelization strategy A determines the schedule for the whole onlooker phase before beginning with the onlooker phase, whereas the former implementation [47] selected the next onlooker on the fly during the onlooker phase, thus introducing a slight bias towards workers that have been selected during the current onlooker phase already. Therefore, our current implementation assumes that each selection of a worker improves its current conformation by a factor, which is calculated by the average improvement of all workers during the onlooker phase of the last cycle. See Supplemental material for a detailed flowchart (see Appendix A).[1]

Our implementation uses implicit onlookers which means that one onlooker step is performed by an additional optimization step of the selected worker. This has the advantage of reduced management overhead for single-threaded runs at the expense of the inability to perform two onlooker steps from different threads T_i in parallel. Therefore, the workers selected for onlooker steps have to be distributed among the available threads. First, we sort the worker by $S(W_i)$. Afterwards, each worker W_m is assigned to the thread T_j with minimal $S(T_j)$. The dispatching begins with the highest $S(W_i)$. This results in an optimal distribution without splitting one worker across multiple threads.

Strategy A relies on the existence of a certain number of workers with comparable energies. Otherwise, it may happen that the selection probability of a certain worker is close to 100%. In this case, strategy A becomes computationally inefficient due to forced synchronizations [57]. This situation is common for TIP5P and Morse clusters and less likely for Lennard-Jones clusters. This problem can be addressed by a certain modification, strategy B, by distributing the workload for one worker across multiple threads T_i. In our implementation, this is only performed if the costs of copying one worker are neglectable compared to the parallelization speedup.

3. Results

3.1. Suggested parameter ranges

We have determined SPR for the control parameters. The first parameter is the number of workers W, which contributes to the explorative character of the ABC. We have shown in Fig. 2 the SPR for LJ clusters of different sizes. We observe that, in general, $W = 3 \ldots 20$ is a good choice for smaller clusters up to 20 particles.

The dependency of the second parameter, the number of onlookers O, on the cluster size is visualized in Fig. 3 using a logarithmic scale. With the exception of one cluster, the range from 80 to 200 is optimal. The upper limit of the SPR increases clearly exponentially with a raising cluster size, just like the computational costs do. The SPR for this parameter is even larger than the SPR for the number of workers. Therefore, the number of onlookers is even less important for performance tuning as long as the selected value is contained in the wide SPR.

Fig. 4 shows the SPR for the number of successive tries with one step size. It turns out to be mostly constant for the analyzed cluster

[1] See Supplemental material in Figures 5 and 6 at http://dx.doi.org/10.1016/j.cpc.2014.03.006 for more details on the structure of the onlooker phase.

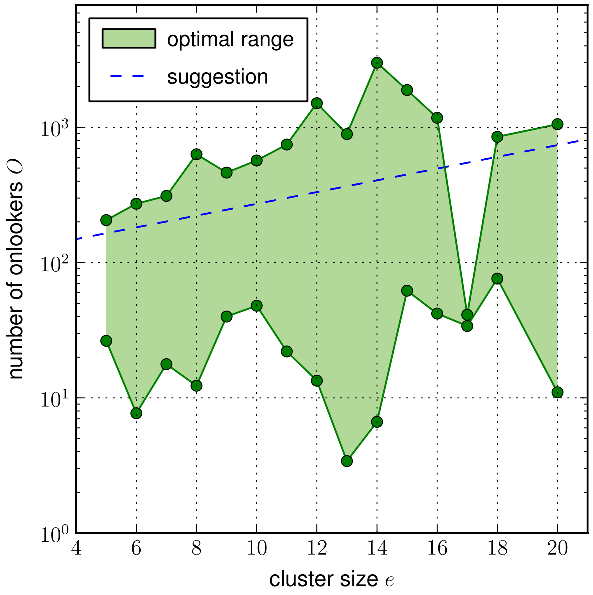


Fig. 3. Suggested parameter range for the number of onlookers w.r.t. the cluster size for the Lennard-Jones potential; a suggested dependency (dashed).

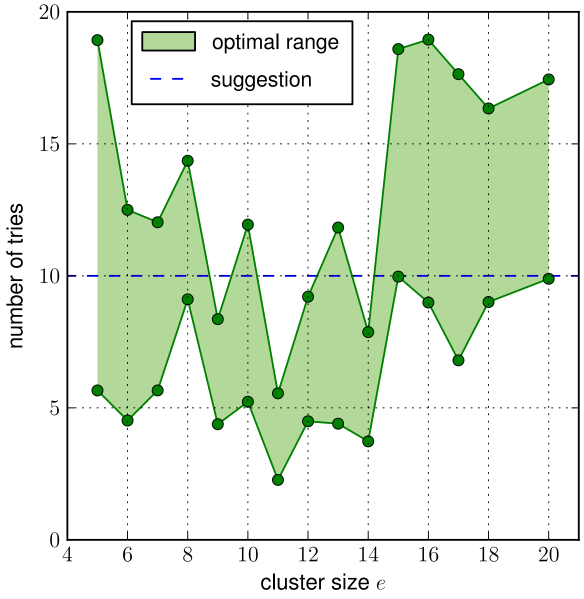


Fig. 4. Suggested parameter range for the number of tries with one step size depending on the cluster size for the Lennard-Jones potential; a suggested dependency (dashed).

sizes. Compared to the wide SPR for the two parameters discussed above, selecting an applicable value at sufficient computational costs is more difficult in this case.

Only two of the parameters of the ABC algorithm represent a physical property: the first one of which is the size of the box used for the initialization of the cluster elements. Fig. 5 shows dependency of the SPR on the cluster size together with the radius of the resulting conformation. The conformation size is approximated by the diameter of the smallest sphere that contains the whole cluster. The obtained data confirms the previously reported observation [47] that the algorithm is faster in expanding a compact cluster to the optimal conformation than it is in compressing the initial cluster.

During the optimization, the difference between the old position and the new candidate position within the search space is limited to the second parameter representing a physical value, the step length within the search space. In Fig. 6, we show the influence of the step length of the workers on the optimization performance. We observe that the SPR gets more narrow with the increasing cluster size. In general, larger clusters require the algorithm to try smaller steps.

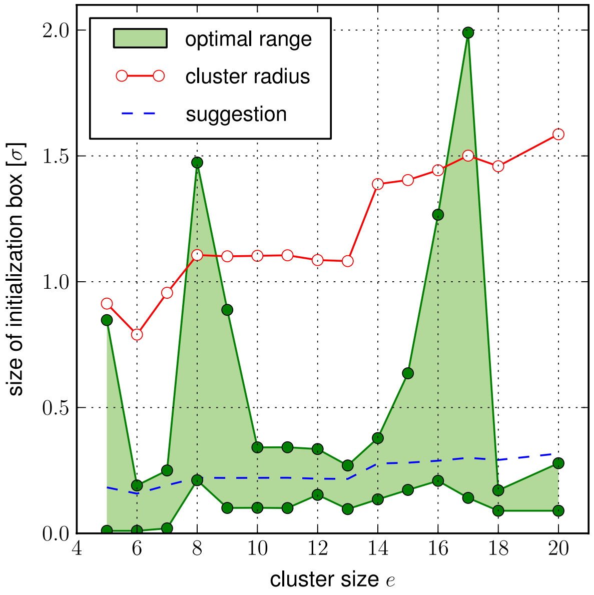


Fig. 5. Suggested parameter range for the initialization box size w.r.t. the cluster size for the Lennard-Jones potential together with the resulting clusters' size and a suggested dependency (dashed).

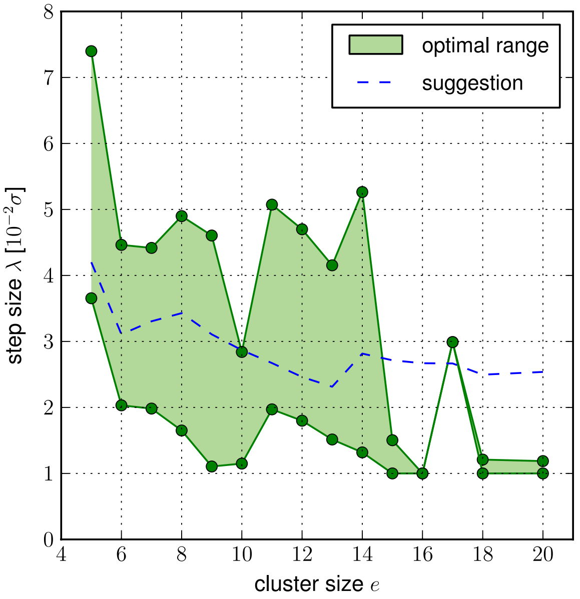


Fig. 6. Suggested parameter range for the step length within the search space for the Lennard-Jones potential depending on the cluster size; a suggested dependency (dashed).

The last parameter, the number of different size levels for the step length used throughout the optimization runs, shows the same behavior for most of the analyzed cluster sizes, as presented in Fig. 7. For values above the SPR, the percentage of runs converging during the allotted time drops significantly, yielding higher average costs. However, the slope is very small, thus allowing the selection of values slightly above the SPR. In general, the SPR is wide for small clusters.

Interestingly, it is observed for each parameter that a value below the suggested SPR leads to a drastic increase (factor 300

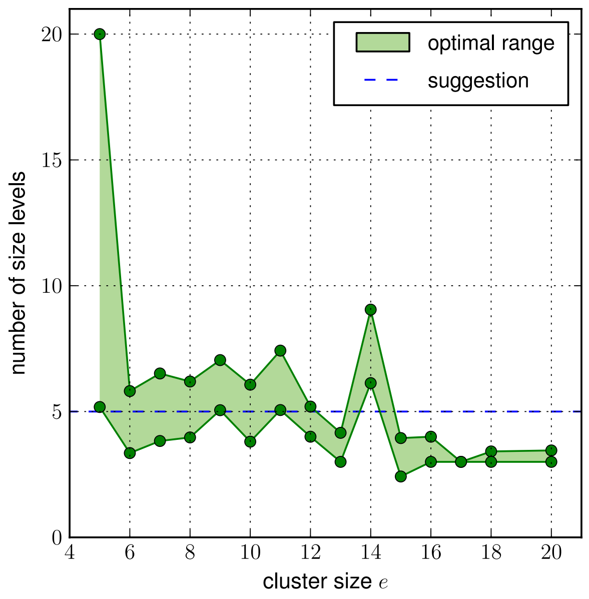


Fig. 7. Suggested parameter range for the number of size levels for the step length applied to the Lennard-Jones potential along with a suggested dependency.

in some cases) of the computational cost; a value above the SPR leads only to a mild increase. This illustrates once more the great importance of the incorporation of swarm-intelligence aspects, such as independent communicating agents, as opposed to a straightforward serial implementation of the search. Additionally, we find that increasing the cluster size both narrows the SPR and increases the share of unconverged runs.

This is a case in point for the importance of proper parameter tuning for large cluster structures. See the Supplemental material (Appendix A) for performance data over the range of clusters used for parameter optimization.[2]

The suggested parameters remain optimal for the long-ranged Morse potential with $a = 3$. See the Supplemental material (Appendix A) for detailed statistics.[3]

3.2. Convergence criterion

As a second part of this study, we examined if the popular criterion of I optimization cycles without finding a better geometry is applicable to determine convergence for the ABC algorithm. For this aim, we have performed a series of independent optimizations using this criterion and determined the share of optimization runs that indeed did converge to the a priori known global minima; this is shown in Fig. 8 for different sizes of Morse and LJ clusters and different choices of I.

We see that Lennard-Jones clusters need a higher value for I than Morse clusters ($a = 3$) in order to reach the same probability for reproducing the reference conformation. A significant share from about 7% to 77% of the runs found the reference cluster directly, that is, in each cycle, an even better conformation was found and the last one was the reference conformation. In general, optimizing larger clusters requires a higher value for I. For $I > 10$, the probability of having found the minimum is roughly proportional to $\log I$. For each cluster size in Fig. 8, there is a value of I that ensures convergence such that increasing I even further does not give better results but increases the computational costs. This maximal value for I ranges from about 170 for M_{10} to about 400,000 for M_{40}.

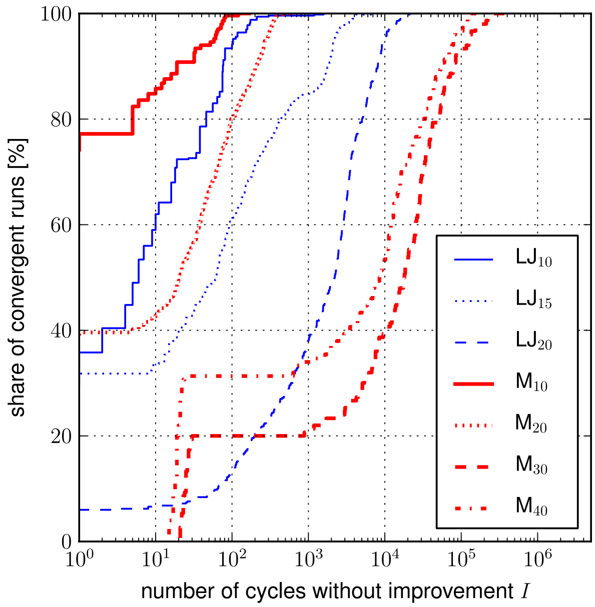


Fig. 8. Cumulative histogram for the share of convergent runs (below expected energy threshold) in the case the runs are aborted after I successive cycles without finding a lower energy. The data are presented for both Lennard-Jones and Morse clusters ($a = 3$). Subscripts denote the cluster size.

Although the suggested parameter ranges were not calibrated by using the performance data for Morse clusters at all, Fig. 8 clearly shows that using the suggested parameters yields a quick convergence for this potential, as well. This is even the case for cluster sizes well beyond the cluster sizes that have been used for SPR determination.

3.3. Parallelization

In order to compare the parallelization strategies, we applied the algorithm to two problems (LJ_{38} and LJ_{55}) with both strategies and measured the throughput in terms of SPC per second. The throughput of the parallelized versions is normalized by dividing it by the single thread performance. The results are presented in Fig. 9. For both strategies, working on a larger cluster is slightly more efficient. In direct comparison, strategy B scales much better than strategy A in spite of the number of additional memory copy operations. For the LJ_{55} cluster, strategy B yields a computational efficiency of 87% with $t = 12$, thus being more efficient than strategy A with only 38%. The absolute throughput of A is higher than the one from B for up to two threads. The influence of the algorithmic changes to allow for parallelization on the convergence behavior is within the standard deviation. See the Supplemental material (see Appendix A) for example data for the LJ_{38} cluster.[4]

4. Discussion

4.1. Suggested parameter ranges

In general, the suggestions are based on an estimated cluster diameter for those parameters that have units of length. For the

[2] See Supplemental material in Table I at http://dx.doi.org/10.1016/j.cpc.2014.03.006 for performance data.

[3] See Supplemental material in Figure 4 at http://dx.doi.org/10.1016/j.cpc.2014.03.006 for performance data.

[4] See Supplemental material in Figure 3 at http://dx.doi.org/10.1016/j.cpc.2014.03.006 for more details on the influence of the parallelization on the convergence behavior.

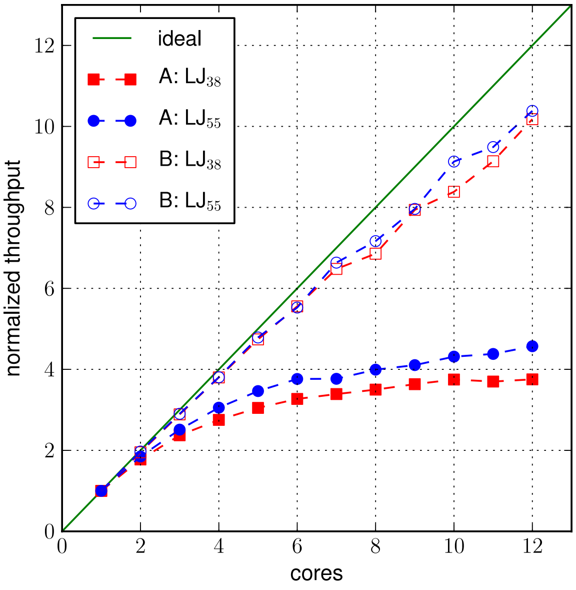


Fig. 9. Scaling behavior for the two strategies A and B and for different sized Lennard-Jones clusters. The throughput for single threaded computation has been set to 1. Each data point is averaged over at least 10^7 energy function evaluations.

other parameters, we selected the most basic model that fits the data obtained by the parameter optimization. Otherwise, the use of the suggested parameter ranges for the optimization of clusters well beyond 20 atoms or potentials other than Lennard-Jones would be overly limited due to overfitting.

For most cluster sizes, the SPR for the first parameter W is rather wide, so the computational costs are only marginally depending on W. When using few workers, the data shows less exploration of the PES. We suggest choosing half the number of particles within the cluster as value for the number of workers.

There is a clear tendency towards lower computational costs for a high number of onlookers compared to the number of workers. Thus, the central idea of the algorithm, namely the weighting of the workers depending on their cluster conformation, turns out to significantly enhance the efficiency. For parallelization, a high number of onlookers is desirable, so we suggest using

$$O(e) = 100\exp(0.1e) \tag{5}$$

for clusters with $e < 30$. As the scaling behavior of the parallelization does not benefit from an even higher number of onlookers, capping the exponential growth after 30 elements is necessary.

As far as the third parameter, the number of successive tries with a certain step size, is concerned, the cluster size seems to be irrelevant for the SPR. Due to a lack of evidence for a clear dependency, we suggest using a value between 10 and 12 for cluster optimization.

The exact value for the initialization box size depends strongly on the implementation. We suggest a value of ten percent of the size D of the resulting conformation. It would be useful to define a default value without needing at least a rough prediction of the dimension of the conformation. However, in most cases the order of magnitude can be deduced from the problem description. When in doubt, using a box slightly bigger than the optimal value is less costly than overcompressing the initial cluster.

For the step length, the exact value depends on the implementation of the energy function in the same way as the initialization box size. Especially when applying the algorithm to larger clusters, tuning this parameter for performance is crucial, as the SPR is very narrow for this kind of problems. As the algorithm itself requires a step size λ bigger than zero, a fit leads to the suggestion

$$\lambda = \left(\frac{0.1}{e} + 0.003\right) D. \tag{6}$$

Using values above the SPR increases the computational costs in two ways: the converging optimization runs get more expensive and the number of converging runs decreases rapidly.

As the SPR for the last parameter, the number of size levels for the step length, is rather small, tuning this parameter is important. Fortunately, only integer values are valid input for this parameter, so the number of candidate values is quite small. Having this in mind, we suggest setting this parameter to five, based on the data from Fig. 7. As this parameter limits the total step length, the allowed difference between two positions on the PES decreases for increasing cluster sizes. The higher the dimension of the search space, the harder it is to explore the hypersphere around any given point.

In summary, for all parameters with the exception of the last one, it is clearly better to use a value slightly above the SPR instead of going below the SPR.

4.2. Convergence criterion

The gathered data justifies defining a certain number I of successive cycles without any improvement as stop criterion for the algorithm. However, the data allows for at least two different approaches. The first one consists of defining a rather high value I_1 for I and performing one optimization run. The other possibility is given by defining a rather low value I_2 and performing several independent optimization runs. Of course, for an exact comparison of these strategies, both I_1 and I_2 as well as the average convergence behavior of the specific optimization problem have to be known. As this non transferable information is of rather limited interest, the data from Fig. 8 merely justifies using the second approach for small clusters and sticking to the first one for big clusters.

4.3. Parallelization

In general, strategy B leads to a better scaling behavior compared to strategy A as shown in Fig. 9. More expensive SPC result in a slightly better scaling behavior, as the synchronization overhead is getting smaller compared to the tasks performed. Using strategy B, the algorithm shows a good scaling behavior, mostly limited by the small part of tasks performed between the worker and onlooker phase and the necessary synchronization.

For small machines, strategy A is better, as the raw throughput is slightly higher. At the expense of a higher implementation complexity and higher memory usage, strategy B leads to far better resource usage for bigger machines.

As all the calculations for the W workers are fully independent and need no preparation or special distribution among the threads, increasing the number of workers also increases the throughput. In particular, using $W < t$ during the computation leads to significant overhead. The dependency of the costs of the onlooker phase on the number of workers is neglectable, hence the optimization runs will not take longer – in terms of wall time, not in terms of SPC – with W as an integer multiple of t.

For O, this rule does not apply. As long as O is so small that the synchronization overhead is big compared to the duration of one onlooker phase, padding O to be an integer multiple of t will not lengthen the necessary wall time but might speed up finding a new solution in the case the algorithm gets stuck within a local minimum. For large O, timing and scheduling issues gain importance, so two threads performing the same number of SPC will not necessarily finish at the same time. Therefore, padding O is not helpful any more.

In theory, the number of size levels for the step length influences the number of worker resets being performed. But in comparison with deriving a new candidate position followed by an energy function evaluation for a Lennard-Jones cluster, resetting a worker is of neglectable costs. Hence, none the other parameters affect the scaling behavior of the algorithm.

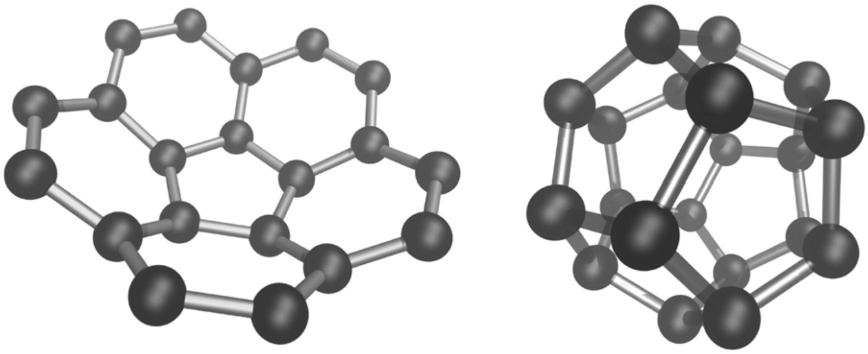


Fig. 10. Local minima for C_{20} Tersoff cluster with an energy of -121.8 eV (left) and -120.2 eV (right).

5. Limitations and perspectives

When applied to C_{20} Tersoff [51] cluster, the ABC algorithm is reliable in finding the fullerene structure but fails to come up with the bowl structure (in Fig. 10), although the latter one is about 1.6 eV lower than the former. Unless we started from a preoptimized conformation similar to the bowl structure, all conformations used for random sampling of the possible cluster structures for the initialization lead to the fullerene structure. This behavior does not change under large variations of the parameters of the ABC algorithm. Hence, we suspect the bowl structure to have a rather small attractive region in conformation space. The difficulties to find non-spherical structures seem to be typical for our approach as long as we do not use coordinate transformations. For Lennard-Jones clusters, using a cuboid as initialization box does have an influence on the convergence behavior. See the Supplemental material for a diagram (see Appendix A).[5] However, this method does not lead to any bowl structures as a result of the optimization algorithm. Apparently, there is a high potential barrier separating the two conformations in Fig. 10. The reason for the apparent bias towards spherical conformations will be the subject of further analysis.

6. Conclusion

In this study, we have quantified the influence of the control parameters on the efficiency of the artificial bee colony (ABC) algorithm. Further, we have examined convergence criteria and two parallelization strategies.

Our findings show that the algorithm that does not make use of special information like cluster symmetries performs well for LJ and Morse clusters with up to 40 atoms and is applicable to long-ranged Morse clusters with up to 66 atoms. The algorithm shows an increased sensitivity to the parameters with increasing size of the clusters; this is a case in point for the importance of parameter tuning for the efficient structure optimization. For intermediate cluster sizes, we use the convergence behavior for the definition of a heuristic stop criterion. The algorithm itself was parallelized with an efficiency that gets close to ideal scaling, thus simplifying the application to more complex problems.

Our central results are the recommendations for the parameter values of the ABC algorithm in the context of molecular geometry optimizations, as shown Section 4.1. On a more general level, our findings show that swarm-intelligence based aspects in search strategies can significantly enhance the performance of optimization schemes that do not incorporate structural bias or gradient information.

Acknowledgments

This work has been supported by the German Research Foundation (DFG) under Grant SE 1008/6. Computing infrastructure was provided by the Northern German Supercomputing Alliance (HLRN) under Grant HLRN/bec00082.

Appendix A. Supplementary material

Supplementary material related to this article can be found online at http://dx.doi.org/10.1016/j.cpc.2014.03.006.

References

[1] Gerald Böhm, New approaches in molecular structure prediction, Biophys. Chem. 59 (1–2) (1996) 1–32.

[2] Markus Christen, Wilfred F. van Gunsteren, On searching in, sampling of, and dynamically moving through conformational space of biomolecular systems: a review, J. Comput. Chem. 29 (2) (2008) 157–166.

[3] Z. Li, Monte Carlo-minimization approach to the multiple-minima problem in protein folding, Proc. Natl. Acad. Sci. USA 84 (19) (1987) 6611–6615.

[4] Martin Saunders, Stochastic exploration of molecular mechanics energy surfaces. Hunting for the global minimum, J. Am. Chem. Soc. 109 (10) (1987) 3150–3152.

[5] Wilfred F. van Gunsteren, The role of computer simulation techniques in protein engineering, Protein Eng. Des. Sel. 2 (1) (1988) 5–13.

[6] Wilfred F. van Gunsteren, Dirk Bakowies, Riccardo Baron, Indira Chandrasekhar, Markus Christen, Xavier Daura, Peter Gee, Daan P. Geerke, Alice Glättli, Philippe H. Hünenberger, Mika A. Kastenholz, Chris Oostenbrink, Merijn Schenk, Daniel Trzesniak, Nico F.A. van der Vegt, Haibo B. Yu, Biomolecular modeling: goals, problems, perspectives, Angew. Chem. Int. Ed. Engl. 45 (25) (2006) 4064–4092.

[7] David J. Wales, H.A. Scheraga, Global optimization of clusters, crystals, and biomolecules, Science 285 (5432) (1999) 1368–1372.

[8] Yaojun A. Du, Thomas J. Lenosky, Richard G. Hennig, Stefan Goedecker, John W. Wilkins, Energy landscape of silicon tetra-interstitials using an optimized classical potential, Phys. Status Solidi B-Basic Solid State Phys. (2011) 2050–2055.

[9] Kuo Bao, Stefan Goedecker, Kenji Koga, Frédéric Lançon, Alexey Neelov, Structure of large gold clusters obtained by global optimization using the minima hopping method, Phys. Rev. B 79 (4) (2009) 041405.

[10] Jonathan P.K. Doye, Mark A. Miller, David J. Wales, The double-funnel energy landscape of the 38-atom Lennard-Jones cluster, J. Chem. Phys. 110 (14) (1999) 6896–6906.

[11] Jonathan P.K. Doye, David J. Wales, R. Stephen Berry, The effect of the range of the potential on the structures of clusters, J. Chem. Phys. 103 (10) (1995) 4234–4249.

[12] Jonathan P.K. Doye, David J. Wales, Mark A. Miller, Thermodynamics and the global optimization of Lennard-Jones clusters, J. Chem. Phys. 109 (19) (1998) 8143.

[13] Tim James, David J. Wales, Javier Hernández-Rojas, Global minima for water clusters $(H_2O)n$, $n \leq 21$, described by a five-site empirical potential, Chem. Phys. Lett. 415 (4–6) (2005) 302–307.

[14] Robert H. Leary, Jonathan P.K. Doye, Tetrahedral global minimum for the 98-atom Lennard-Jones cluster, Phys. Rev. E 60 (6) (1999) R6320–R6322.

[15] Lesley D. Lloyd, Roy L. Johnston, Christopher Roberts, Thomas V. Mortimer-Jones, Geometry optimisation of aluminium clusters using a genetic algorithm, Chem. Phys. Chem. 3 (5) (2002) 408–415.

[16] R. Martoňák, Artem R. Oganov, C.W. Glass, Crystal structure prediction and simulations of structural transformations: metadynamics and evolutionary algorithms, Phase Transitions 80 (4–5) (2007) 277–298.

[5] See Supplemental material in Figure 2 at http://dx.doi.org/10.1016/j.cpc.2014.03.006 for more details on the influence of the initialization box shape.

[17] Chris Roberts, Roy L. Johnston, Nicholas T. Wilson, A genetic algorithm for the structural optimization of Morse clusters, Theor. Chem. Acc. 104 (2) (2000) 123–130.
[18] Michael Springborg, Yi Dong, Valeri G. Grigoryan, Violina Tevekeliyska, Denitsa Alamanova, Elisaveta Kasabova, Sudip Roy, Jan-Ole Joswig, Abu Md. Asaduzzaman, Theoretical studies of structural, energetic, and electronic properties of clusters, Z. Phys. Chem. 222 (2–3) (2008) 387–405.
[19] Dervis Karaboga, Bahriye Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, J. Global Optim. 39 (3) (2007) 459–471.
[20] Dervis Karaboga, Bahriye Akay, A comparative study of Artificial Bee Colony algorithm, Appl. Math. Comput. 214 (1) (2009) 108–132.
[21] R. Venkata Rao, P.J. Pawar, Parameter optimization of a multi-pass milling process using non-traditional optimization algorithms, Appl. Soft Comput. 10 (2) (2010) 445–456.
[22] Guopu Zhu, Sam Kwong, Gbest-guided artificial bee colony algorithm for numerical function optimization, Appl. Math. Comput. 217 (7) (2010) 3166–3173.
[23] David J. Wales, Jonathan P.K. Doye, Global optimization by Basin-Hopping and the lowest energy structures of Lennard-Jones clusters containing up to 110 atoms, J. Phys. Chem. A 101 (28) (1997) 5111–5116.
[24] Mark T. Oakley, Roy L. Johnston, David J. Wales, Symmetrisation schemes for global optimisation of atomic clusters, Phys. Chem. Chem. Phys. 15 (2013) 3965–3976.
[25] Xiang Jing Lai, Ru Chu Xu, Wen Qi Huang, Prediction of the lowest energy configuration for Lennard-Jones clusters, Sci. China-Chem. 54 (6) (2011) 985–991.
[26] R.C. Glen, A.W.R. Payne, A genetic algorithm for the automated generation of molecules within constraints, J. Comput.-Aided Mol. Des. 9 (2) (1995) 181–202.
[27] Vladimir A. Froltsov, Karsten Reuter, Robustness of 'cut and splice' genetic algorithms in the structural optimization of atomic clusters, Chem. Phys. Lett. 473 (4–6) (2009) 363–366.
[28] George Chang, Wayne C. Guida, W. Clark Still, An internal-coordinate Monte Carlo method for searching conformational space, J. Am. Chem. Soc. 111 (12) (1989) 4379–4386.
[29] Ulrich H.E. Hansmann, Parallel tempering algorithm for conformational studies of biological molecules, Chem. Phys. Lett. 281 (1–3) (1997) 140–150.
[30] Jian Lv, Yanchao Wang, Li Zhu, Yanming Ma, Particle-swarm structure prediction on clusters, J. Chem. Phys. 137 (8) (2012) 084104.
[31] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing., Science 220 (4598) (1983) 671–680.
[32] R.C. van Schaik, Wilfred F. van Gunsteren, H.J. Berendsen, Conformational search by potential energy annealing: algorithm and application to cyclosporin A, J. Comput.-Aided Mol. Des. 6 (2) (1992) 97–112.
[33] Jooyoung Lee, Harold A. Scheraga, S. Rackovsky, New optimization method for conformational energy calculations on polypeptides: conformational space annealing, J. Comput. Chem. 18 (9) (1997) 1222–1232.
[34] René Fournier, Satya Bulusu, Stephen Chen, Jamie Tung, Using swarm intelligence for finding transition states and reaction paths, J. Chem. Phys. 135 (10) (2011) 104117.
[35] Fred Glover, Tabu search for nonlinear and parametric optimization (with links to genetic algorithms), Discreet Appl. Math. 49 (1–3) (1994) 231–255.
[36] Christoph Grebner, Johannes Becker, Svetlana Stepanenko, Bernd Engels, Efficiency of tabu-search-based conformational search algorithms, J. Comput. Chem. 32 (10) (2011) 2245–2253.
[37] Svetlana Stepanenko, Bernd Engels, Gradient tabu search, J. Comput. Chem. 28 (2) (2007) 601–611.
[38] Sandro E. Schönborn, Stefan Goedecker, Shantanu Roy, Artem R. Oganov, The performance of minima hopping and evolutionary algorithms for cluster structure prediction, J. Chem. Phys. 130 (14) (2009) 144108.
[39] Christoph Schiffmann, Daniel Sebastiani, Artificial bee colony optimization of capping potentials for hybrid quantum mechanical/molecular mechanical calculations, J. Chem. Theory Comput. 7 (5) (2011) 1307–1315.
[40] Arvid Conrad Ihrig, Christoph Schiffmann, Daniel Sebastiani, Specific quantum mechanical/molecular mechanical capping-potentials for biomolecular functional groups, J. Chem. Phys. 135 (21) (2011) 214107.
[41] R. Venkata Rao, P.J. Pawar, Parameter optimization of a multi-pass milling process using non-traditional optimization algorithms, Appl. Soft Comput. 10 (2) (2010) 445–456.
[42] Mustafa Sonmez, Artificial Bee Colony algorithm for optimization of truss structures, Appl. Soft Comput. 11 (2) (2011) 2406–2418.
[43] Alok Singh, An artificial bee colony algorithm for the leaf-constrained minimum spanning tree problem, Appl. Soft Comput. 9 (2) (2009) 625–631.
[44] Shyam Sundar, Alok Singh, A swarm intelligence approach to the quadratic minimum spanning tree problem, Inf. Sci. 180 (17) (2010) 3182–3191.
[45] Tsung-Jung Hsieh, Hsiao-Fen Hsiao, Wei-Chang Yeh, Forecasting stock markets using wavelet transforms and recurrent neural networks: an integrated system based on artificial bee colony algorithm, Appl. Soft Comput. 11 (2) (2011) 2510–2525.
[46] Rui Zhang, Cheng Wu, An artificial bee colony algorithm for the job shop scheduling problem with random processing times, Entropy 13 (9) (2011) 1708–1729.
[47] Christoph Wehmeyer, Guido Falk von Rudorff, Sebastian Wolf, Gabriel Kabbe, Daniel Schärf, Thomas D. Kühne, Daniel Sebastiani, Foraging on the potential energy surface : a swarm intelligence-based optimizer for molecular geometry, J. Chem. Phys. 137 (19) (2012) 194110.
[48] Dervis Karaboga, Bahriye Basturk, On the performance of artificial bee colony (ABC) algorithm, Appl. Soft Comput. 8 (1) (2008) 687–697.
[49] J.E. Jones, On the determination of molecular fields. ii. From the equation of state of a gas, Proc. R. Soc. A-Math. Phys. Eng. Sci. 106 (738) (1924) 463–477.
[50] Philip Morse, Diatomic molecules according to the wave mechanics. II. vibrational levels, Phys. Rev. 34 (1) (1929) 57–64.
[51] J. Tersoff, Empirical interatomic potential for carbon, with applications to amorphous-carbon, Phys. Rev. Lett. 61 (25) (1988) 2879–2882.
[52] T. Halicioglu, Reproducibility of various properties of small carbon clusters using model potential functions, Chem. Phys. Lett. 179 (1–2) (1991) 159–164.
[53] Michael W. Mahoney, William L. Jorgensen, A five-site model for liquid water and the reproduction of the density anomaly by rigid, nonpolarizable potential functions, J. Chem. Phys. 112 (20) (2000) 8910–8922.
[54] Zoe E. Brain, Matthew A. Addicoat, Optimization of a genetic algorithm for searching molecular conformer space, J. Chem. Phys. 135 (17) (2011) 174106.
[55] Suman Samanta, Shankar Chakraborty, Parametric optimization of some non-traditional machining processes using artificial bee colony algorithm, Eng. Appl. Artif. Intell. 24 (6) (2011) 946–957.
[56] David J. Wales, Jonathan P.K. Doye, A. Dullweber, Matthew P. Hodges, F.Y. Naumkin, F. Calvo, Javier Hernández-Rojas, T.F. Middleton, The Cambridge Cluster Database, 2012, http://www-wales.ch.cam.ac.uk/CCD.html.
[57] Bernhard Bandow, Bernd Hartke, Larger water clusters with edges and corners on their way to ice: structural trends elucidated with an improved parallel evolutionary algorithm, J. Phys. Chem. A 110 (17) (2006) 5809–5822.