Larger Water Clusters with Edges and Corners on Their Way to Ice: Structural Trends Elucidated with an Improved Parallel Evolutionary Algorithm

Bernhard Bandow and Bernd Hartke*

Institut für Physikalische Chemie, Christian-Albrechts-Universität, Olshausenstrasse 40, 24098 Kiel, Germany

Received: January 24, 2006; In Final Form: March 6, 2006

For the difficult task of finding global minimum energy structures for molecular clusters of nontrivial size, we present a highly efficient parallel implementation of an evolutionary algorithm. By completely abandoning the traditional concept of generations and by replacing it with a less rigid pool concept, we have managed to eliminate serial bottlenecks completely and can operate the algorithm efficiently on an arbitrary number of parallel processes. Nevertheless, our new algorithm still realizes all of the main features of our old, successful implementation. First tests of the new algorithm are shown for the highly demanding problem of water clusters modeled by a potential with flexible, polarizable monomers (TTM2-F). For this problem, our new algorithm not only reproduces all of the global minima proposed previously in considerably less CPU time but also leads to improved proposals in several cases. These, in turn, qualitatively change our earlier predictions concerning the transitions from all-surface structures to cages with a single interior molecule, and from one to two interior molecules. Furthermore, we compare preliminary results up to $n = 105$ with locally optimized cuts from several ice modifications. This comparison indicates that relaxed ice structures may start to be competitive already at cluster sizes above $n = 90$.

1. Introduction

Atomic and molecular clusters are important objects of current experimental and theoretical studies, because they bridge the gap between single particles and bulk matter, across the nanoscale regime. Just in this most interesting size regime, structures and properties of clusters turn out to differ from the bulk and depend on cluster size in ways that are hard to predict.[1]

There are two main problems for the theoretical treatment of clusters. The usual chemical intuition for the structures of common molecules often fails for clusters. At the same time, finding the lowest-energy structure in an unbiased way turns out to be a problem that appears to scale exponentially with cluster size. Formal proofs of NP hardness have been proposed[2] but do not seem to be fully conclusive.[3] Nevertheless, many items of empirical evidence support this size scaling, starting with the historical counting of lcoal minima for Lennard-Jones (LJ) clusters by Hoare et al.[4]

This already disastrous scaling pertains for atomic clusters. In molecular clusters, each molecule has three additional orientational degrees of freedom. These additional degrees of freedom alone constitute yet another exponentially scaling optimization problem,[5] inseparably intertwined with the exponentially scaling positional optimization problem. Therefore, finding optimal structures for molecular clusters is even more difficult than for atomic clusters. Hence, global optimization of atomic clusters has always been done for considerably larger sizes than for molecular clusters. For example, LJ clusters are currently studied with unbiased global optimization methods up to about $n = 300$ particles,[6−9] but full, systematic optimization of water clusters has not significantly proceeded beyond $n = 30$ yet.[10]

Although deterministic global optimization methods are being improved substantially at present,[11−13] they are still not applicable to cluster sizes of practical interest. Therefore, stochastic-heuristic methods are needed. At the expense of the guarantee for finding the global minimum energy structure, they offer practical applicability for much larger clusters while still offering a good probability for finding global and lowest-energy local minima, if applied properly. Algorithms of this class used most often recently include simulated annealing[14] and its more recent cousins such as conformational space annealing,[6] basin-hopping,[15,16] and genetic algorithms (GA) or, more generally, evolutionary algorithms (EA) (see ref 1 for further discussion and literature citations).

Water clusters have been studied with unbiased global structure optimization methods several times (see, for example, refs 5 and 17 and further literature cited therein), mostly with the TIP4P empirical potential. For the smallest pure neutral clusters, there are direct connections to detailed experimental information from IR spectra.[18] Also, there are obvious connections to the role of water as solvent via the study of solvation clusters.[19,20] Indeed, this has grown into such a broad field that no full overview will be attempted here.

Although pure neutral TIP4P water clusters have become an important benchmark for molecular clusters,[21] and although TIP4P actually does surprisingly well for small water clusters (up to about $n = 12$) when compared to ab initio and experimental results, it turned out recently that TIP4P begins to fail qualitatively for slightly larger clusters: most notably, in the size region $n = 17−21$, where the transition from all-surface structures to molecule-centered ones occurs, strong disagreements were found[10] between structures based on TIP4P and structures based on the elaborate TTM2-F potential by Burnham and Xantheas.[22] This actually is a failure of TIP4P because the TTM2-F results were later confirmed by MP2 ab initio calculations[23] (see also ref 24). However, evaluation of TTM2-F energies and gradients costs a factor of about 20 more

* Corresponding author. E-mail: hartke@phc.uni-kiel.de.

10.1021/jp060512l CCC: $33.50 © 2006 American Chemical Society
Published on Web 04/01/2006

than for TIP4P, which exacerbates the global water cluster structure optimization problem further.

Therefore, already in ref 10 a parallel implementation of an EA approach running on 60 processors of a Cray T3E was needed to be able to do water cluster global structure optimization with TTM2-F up to $n = 30$. Although this implementation succeeded in parallelizing the most compute-intensive parts by distributing the many local optimization tasks, it did so by mapping the standard generational EA paradigm onto a master-slave model. This lead to an alteration between massively parallelized code sections with purely serial bottlenecks of significant lengths, which obviously is detrimental to the overall performance of the algorithm and constitutes a misuse of parallel computing resources.

In this paper, we report a considerably improved implementation. It largely realizes the same overall algorithm but completely eliminates serial bottlenecks and hence processor idle times by replacing the rigidly preordered generational model with a less structured and hence more flexible pool model. We are not sure if this is the first EA implementation of this type. For example, Ge and Head briefly mention a generation-free parallel GA being used in their work,[25] albeit without giving any details. In any case, such aspects appear to be absent from recent review works on parallel EA implementations (see, for example, ref 26, where the focus is on generational master-slave style implementations and on distributed subpopulations).

Therefore, we describe our new algorithm and its implementation in some detail in Section 2, in contrast to the old algorithm. For a real-life check, we have tested this implementation by applying it to pure neutral water clusters modeled by the full TTM2-F potential with flexible monomers, up to $n = 34$. To perform various direct comparisons with our previous generation-based master-slave implementation more quickly, we have also employed the TIP4P potential. The results of these two series of tests are described briefly in Section 3. Section 4 focuses on the TTM2-F global minimum structures, which partially extend and change our previous understanding of water clusters in this size range. This is followed in section 5 by a comparison of preliminary results for still larger water clusters (up to $n = 105$) with relaxed ice cutouts. The article ends in Section 6 with a summary of conclusions and future work.

2. Development of the New Algorithm

2.1. Old Generation-Based Algorithm. The global cluster structure optimization algorithm PHENIX used previously in our group is described in refs. 5 and 27. Full technical details can be found in these two publications. For reference in the following sections, a rough sketch is given below.

Two important ingredients of this algorithm are a phenotype crossover for generating new clusters and a niching concept to maintain a minimum level of structural diversity in the population (hence the acronym). In our work on Lennard-Jones clusters,[27] directed mutations proved to be valuable in exploring the immediate neighborhood of trial solutions in a nonrandom fashion. For molecular clusters, the individuals in the population consist of coordinate sixtuples (three Cartesian coordinates of the center of mass, and three Eulerian angles for the rotational orientation of the molecule with respect to a given reference orientation). For this case, it turned out that a global optimization of the Euler angles alone (with all positions fixed) as an additional step leads to improved performance.[5] All global optimization steps are followed by local optimizations using standard algorithms (conjugate gradient or quasi-Newton). These local optimizations take >95% of the total CPU time but also lead to substantial performance gains.

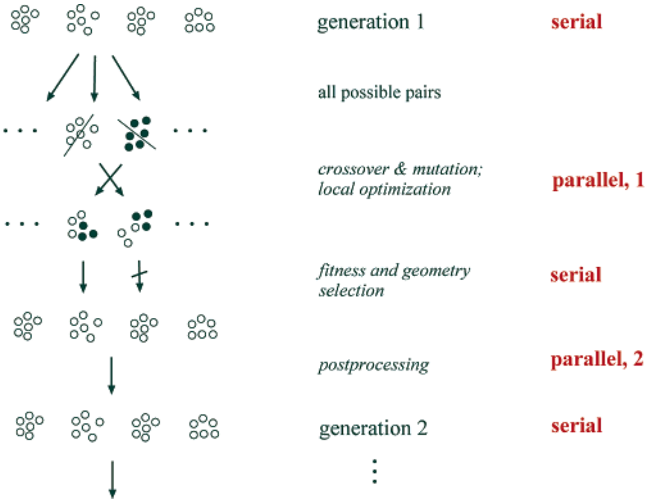


Figure 1. Scheme of the old parallelized generation-based algorithm.

This algorithm was originally run as one single serial process. However, because the local optimization of an individual after each operation is independent from all of the others (within a given stage), the algorithm can be parallelized easily on this coarse-grained level. The gains in speed are substantial because these local optimizations take almost all CPU time. Our first attempt at such a parallelization[10] consisted of a straighforward master−worker scheme. Although the master holds the complete list of data (the whole population), it distributes small portions of the whole problem to the workers. Essentially, the parallelization is mapped onto the problem to solve. For reference, we depict this implementation in Figure 1.

At startup, a set of n individuals is generated randomly, or individuals from an earlier run are read in. Then, the generational cycle begins with a mating phase, where all possible pairs of individuals are formed. With a population size of n, this leads to $n(n + 1)/2$ pairs (including self-pairing). Bundled tasks of work on these pairs (consisting of crossover, mutation, and local optimization with a certain convergence threshold) was distributed over m processes, constituting a first large parallel phase.

The results were collected by the master process into an interim list of new individuals. As an elite-like measure, the parents of the last generation were appended to this interim list. From this longer interim list, n new individuals were selected for the next generation, according to fitness (best energy) and geometry criteria (niching). The master process performed this selection serially and had to wait for all slave processes to finish their crossover work. Even worse, all slave processes had to wait for the master and for each other before the next parallel phase could start.

After selection, the n new individuals were subjected to postprocessing steps (directed mutation and global orientational optimization, described in detail in refs. 5, 10, and 27), which were again distributed in parallel, constituting a second, smaller parallel phase. Directed mutation puts a three-dimensional grid on the cluster. Molecules with the smallest contribution to the cluster total energy are then removed. Several attempts with different orientations are made to reintroduce these molecules into the centers of these boxes, trying to lower the total energy of the individual. If a following local optimization attains no improvement, then the number of molecules is increased and directed mutation is tried again. After a maximum number of unsuccessful attempts, the original individual is delivered back. Global orientational optimization is a full EA acting on the orientations of the molcules only, that is, the set of Eulerian

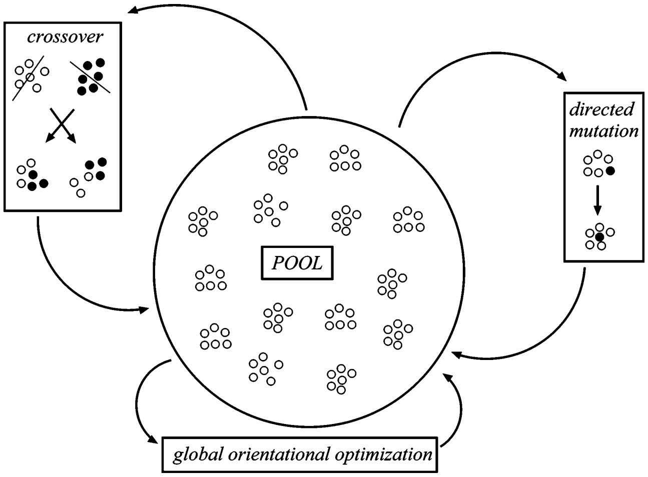


Figure 2. Sketch of the pool concept of our new parallel EA implementation.

angles of one cluster, with the positions of the centers kept fixed. This is followed by a local optimization and the original individual here also is delivered back if no improvement is attained. Local optimizations here are performed with a higher threshold than that used during the mating phase.

This was again followed by a serial collection and evaluation phase by the master. If an individual had lower total energy after orientational optimization, then it was chosen as the parent for the next generation; otherwise the result of the directed mutation was selected. Then a test for convergence or a counter of generations decided whether to quit the simulation or to start a next generation with the selected indivduals as parents.

Because the program was intended to run on supercomputers with shared memory architecture as well as on cluster computers that provide distributed memory, parallelization was implemented with the message passing interface (MPI). Static load balance was realized by choosing the number, n, of individuals and the number, m, of parallel processes such that either $n(n + 1)/2$ (the number of pairs of individuals during the crossover phase) or $2n$ (the number of individuals during the postprocessing phase) becomes an integer multiple of m. Obviously, this led to substantial restrictions in practice. Standard MPI send and receive commands were used to distribute new parallel tasks to processes that had completed their previous task. Although the rigid order of processes used in MPI could lead to imbalances,[28] for all but the smallest clusters the tasks are long enough to mask this behavior. Hence, a crude dynamic load balancing could be achieved. Nevertheless, strong imbalances in system load and long process idle times were visible (see section 3.3) simply because the CPU time needed for local optimization tasks of larger clusters varies very strongly and unpredictably with the starting geometries.

2.2. New Asynchronous Algorithm. Because a sufficiently accurate prediction of CPU times for local optimization of a given starting geometry presumably is either very difficult or very CPU intensive itself (or both), we decided to change the flowchart of the global optimization algorithm itself. It turned out that there was no need to abandon the simple master-slave scheme in favor of more complicated schemes with processes of equal rank. Instead, a more promising ansatz was mapping the problem onto the parallelization instead of mapping the parallelization onto the problem. Specifically, we abandoned the rigidity of the successive generational scheme of standard serial EAs. The general framework of the new asynchronous algorithm is depicted in Figure 2.

A population of n individuals held by the master acts as a pool of constant size. The genetic operators are applied by the slaves to the members of this pool. Crossover, orientational optimization, and directed mutation are the same as before, but now they are performed by the slaves with a certain probability (instead of in a certain succession). Selection and niching constituted the serial bottleneck of our old generation-based algorithm, but because they are essential they have to be retained in some fashion. Fortunately, their CPU time is insignificant compared to the local optimization tasks; therefore, we can afford to apply them not only once (after the crossover phase in the old algorithm) but now in every step.

From the existing population the master selects a number of individuals applying the energy and niching criteria. With the appropriate probability, a single individual or a pair of individuals are then picked at random from this selection and a copy is send to a slave to perform the desired operations. The resulting individuals are then returned to the master who checks them for similarity to the already existing ones in the pool. This is done by comparing the energy of the new individual with the energy of all of the others. Assuming that identical energies belong to identical individuals, a new individual is skipped if its energy already exists. Otherwise the new individual replaces a randomly chosen individual except the energetically best one. Hence, as a welcome side effect, the need for storing a significantly larger population built up from all possible pairs is eliminated (although, of course, all of those pairs are potentially still feasible).

3. Algorithm Verification and Comparison

3.1. Used Hardware and Comparabilty. For the comparison of these two algorithms, two series of simulation runs on two different types of computers were performed, a 22 node dual Opteron cluster at 2.2 GHz, using the Portland compiler, and a SGI altix 3700 with 128 Itanium 2 processors at 1.3 GHz with shared memory, using the Intel compiler. The first series employed the TTM2-F potential and was performed on the dual Opteron cluster using 20 processors on 10 nodes with 2 processors each. The second one used the TIP4P potential and was performed on the SGI altix using 32 processors.

Because the asynchronous algorithm permits crossover, directed mutation, and orientational optimization to take place with certain probabilities, these had to be set in a way that the number of operations is the same on average as that in the generation-based algorithm. Let *ngeo* be the number of clusters forming a population and *ngener* the number of generations. The generation-based algorithm is divided into two parallel phases in each generation. In the first phase, it does crossover with all cluster pairs including self-crossover; these are $1/2(ngeo{\cdot}ngeo + ngeo)$ operations. During postprocessing $2ngeo$ operations are executed. Hence, we arrive at a total of $ngener\,[1/2(ngeo{\cdot}ngeo + ngeo) + 2ngeo]$ operations during a whole simulation run. The new asynchronous algorithm does *niter* steps in total, with the operations crossover, directed mutation, and orientational optimization taking place with the probabilities $p(xover)$, $p(dirmut)$, and $p(oropt)$. The probabilites were chosen so that the following relations were fulfilled:

$$niter{\cdot}p(xover) \approx ngener \cdot 1/2(ngeo \cdot ngeo + ngeo)$$

$$niter{\cdot}p(dirmut) \approx ngener \cdot ngeo$$

$$niter{\cdot}p(oropt) \approx ngener \cdot ngeo$$

Local optimization after each of these operations was performed with the corresponding convergence thresholds, that

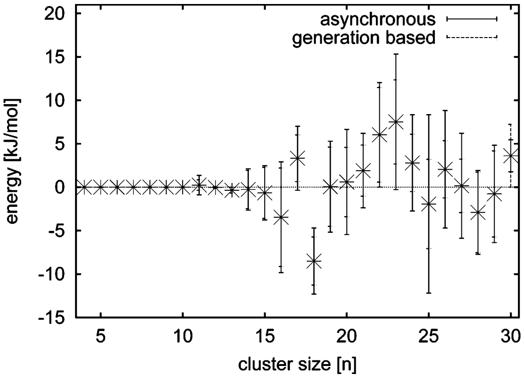


Figure 3. Differences of averaged final energy in kJ/mol vs cluster size *n* for the generation-based algorithm (20 generations) and for the asynchronous algorithm (5000 iterations) applied to pure neutral TTM2-F water clusters.

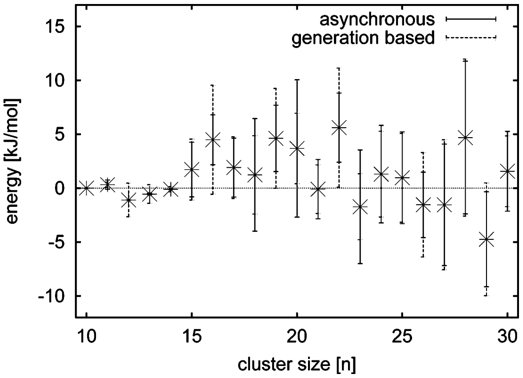


Figure 4. Differences of averaged final energy in kJ/mol vs cluster size *n* for the generation-based algorithm (80 generations) and for the asynchronous algorithm (20000 iterations) applied to pure neutral TIP4P water clusters.

is, a loose threshold after crossover and a tighter one after directed mutation and orientational optimization.

3.2. Performance of the Asynchronous Algorithm. Because local optimization with the TTM2-F potential is very time-consuming, we have used a very loose threshold of 10^{-1} for the relative energy convergence in the local TTM2-F optimizations following the crossover operations and a tight threshold of 10^{-8} in the optimizations following orientational optimization and directed mutation. The cluster size range $n = 4-30$ was covered with six simulation runs at each size.

With this setup, we have varied all of the standard GA parameters, in particular total run length, pool size (number of individuals), and crossover probability (coupled to amount of loose-threshold local optimizations). In all cases, the behavior of the new algorithm is in perfect accord with the generic behavior expected from standard generation-based algorithms. Therefore, we refrain from reporting all of these results here; a detailed account will be published elsewhere.[29]

Only one item was slightly unexpected: From these studies, we conclude that a population of 30 individuals is enough to reach good results for the cluster sizes examined here; larger populations produce no significant improvement. This is a surprising finding because the new algorithm lacks the larger intermediate population of the old algorithm.

3.3. Comparison between the Algorithms. Remembering the great expenses of time for local optimization employing the TTM2-F potential, we chose the same loose settings for energy convergence as mentioned in the beginning of section 3.2 to be able to run the tests more quickly. Additionally, the runs were limited to a population size of 20 individuals and a duration of 20 generations for the generation-based algorithm. According to the above, this corresponds to 5000 iterations for the asynchronous algorithm, or more specifically to 4200 crossover operations, attained by setting $p(xover) = 84\%$, and 400 directed mutations and orientational optimizations each, attained by setting $p(dirmut) = p(oropt) = 8\%$. Directed mutations were tried in 15^3 boxes while orientational optimization was performed on a population of 10 individuals lasting 10 generations. The threshold for local optimization following a crossover was set to 10^{-1}, and to 10^{-8} after directed mutation and orientational optimization. Cluster sizes ranged from $n = 4$ to $n = 30$ with six simulation runs at each size.

A second series of simulations used the TIP4P potential. Runs using the generation-based algorithm were done with a population size of 20 individuals, lasting 80 generations. With $p(xover) = 84\%$ and $p(dirmut) = p(oropt) = 8\%$ the number of iterations for the asynchronous algorithm was set to 20000. Directed mutations were also tried in 15^3 boxes but orientational optimization was performed on a population of 10 individuals lasting 100 generations. The threshold for local optimization after directed mutation and orientational optimization was kept at 10^{-8} and set to the usually applied value of 10^{-4} following a crossover. A cluster size range of $n = 10-30$ was also covered by six runs each.

Figures 3 and 4 directly compare final energies in kJ/mol versus cluster size for the old and the new algorithm. To emphasize the small differences in comparison to the statistical scatter, both plots show the differences between the average values of both data sets at each cluster size, while the length of the errorbars corresponds to the standard deviation. The order of entries in the legend reflects the order in the calculation of the difference: values belonging to the first entry were subtracted from those belonging to the second one.

As shown in Figure 3 for the TTM2-F potential, the final energies for cluster sizes $n \leq 10$ are the same for both algorithms. Results for $n \geq 11$ are in 12 cases greater than zero and in 9 cases smaller. Only at $n = 18$ the zero line is clearly not within the spread of the standard deviation; such an outlier result can be expected for these rather short test runs. Additionally, the standard deviations show similar spreads for both algorithms in all cases. Therefore, in general, the quality of the results does not depend on the underlying algorithm.

Figure 4 depicts the case of the TIP4P potential. The energy differences are in 12 cases greater than zero and in 8 cases smaller. Except for $n = 22$, the zero line is always inside the spread of the standard deviation. The overlap of the standard deviation spreads is significant and comparable to the case of the TTM2-F simulations. So the results not only do not depend on the algorithm but they are also independent from the potential.

Local optimizations are used many times throughout both global optimization algorithms and require the major part (>95%) of computer time in serial implementations. Therefore, these test calculations were speeded up considerably by using loose thresholds. The price to pay is that for some of the larger clusters we do not quite reach the best energies published previously.[10] However, this happens to the same degree for both the new and old implementation (neither one is consistently better in Figures 3 or 4, and even the spread of energies is very similar); therefore, comparability is retained.

Throughout the whole cluster size range tested, all of these findings prove that the new algorithm is able to produce the same results as the old one, within the usual variations to be expected from stochastic global optimization algorithms, for both

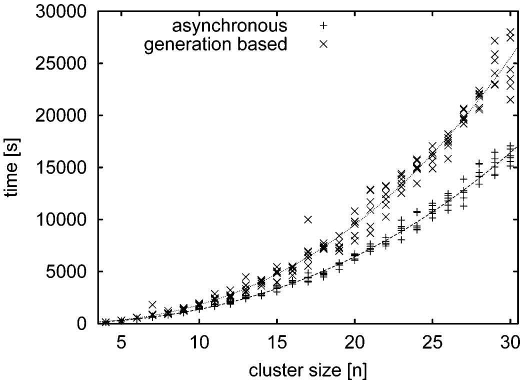


Figure 5. Run time in *s* vs cluster size *n* for the generation-based algorithm (20 generations) and the asynchronous algorithm (5000 iterations) using the TTM2-F potential.

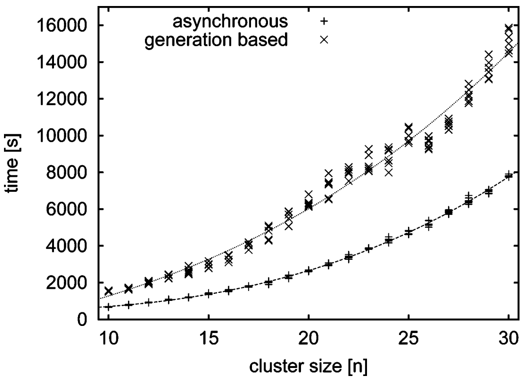


Figure 6. Run time in *s* vs cluster size *n* for the generation-based algorithm (80 generations) and the asynchronous algorithm (20000 iterations) using the TIP4P potential.

potentials tested here. As discussed above, the removal of serial bottlenecks in the new asynchronous variant should have a measurable influence on the run time of the program. Therefore, we also examined how long each algorithm took to complete all of the steps. In Figures 5 and 6 the total run time versus the number of steps is plotted for the two different potentials together with third-order polynomial fits.

In both cases, six runs were performed for each cluster size. The observable variability in run time for a given cluster size (due to the stochastic nature of both algorithms) increases with the size of the cluster. It reaches some thousands of seconds for the TTM2-F runs and approximately a thousand seconds for the TIP4P runs with the generation-based algorithm, and less than a thousand seconds with the new algorithm. This increase reflects the increasing time difference for local optimizations of different starting structures, with growing cluster size. For both potentials and algorithms we observe stochastic deviations from the fit curve. Additionally, in the case of the generation-based algorithm, all values for certain cluster sizes deviate collectively from the fit. This presumably reflects the effect of the number of operations not being an integer multiple of the number of processes. Collapsing this scatter to a mean value, we observe a speedup gained from the usage of the new asynchronous algorithm, which takes on values up to 50%.

Figure 7 shows that the speed improvement is due to the complete elimination of processor idle times. For the old algorithm, "dips" in the overall load occur periodically, indicating processor idle times. Because of the forced synchronizations, at the end of these dips almost all processors are waiting, therefore these "dips" can take away up to 50% of the overall system load. From the description of the new algorithm in section 2.2, it is clear that these "dips" have now been replaced by productive generation of new solution attempts.

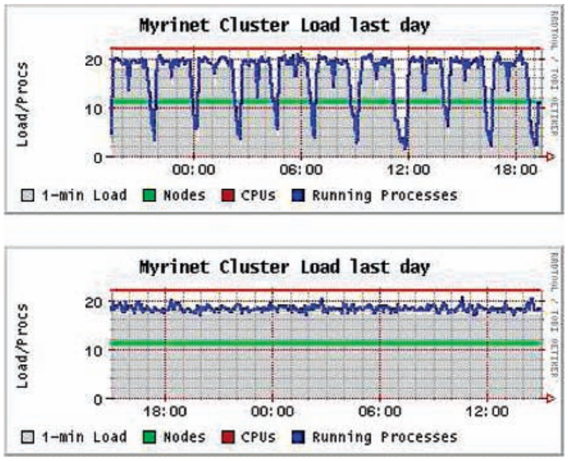


Figure 7. System load on an Athlon cluster vs time; top: generation-based algorithm, saw-tooth-like dips in the load indicate processors waiting for the next synchronization event; bottom: asynchronous algorithm; all dips in the system load have been eliminated.

4. Global Minimum Energy Structures

Until this point the focus of this work was completely on the technical realization and first tests of this new implementation. Furthermore, we have reduced the global convergence abilities of the algorithms by a weak local optimization convergence threshold (see above). Nevertheless, we are confident that the old and new algorithm produce results that are essentially identical (as far as this is possible for stochastic algorithms). The main difference is that the new algorithm does this substantially faster.

Therefore, we have also started production runs using the standard (tighter) local optimization convergence thresholds (10^{-4} after crossover, 10^{-8} after directed mutation and orientational optimization) employed in our previous study.[10] In these runs, we have already managed to discover at least five improvements over the previously published best structures for $(H_2O)_n$ within the TTM2-F model, namely, for the sizes $n =$ 22, 24, 27, 29, and 30. From the remaining cases of $n \leq 30$, all of the previous results could be reproduced. Only for $n = 28$ longer runs and/or a tighter convergence threshold would have been necessary. Successions of (putative) global minimum structures of water clusters have been presented and discussed several times before (also in our previous work, ref 10). Nevertheless, we again present a complete series of structures starting with $n = 6$, mainly because some of the new structures we have found partially change the structural interpretations and transitions we have presented in our previous work.

Experiment and theory agree upon simple, homodromic rings as the best structures for the trimer, tetramer, and pentamer (with the latter two being slightly bent out of plane at one edge). The first interesting case is the hexamer because at least at low temperatures the homodromic (quasi-)planar six-ring is not the global minimum, but a three-dimensional structure (cf. Figure 8), either the cage or the prism (depending on the empirical potential used or on the level of ab initio theory, and even on the inclusion of zero-point vibrational energy[30]). In any case, a decisive observation that carries through to far larger clusters is that four-rings and five-rings appear to be energetically more favorable than six-rings (or three-rings). The hexamer prism

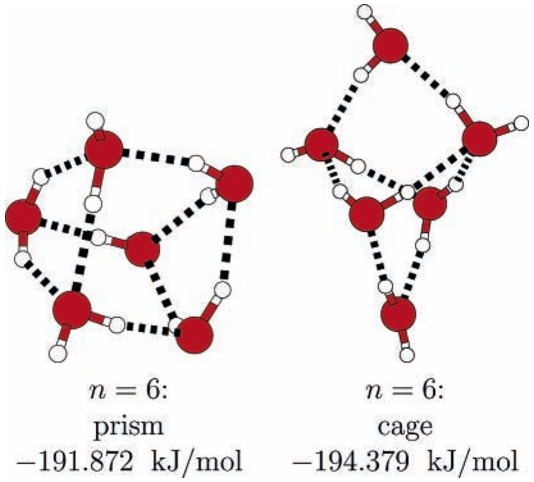


Figure 8. TTM2-F minimum structures for $(H_2O)_6$. The cage is the global minimum for this potential (not including zero-point energy).

contains three four-rings and two three-rings, while the cage can be viewed as consisting of four (annealed) four-rings.

All larger water cluster structures are three-dimensional. Apart from a preference for cubes and pentagonal prisms, previous authors have found it hard to discern any obvious building blocks or buildup sequences for these clusters. However, at least in the size range $n = 6-13$, the best structures for the odd-numbered clusters $n = 2m + 1$, $m = 3$, 4, 5, and 6, can always be understood as insertion of an edge-bridging water molecule into the best structure for the smaller even-numbered case $n = 2m$ (cf. Figures 9−11). Interestingly, this rule also includes the transition from $n = 6$ to $n = 7$, for *both* the cage and the prism. Allowing for some very small rearrangements, this rule can be complemented by recognizing that the transition from each odd-numbered structure $n = 2m + 1$ to the next larger even-numbered structure $n = 2m + 2$ can be effected by adding another bridging water next to the first one. Equivalently, one can get from an even-numbered structure with $n = 2m$ to the next one with $n = 2m + 2$ by insertion of a water dimer. This then also includes the case $n = 12 \rightarrow 14$.

This rather simple succession also encompasses the step from $n = 16$ to $n = 18$ (cf. Figure 12), but it already appears to be first broken by $n = 15$, which is a stack of two pentaprisms that cannot be derived from the best structure for $n = 14$ by attaching an edge-bridging water molecule. In fact, upon closer examination, the above rules also do not include the transition from the single pentaprism at $n = 10$ to the stack of two cubes at $n = 12$ (which repeats upon going from $n = 14$ to $n = 16$). One may attempt to "save" these rules for the latter two cases by a small extension: The $n = 12$ structure may also be viewed as a stack of three squares. If two opposite bonds of the equatorial square are broken, then one arrives at a stack of two six-rings. These, in turn, can be constructed from the $n = 10$ pentaprism by inserting a water dimer.

However, this line of rationalization starts to get quite convoluted at this point. Therefore, it is more helpful to again include the more general observation that cubes and pentaprisms are the preferred building blocks (which is consistent with the above observation that rings with four and five members are preferred). The above rules apply for small clusters, until enough water molecules are present to form these building blocks. Also, clearly, an odd-numbered cluster cannot be built by assembling only cubes and/or pentaprisms, so the above edge-capping rule is still realized in these cases. Starting from $n = 8$, however, even-numbered clusters appear to realize the requirement of the

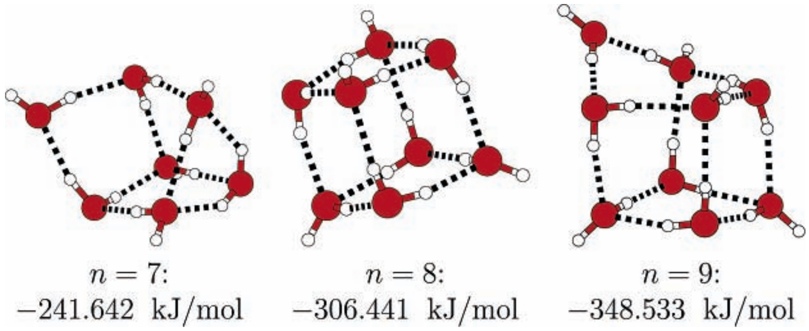


Figure 9. Global TTM2-F minimum structures for $(H_2O)_n$, $n = 7−9$.

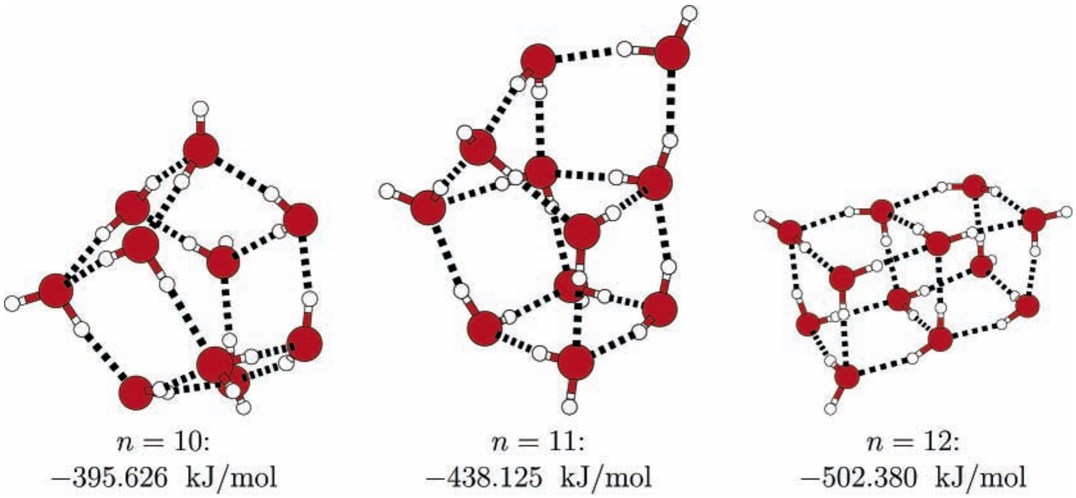


Figure 10. Global TTM2-F minimum structures for $(H_2O)_n$, $n = 10−12$.

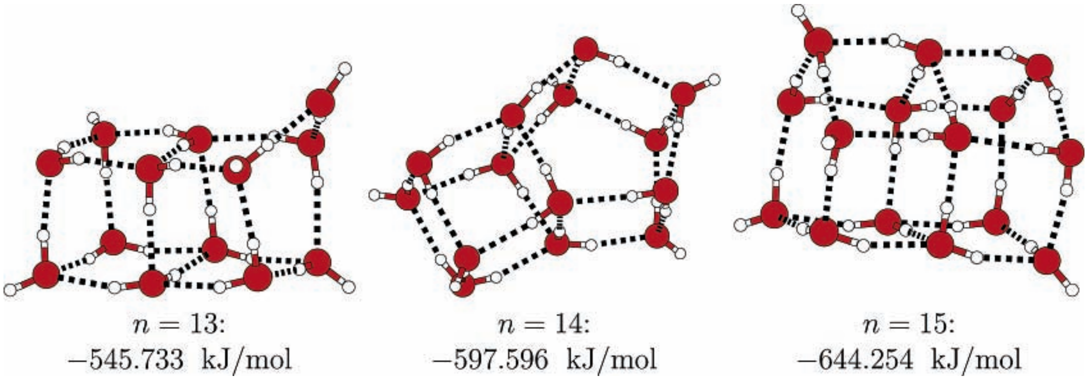


Figure 11. Global TTM2-F minimum structures for $(H_2O)_n$, $n = 13-15$.

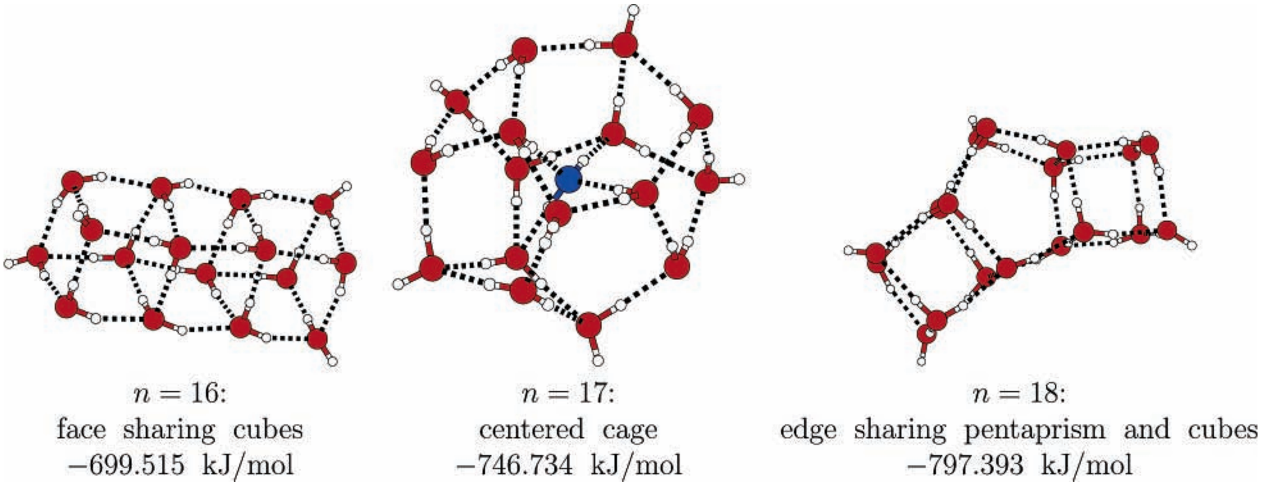


Figure 12. Global TTM2-F minimum structures for $(H_2O)_n$, $n = 16-18$.

best structure being formed by the smallest number of cubes and/or pentaprisms. This rule for the even-numbered cases then actually holds all the way up to $n = 24$ (see below), for our new improved set of TTM2-F minimum structures.

The first case that is definitely not classifiable by the above rules or any simple modifications of them occurs for the 17mer (Figure 12): With the TTM2-F potential, it gives rise to a clathrate-like cage with one internal molecule (shown in blue in Figure 12). No cubes or pentaprisms are visible in this structure. At this point we should also re-emphasize a main finding from our previous work:[10] With the TIP4P potential, the structure shown here is not the global minimum, but a qualitatively very different one that does contain at least two clearly identifiable cubes. Additionally, no molecule can be viewed as being in the interior of this TIP4P cluster. MP2 calculations[23] have clearly confirmed the expectation that the much more complicated (and computationally expensive) TTM2-F potential produces the correct result in this situation. Although this confirms the qualitative failure of TIP4P, the error is not as gross as it may seem: Both structures are stable minima for both potentials, and in both cases the energy difference is only a few kJ/mol.[10] These small differencs indicate that the appealing TTM2-F cage structure with an interior molecule is not an overwhelming necessity for $n = 17$ but rather the chance result of a delicate balance of several opposing factors, presumably including the preference for cubes and pentaprisms on one hand and the superior stability due to the first occurrence of a central four-coordinated molecule on the other hand (which is clearly less strained than four-coordinated molecules at the surface).

Four-, five-, and six-membered rings are the common pattern of the surfaces of larger clusters; occasionaly larger rings also occur. As soon as one allows for the possiblity of nonconvex surfaces, classifying water molecules in clusters as surface or interior becomes a matter of somewhat arbitrary definitions. In Figure 13, we show two typical six-membered surface rings in a section from a larger cluster, together with interior molecules directly below the ring centers, marked in blue. Clearly, with both rings, one would also have the freedom of classifying them as “dips” in the surface, formed by two five-membered rings including the blue molecule. Because the 17mer as a global minimum shows two such six-membered rings at its surface, we tend to accept all six-membered rings as surface structures. Furthermore, because seven-membered and larger surface rings/dips leave more room for molecules bridging them without undue strain, we tend to question the global-minimum status of clusters containing such larger surface rings.

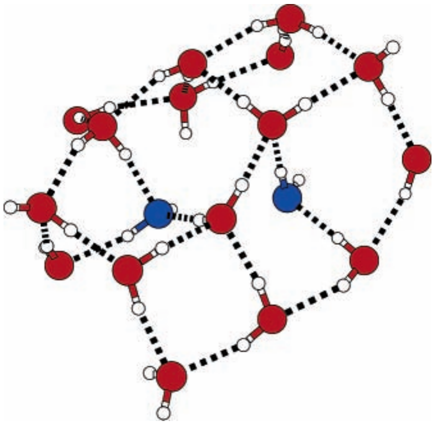
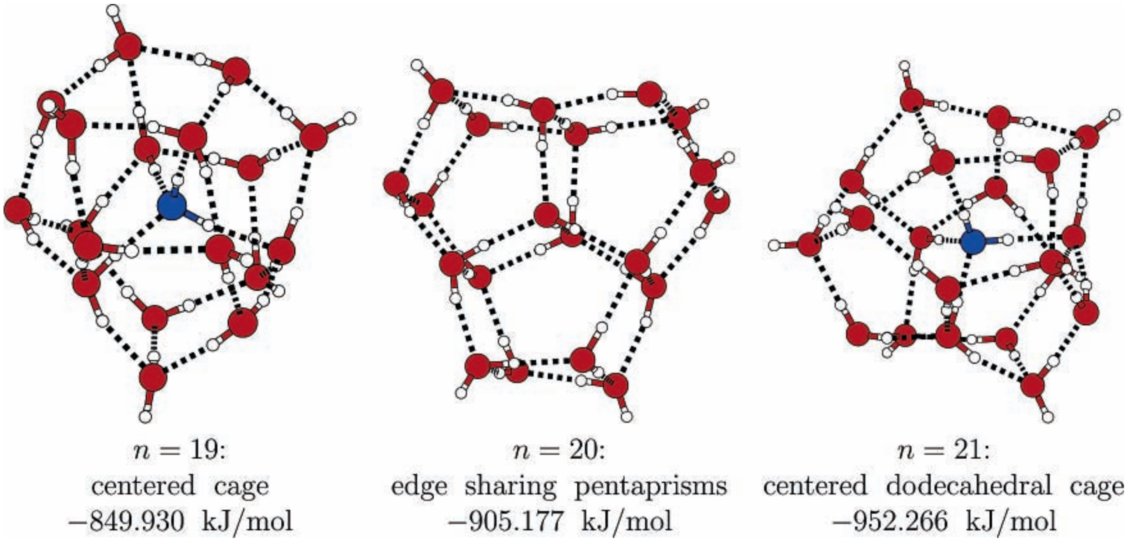


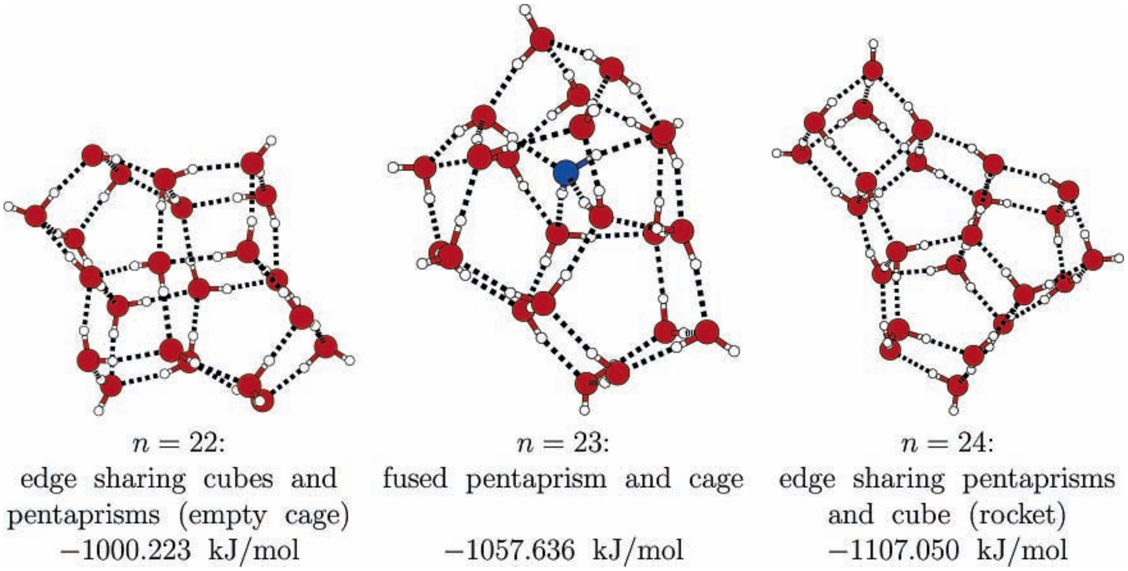
Figure 13. Two six-membered rings as part of a cluster surface.


$n = 19$:
centered cage
-849.930 kJ/mol

$n = 20$:
edge sharing pentaprisms
-905.177 kJ/mol

$n = 21$:
centered dodecahedral cage
-952.266 kJ/mol

Figure 14. Global TTM2-F minimum structures for $(H_2O)_n$, $n = 19-21$.


$n = 22$:
edge sharing cubes and pentaprisms (empty cage)
-1000.223 kJ/mol

$n = 23$:
fused pentaprism and cage
-1057.636 kJ/mol

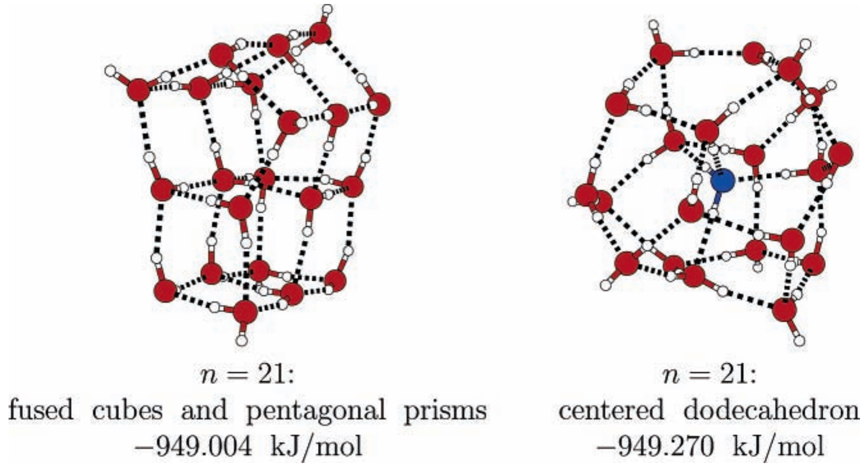
$n = 24$:
edge sharing pentaprisms and cube (rocket)
-1107.050 kJ/mol

Figure 15. Global TTM2-F minimum structures for $(H_2O)_n$, $n = 22-24$.

If we accept that at $n = 17$ a cage with a single interior molecule (narrowly) beats the otherwise expected singly edge-capped arrangements of cubes or pentaprisms, then the following development toward larger clusters appears quite logical: At $n = 17$, the water network is barely large enough to close around the central molecule, in fact it can do so only at the expense of exhibiting two less favorable six-rings. Therefore, for $n = 19$, 21, 23,... these structures should be energetically even more favored compared to the cube-and-pentaprism structures. However, for even-numbered structures, arrangements of cubes and pentaprisms may still be better for some of the larger cluster sizes because in these cases we need not add an edge-capping water molecule (which contributes less to the total energy because it is only 2-coordinated). This is exactly our observation (cf. Figures14 and 15): We observe a strict even−odd alternation of cubes/pentaprisms with singly filled cages. In our previous work,[10] we had seen this alternation only for the size range $n = 17-21$, with the even-numbered clusters also switching to cages above this size range. With the improved global minimum structures this “switching” is observed throughout the size range $n = 17-25$. The qualitative implication of this observation is that the transition from clusters with all molecules on the surface to clusters with interior molecules cannot be said to occur at a certain size but rather is smeared out over a rather wide size range.

The following is a side note of caution for other researchers in this area: Even subtle distinctions in energy can be accompanied by great differences in geometrical structure. This is proved by the 21mer whose global minimum is a centered dodecahedral cage. A slightly different centered dodecahedron is found at $E = -949.270$ kJ/mol, but a totally different geometry of two fused cubes and two fused pentaprisms is located at $E = -949.004$ kJ/mol, which differs by only 0.03% (cf. Figure 16). Note that the relative difference in energy between this dodecahedron with $E = -949.270$ kJ/mol and the global minimum with $E = -952.266$ kJ/mol is 0.3% and hence also quite small.

Examining this cube-and-pentaprisms $n = 21$ local minimum structure more closely, another lesson can be learned: In the equator (at the borderline between one pentaprism and one cube), one H bond is “broken”. It turns out that it is “topologically” impossible to close this bond without breaking another one elsewhere. This is not a peculiarity of this particular cluster structure, but rather one example for a more general phenomenon: Water molecules can be arranged in isolated cubes and pentaprisms with one H bond along each edge. Likewise, these



$n = 21$:
fused cubes and pentagonal prisms
-949.004 kJ/mol

$n = 21$:
centered dodecahedron
-949.270 kJ/mol

Figure 16. Two low-energy local TTM2-F minima for $(H_2O)_{21}$.

cubes and pentaprisms can be "stacked" (or fused at common faces) in one direction (viz. $n = 12, 15, 16, 18$), forming quasi-one-dimensional strings, and still there can be exactly one h bond along each edge. However, as soon as stacking occurs into more than one direction (not only for completely filled quasi-two-dimensional arrangements, but already for simple isolated branches), restrictions appear at the branching points: If branching involves only pentaprisms, as, for example, in the $n = 24$ global minimum structure (cf. Figure 15), then only three pentaprisms (and correspondingly only three edges on each surface) meet at such a branching point. Therefore, we can still have one H bond along each edge, which is why there is a H bond linking the two central water molecules at the branching point in the $n = 24$ structure. In contrast, when branching involves cubes (or a mixture of cubes and prisms, as in $n = 22$), four polyhedra (corresponding to four edges on each surface) can meet at one point, leading to five edges meeting at this point (including the edge perpendicular to the stacking directions). Because one water molecule can have at most four H bonds (without incurring excessive energy penalties), it is impossible in such a situation to still have one H bond along each edge. This is why the central H bond is missing in $n = 22$ (as opposed to $n = 24$) and also why one H bond is missing in the cubes-and-pentaprisms local minimum of $n = 21$. (Because of the missing central bond connecting the two layers, the $n = 22$ structure could also be classified as an "empty cage"; however, this is probably misleading because the layers are too close to each other for another molecule to fit in between).

As noted above, the $n = 17$ cage is "tight" and has two six-rings in the surface. The looser and more favorable $n = 19$ cage can be built with one six-ring and two four-rings. Consequently, in the $n = 21$ cage we have no six-rings anymore (and no four-rings), only five-rings. Formally, this is a dodecahedral cage, but a distorted one, due to the four H bonds linking the central molecule with the cage. Clathrate-like dodecahedral water molecule cages can also form around a variety of other host atoms and molecules (neutral and ionic). Hence, they have been conjectured to be responsible for magic numbers, for example, in alkali cation microhydration clusters, a hypothesis that was tested and replaced recently.[31]

Because of this special role of the dodecahedral cage, one could expect that larger water clusters may form by edge-capping the dodecahedral cage at the outside. As already diagnosed in ref 10, this is not the case; instead, the cage at $n = 23$ is formed by extending the $n = 19$ cage by face-capping the six-ring with a four-ring, forming a local protrusion similar to a pentagonal prism (cf. Figure 15).

As already remarked above, at $n = 24$ we find the last member of the structural family built purely from cubes and pentaprisms. Building larger clusters in that fashion presumably entails one of two penalties: As discussed above in connection with $n = 21$, this build-up pattern has its restrictions. Continuing to aim at structures with exactly one H bond along each edge, the resulting structures would have to consist of branched networks ($n = 24$ being one such structure with one branching point) with a fractal dimension between 1 and 2 and a correspondingly large surface-to-volume ratio or, equivalently, a large amount of surface strain. Conversely, attempts to stack cubes and pentaprisms to filled two- or three-dimensional arrangements lead to structures where along some edges (on the surface and/or in the interior) there can be no H bonds; instead, along such an edge, there is an unfavorable interaction between neighboring water molecules. Additionally, while cubes can be stacked in a 3D space-filling fashion, this fails as soon as pentaprisms with fivefold (quasi-)symmetry axes are involved. In this way, it is not surprising that at this size the transition to cages is complete.

As a specific example of this line of argument, at $n = 27$ one would have the theoretical possibility of reconciling cubes as building blocks with cages by constructing a structure composed of only eight cubes, arranged in a two-by-two-by-two pattern, formally containing exactly one interior molecule in the center. As analyzed above, however, this would lead to several branching points where five edges meet and to a central point with six edges meeting. Thus, the inherent strain of the cuboid arrangement (where all bond angles strongly deviate from the ideal one for water) cannot be compensated by a correspondingly high number of H bonds. Therefore, we have not observed a structure like that.

For larger clusters, we find only filled cages, and the new cage growth pattern just described continues (cf. Figures 17 and 18). In contrast to our findings in ref 10, the new series of global minima including our improved ones does not exhibit cuboid protrusions anymore but merely pentaprismatic ones. Very interestingly, we now also see cages with two pentaprisms protruding from them. For the special case of $n = 29$, these two pentaprisms are exactly opposite to each other, leading to a surprisingly high symmetry of the O-atom scaffold. In all of these cases ($n = 23, 25, 26, 27, 29$), the pentaprismatic protrusions are not true pentaprisms because one of their H bonds (the one toward the cage center) is broken, which is why we call this an extension of the cage rather than an outside attachment.

For $n \leq 27$, our improved global minimum cage structures contain one interior molecule (in contrast to our preliminary

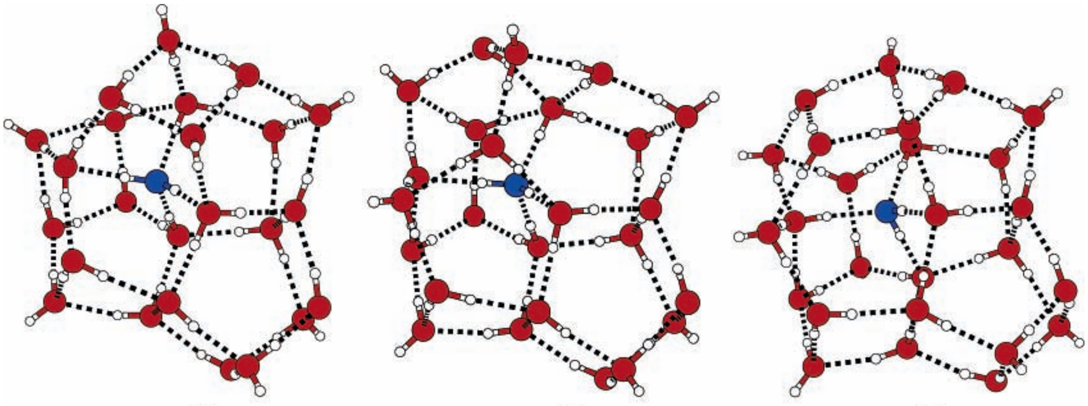


Figure 17. Global TTM2-F minimum structures for $(H_2O)_n$, $n = 25-27$.

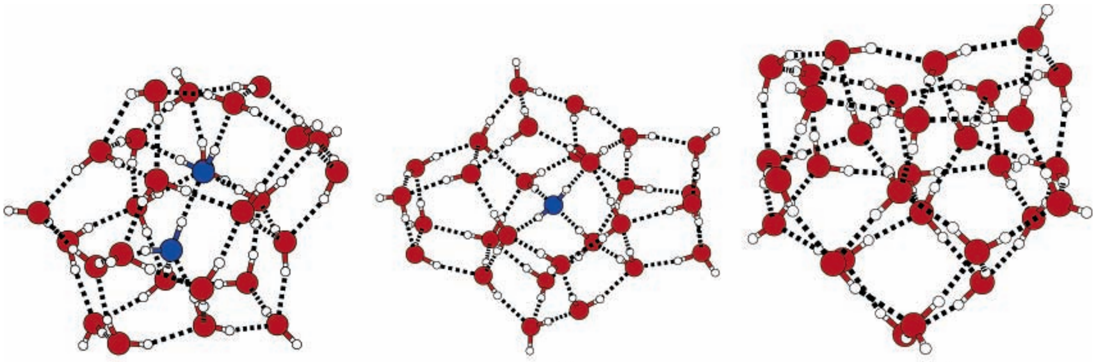


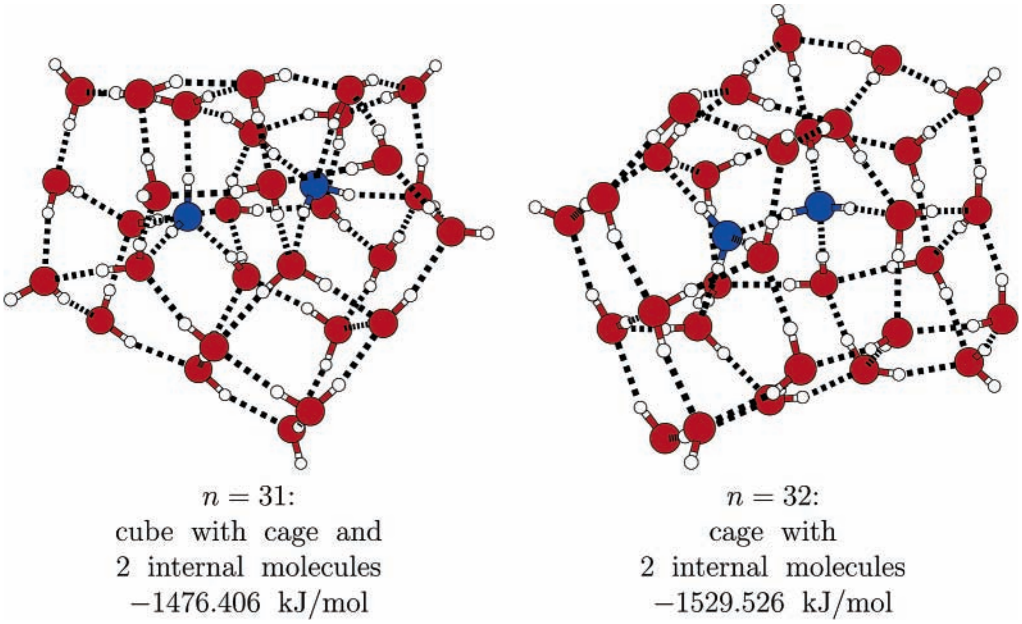
Figure 18. Global TTM2-F minimum structures for $(H_2O)_n$, $n = 28-30$.

finding of two internal molecules in ref 10). The surface of the 28mer contains a three-ring and two six-rings (possibly indicating some strain, as for $n = 17$), and now there are enough molecules present to build a cage with two internal molecules. The 29mer contains only one internal molecule, while between $n = 30$ and $n = 34$ the number of internal molecules varies from zero to two: In case of the 30mer there is no internal molecule, but inside the 31mer again two molecules are contained (cf. Figures 19 and 20). Both clusters have cuboid protrusions instead of pentaprismatic ones as they occur for smaller cages. With the 32mer, there follows again a cage structure with two internal molecules, whereas the 33mer is an irregular empty cage and the 34mer again contains two internal molecules. Interestingly, from all of our proposed global minimum structures with two internal molecules, only for $n = 28$ and $n = 32$ there is an H bond between the two, although it is hard to imagine how to continue building larger clusters without H bonds between the interior molecules. These findings support the conclusion that the transition toward two internal molecules is again smeared out, similar to the one toward one molecule in the range of $n = 17-25$.

We refrain from analyzing this second transition in more detail because we see several indications of not having found the global minimum structures for $n \geq 30$ yet. The 30mer and the 32mer exhibit large dips in their surfaces. In both cases, there are two molecules at the bottom of these dips. In case of the 30mer, the rim of the dip is an eight-ring, and it is a seven-ring in case of the 32mer. As remarked above, to our experience surface rings of these sizes are often signatures of nonoptimal structures. Furthermore, note that the energies for $n = 32$, 33, 34 are visibly above the linear trend in Figure 21 below, which confirms that they are probably not yet good enough.

In the traditional plot of energy versus inverse cluster radius (cf. Figure 21), a linear trend seems to be visible already for $n \geq 12$. However, as should be evident from the discussion above, this is by no means a signature of bulk structures. On the contrary, upon magnification, the data exhibit an irregular pattern of steps and oscillations around the apparent linear trend, more in line with the varying trends described above.

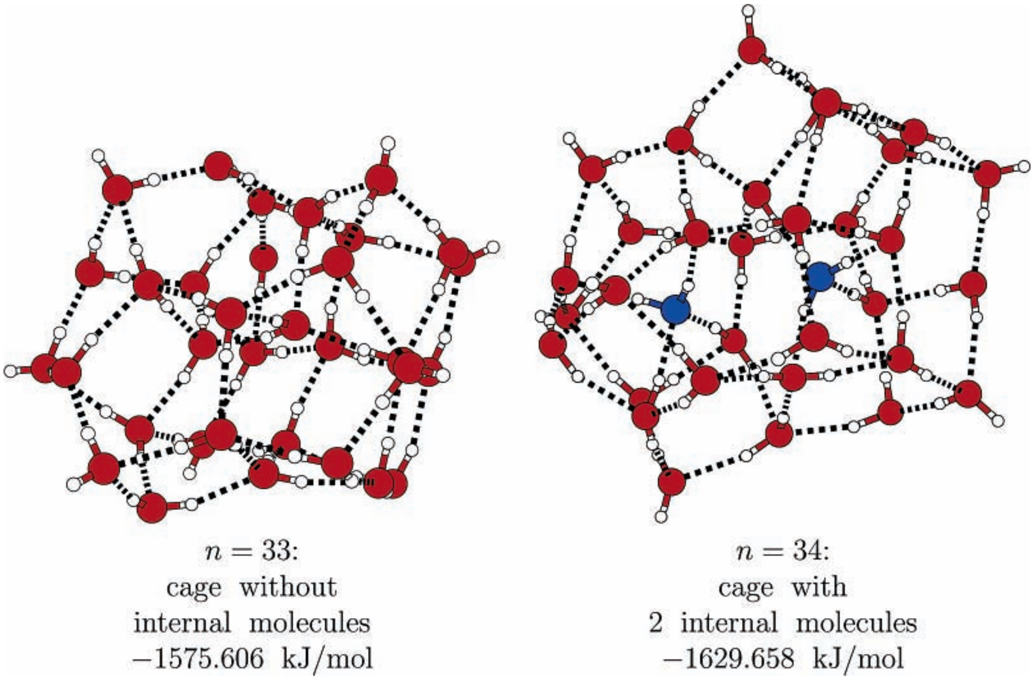
The stability function (cf. Figure 22) is often used to assign magic number status to a given cluster size. However, with the possible exceptions of $n = 8$ and $n = 12$, no single cluster size is really convincingly outstanding in this plot. Nevertheless, besides confirming our suspicion that $n = 33$ and 34 probably are not yet close enough to the true global minimum energies, it does allow for some weaker observations. For example, none of the early cages $n = 17$, 19, or 21 stand out as particularly stable, including the famous dodecahedral cage at $n = 21$. Instead, the neighboring structures $n = 16$ and 20 built from cages and pentaprisms fare better. Nevertheless, the cage



$n = 31$:
cube with cage and
2 internal molecules
-1476.406 kJ/mol

$n = 32$:
cage with
2 internal molecules
-1529.526 kJ/mol

Figure 19. Global TTM2-F minimum structures for $(H_2O)_n$, $n = 31-32$.



$n = 33$:
cage without
internal molecules
-1575.606 kJ/mol

$n = 34$:
cage with
2 internal molecules
-1629.658 kJ/mol

Figure 20. Global TTM2-F minimum structures for $(H_2O)_n$, $n = 33-34$.

stabilities are rising toward larger clusters, culminating in $n = 23$, which turns out to be more stable than its immediate neighbors $n = 22$ and 24 that are built from cages and pentaprisms. As evidenced by the decrease toward a local stability minimum at $n = 28$, attaching appendages to the outside of cages does not lead to good structures. Finally, the dips at $n = 30$ and $n = 33$ confirm our suspicion that also at these sizes we have probably not seen the true global minimum structure yet.

5. Comparison with Relaxed Ice Structures

In a recent study,[32] an attempt was made to find water clusters with a crystalline core using the TIP4P potential and a certain protocol that mixed several approximate optimization strategies, for several amorphous and crystalline input structures at a few isolated cluster sizes. Although smaller input structures with $n = 48$ and $n = 123$ lost their crystallinity during this process, at a cluster size of $n = 293$ a crystal-like core was obtained inside an amorphous hull, which matches up nicely with structural expectations from other theoretical studies and experiments.

The global minimum cluster structures presented in Section 4 of this paper definitely cannot contain ice-like features, simply because up to $n = 34$ we observe a maximum number of two internal molecules, and, as expected, surface molecules are subject to significant deviations from ice structures upon optimization (see below). However, we already have preliminary results from extending our global optimization approach systematically up to approximately $n = 100$. We are sure that these results are far from converged yet because the total operation count in each of these runs for larger clusters has been smaller than in those for the largest clusters we have shown explicitly in Section 4 above. Nevertheless, it is tempting to compare these

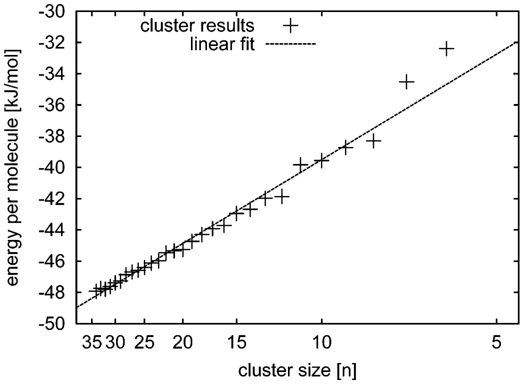


Figure 21. Total energy in kJ/mol per molecule vs number of molecules.

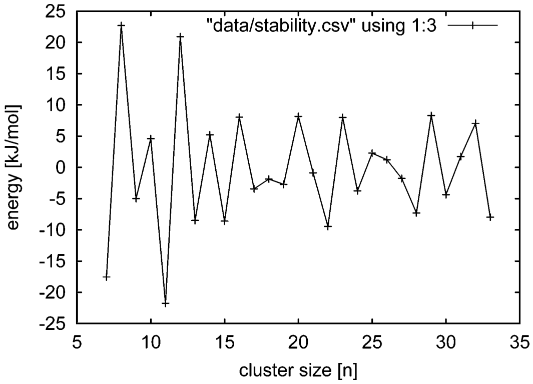


Figure 22. Stability function: $E(n - 1) + E(n + 1) - 2E(n)$ vs cluster size n.

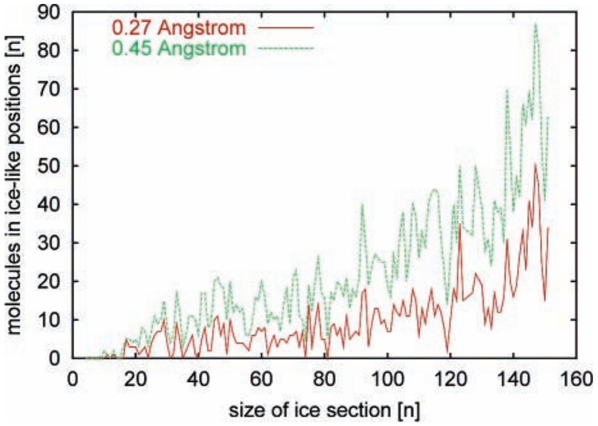


Figure 23. Number of molecules in ice-like positions vs total number of molecules contained in the originial spherical section of ice Ic.

first systematic global optimization results for larger water clusters to relaxed ice structures of the corresponding sizes. While this is not going to provide reliable answers, it does provide fairly tight bounds and reliable qualitative insight because the most general and most reliable characteristic of EA minimization methods is their quick decrease in energy during the very first stages.

For this comparison, we have generated spherical ice sections in the size range from $n = 4$ up to $n = 150$, for six different ice modifications, employing the TTM2-F potential in purely local optimization mode. Specifically, spherical sections of the chosen modifications of ice had to be prepared first. This was done in a similar manner as suggested in ref 32. Specifications of the crystal structures of the ice modifications Ic,[33] Ih,[34] VI, VIII,[35] and IX[36] were obtained from the Inorganic Crystal Structure Database.[37] A spherical volume was chosen large enough to contain ~1000 molecules. Then the center of mass of these sections was calculated and all molecules were sorted by their distance to this center of mass. The first n molecules ($n = 4$,..., 150) were chosen successively to build a spherical section of ice.

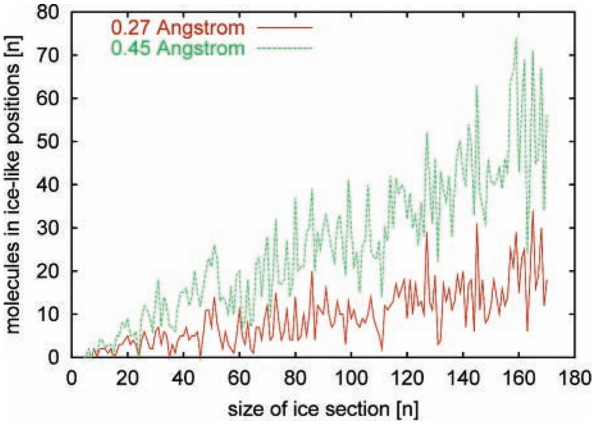


Figure 24. Number of molecules in ice-like positions vs number of molecules contained in the originial spherical section of ice Ih.

Each of these unrelaxed cutouts was then relaxed with the TTM2-F potential, in two steps. First, a purely orientational but global optimization was performed, with the O-atoms kept fixed, to obtain a reasonable hydrogen-bond network. This was done with an EA that resembles the generation-based variant of our phenix algorithm. The ice sections resulting from this procedure then underwent a single local optimization with a tight threshold of 10^{-8}, as used in our global cluster optimization after orientational optimization or directed mutation. This threshold is the best compromise because looser thresholds yield worse results and tighter thresholds do not necessarily produce better results for the cost of extended CPU time.

After this relaxation, all of these ice cutouts exhibit considerable reconstructions in their outermost surface layers. If the cutouts are too small, then these reconstructions affect the whole system, resulting in a cluster that can only be categorized as amorphous. Clearly, these findings in themselves already put a lower bound on the size of the first ice-like cluster. For an easy, automatic determination of the size of the ice-like core (if any) in these relaxed ice sections, we have used the following protocol: Employing the impose routine of the TINKER[38] program package, the root-mean-square deviation (RMSD) of all molecules in the cutout was calculated, relative to the original ice lattice. However, it turned out that the total RMSD value was not sensitive enough to characterize ice content because of its averaging over all molecules. Therefore, the molecules in the system were categorized according to their individual contribution to the total RMSD. All molecules with a displacement smaller than a given threshold were accepted as located in a position in accordance with the initial ice lattice. This is an unbiased method of abstracting displaced surface molecules, with the advantage of avoiding arbitrary definitions of surface layer thickness. Because protons are disordered in the chosen modifications, Ic, Ih, and VI, the process of superimposition was restricted to the skeleton of O atoms of each pair of structures.

In Figures 23 and 24, the numbers of ice-like molecules (identified in the above way as located in positions according to the parent modification of ice) are plotted versus the number of molecules of the initial unrelaxed spherical section of the ice crystal, for two examples of all ice modifications studied here in this way. As expected, these numbers of ice-like

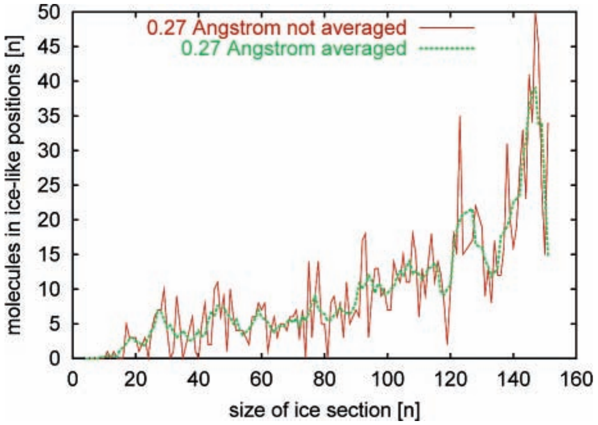


Figure 25. Number of molecules sited in ice position vs number of molecules contained in the originial spherical section of ice Ic and the same data provided with a moving average.

molecules increase both with the size of the initial cutout (i.e., from left to right in each figure) and with a growing threshold for accepting displacments from original ice crystal positions as still ice-like (as shown by the two different data sets in each figure). We would have also expected small oscillations superimposed on the former monotonic trends because taking successively larger cutouts from an ice crystal can easily lead to more or less favorably positioned molecules at the outer fringes. In all cases, however, these oscillations are larger than expected. To quote an arbitrary medium-sized example, for ice Ic at a moderate threshold value of 0.27 Å (∼10% of the average O−O distance; lower curve in Figure 23), from an initial cutout of 123 molecules, an inner core of 35 molecules is ice-like after relaxation. If the cutout contains only 1 molecule more or less, then the ice-like core shrinks to only 15 molecules. Accordingly, in each curve, a few sizes with particularly large ice-like cores stick out, similar to magic numbers; in the above example of ice Ic at a threshold of 0.27 Å; these are $n = 123$ with a core of 35 molecules and $n = 147$ with a core of 50 molecules (and possibly also a few smaller values of n). A comparison between the two figures quickly shows that such peaks occur at different locations and that the whole oscillation patterns are largely different. (Similar plots for the ice modifications VI, VIII, and IX are qualitatively similar but quantitatively different.)

Of course, some caution is advisable here. Clearly, our cutout and relaxation procedures have been chosen arbitrarily, and furthermore the first step of the relaxation employed a nondeterministic algorithm, for which (because of the size of these cutouts) we cannot claim to have reached global convergence in the larger cases. We expect changes in this procedure (or merely repetitions of the same procedure) to yield different oscillation patterns. For more meaningful results, one would have to average over a large ensemble of such curves (possibly also including small random initial displacements of the molecules from their ice positions), which would lead to these oscillations being averaged out to some (unknown) extent. Such an averaging was simulated in Figure 25 by a moving average with a spread of five cluster sizes. In comparison to the original lower curve from Figure 23, the peaks are broadened but still resemble the characteristics of the underlying original curve. In any case, this oscillative behavior indicates that focusing on single, isolated cluster sizes, as was done in ref 32, may lead to less reliable results than a survey of several broader size intervals.

As argued above, further analysis of these data should yield lower bounds for the first occurrence of ice cores in water clusters. However, as the strong differences between the two curves in each of the Figures 23 and 24 indicate, these lower bounds depend strongly on the degree of similarity threshold value used to accept structures as ice-like. Furthermore, they also depend on an arbitrary definition of how large an ice-like arrangement has to be as such. There are no definite answers to these questions. The number of molecules in a single unit cell of a given ice modification appears to be a reasonable minimal size. Taking this at face value, however, the cuboid octamer is the smallest crystal of ice Ic. Clearly, a more reasonable definition should, for example, claim that beyond such a minimal cluster size, the size of the ice-like core should not drop significantly below the size of the unit cell. Again taking the threshold value of 0.27 Å; (∼10% of the average O−O distance) as an arbitrary but perhaps reasonable measure, the smallest cluster with an ice-like core should be expected at $n \approx 95$ for ice Ic and for ice Ih.

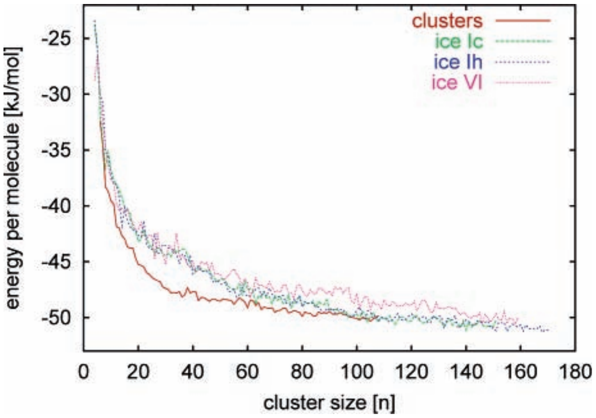


Figure 26. Energy in kJ/mol per molecule vs number of molecules of optimized clusters and locally optimized spherical sections of ices Ic, Ih, and VI. The cluster results beyond $n = 34$ are not claimed to be global minima.

These predictions can be refined (and perhaps also made less arbitrary) by checking at which size an unbiased global structure optimization fails to find a water cluster lower in energy than those generated by the above ice-cutout procedure. For this purpose, in Figure 26, we have plotted energies per molecule vs number of molecules, for spherical sections of ices Ic, Ih, and IV (as generated above) up to about $n = 150$, in direct comparison to our preliminary global optimization results for water clusters up to $n = 105$ (which we do not claim to have reached global convergence beyond $n = 34$, see the discussion above). Clearly, our cluster results are well below all ice-cutout results at least up to $n \approx 60$, and still almost strictly below up to $n \approx 90$. Because our results are strict upper bounds for the true global minimum energies, and because we have so far failed to locate any ice structures by the above matching procedure in our globally optimized structures, the transition to clusters with ice-like cores should not be expected below $n = 90$.

Performing unbiased global cluster structure optimization at cluster sizes of $n \approx 90$ with any reasonable chance for obtaining a final best structure that is reasonably close to the global minimum, both in energy and structure, is extremely expensive. Therefore, we plan to speed up (but also partially bias) our future searches by including ice cutouts into the initial guess pool. As shown in this section, this should be done not for isolated cluster sizes, but for several neighboring sizes, to avoid hitting cluster sizes that can accommodate unusually large or small ice cores.

6. Conclusions and Future Work

With this work, we have established, analyzed, and tested a generation-free parallel EA implementation for global cluster structure optimization. By replacing the traditional generation concept with a new pool concept, operated upon by an arbitrary number of unsynchronized parallel processes, it completely eliminates serial bottelnecks and the processor idle times associated with them. This reduces the wall-clock times by about a factor of 2, without affecting the quality of the results and without changing our previous successful algorithm to any significant extent. At the same time, this conceptual change in implementation makes the algorithm efficient for arbitrary numbers of parallel processes, severing previously important links between population size and number of processes.

Clearly, this enables us to treat larger clusters than before in a much more flexible manner. As evidence of this, we could not only reproduce our previous results for TTM2-F water clusters up to $n = 30$ but even improve some of them and extend the results to $n = 34$. This has led to modifications of our earlier results, in particular to the prediction of an unusually broad and strictly even−odd-oscillative size range $n = 17-24$ for the transition from all-surface structures to cages with one interior molecule, and to the proposal of a similarly oscillative and broad transition from one to two interior molecules in the size range $n = 28-34$.

Applications to still larger TTM2-F water clusters are underway, with the aim of checking and extending the TIP4P results by Buch et al.[32] on the first occurrence of ice cores in water clusters. Comparisons between our first preliminary results (up to $n = 105$) and relaxed sections from several ice modifications indicate that the onset of ice cores is difficult to pin down unambiguously, but a lower bound of $n \approx 90$ may tentatively be extracted from our data.

Acknowledgment. We thank the computing center of the University of Kiel and the high-performance computing center of northern Germany (HLRN) for technical assistance and for generous amounts of computer time.

References and Notes

(1) Hartke, B. *Angew. Chem., Int. Ed.* **2002**, *41*, 1468.

(2) Wille, L. T.; Vennik, J. *J. Phys. A* **1985**, *18*, L419.

(3) Srivastav, A., University of Kiel. Personal communication.

(4) (a) Hoare, M. R.; Pal, P. *Adv. Phys.* **1971**, *20* 161. (b) Hoare, M. R. *Adv. Chem. Phys.* **1979**, *40*, 49.

(5) Hartke, B. *Z. Phys. Chem.* **2000**, *214*, 1251.

(6) Lee, J.; Lee, I.-H.; Lee, J. *Phys. Rev. Lett.* **2003**, *91*, 080201.

(7) Shao, X.; Jiang, H.; Cai, W. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 193.

(8) Pullan, W. *J. Comput. Chem.* **2005**, *26*, 899.

(9) Gregor, T.; Car, R. *Chem. Phys. Lett.* **2005**, *412*, 125.

(10) Hartke, B. *Phys. Chem. Chem. Phys.* **2003**, *5*, 275.

(11) Horst, R.; Pardalos, P. M.; Thoai, N. V. *Introduction to Global Optimization*, 2nd ed.; Kluwer: Dordrecht, The Netherlands, 2000.

(12) Neumaier, A. Complete Search in Continuous Global Optimization and Constraint Satisfaction. In *Acta Numerica 2004*; Iserles, A., Ed.; Cambridge University Press, 2004; pp 271−369.

(13) Neumaier, A.; Shcherbina, O.; Huyer, W.; Vinko, T. *Math. Programming B* **2005**, *103*, 335−356.

(14) Kirkpatrick, S.; Gellat, C. D., Jr.; Vecchi, M. P. *Science* **1983**, *220*, 671.

(15) Li, Z.; Scheraga, H. A. *Proc. Natl. Acad. Sci. U.S.A.* **1987**, *84*, 6611.

(16) Wales, D. J.; Scheraga, H. A. *Science* **1999**, *285*, 1368.

(17) Wales, D. J.; Hodges, M. P. *Chem. Phys. Lett.* **1998**, *286*, 65.

(18) Sadlej, J.; Buch, V.; Kazimirski, J. K.; Buck, U. *J. Phys. Chem. A* **1999**, *103*, 4933.

(19) Hartke, B.; Charvat, A.; Reich, M.; Abel, B. *J. Chem. Phys.* **2002**, *116*, 3588.

(20) Schulz, F.; Hartke, B. *Chem. Phys. Chem.* **2002**, *3*, 101.

(21) Wales, D. J.; Doye, J. P. K.; Dullweber, A.; Naumkin, F. Y. The Cambridge Cluster Database, http://brian.ch.cam.ac.uk/CCD.html, 1997.

(22) Burnham, C. J.; Xantheas, S. S. *J. Chem. Phys.* **2002**, *116*, 5115.

(23) Lagutschenkov, A.; Fanourgakis, G. S.; Niedner-Schatteburg, G.; Xantheas, S. S. *J. Chem. Phys.* **2005**, *122*, 194310.

(24) Lenz, A.; Ojamäe, L. *Chem. Phys. Lett.* **2006**, *418*, 361.

(25) Ge, Y.; Head, J. D. *Chem. Phys. Lett.* **2004**, *398*, 107.

(26) Cantu-Paz, E. *Efficient and Accurate Parallel Genetic Algorithms*; Kluwer Academic Publishers: Boston, MA, 2001.

(27) Hartke, B. *J. Comput. Chem.* **1999**, *20*, 1752.

(28) Sanders, P.; Worsch, T. *Parallele Programmierung mit MPI*; Logos, 1997.

(29) Bandow, B. Ph.D. Thesis, University of Kiel, 2006.

(30) (a) Gregory, J. K.; Clary, D. C. *J. Phys. Chem.* **1996**, *100*, 18014. (b) Liu, K.; Brown, M. G.; Carter, C.; Saykally, R. J.; Gregory, J. K.; Clary, D. C. *Nature* **1996**, *381*, 501.

(31) Schulz, F.; Hartke, B. *Theor. Chem. Acc.* **2005**, *114*, 357.

(32) Kazimirski, J. K.; Buch, V. *J. Phys. Chem. A* **2003**, *107*, 9762.

(33) Arnold, G. P.; Finch, E. D.; Rabideau, S. W.; Wenzel, R. G. *J. Chem. Phys.* **1968**, *49*, 4365.

(34) Peterson, S. W.; Levy, H. A. *Acta Crystallogr.* **1957**, *10*, 70.

(35) Kuhs, W. F.; Finney, J. L.; Vettier, C.; Bliss, D. V. *J. Chem. Phys.* **1984**, *81*, 3612.

(36) la Placa, S. J.; Hamilton, W. C.; Kamb, B.; Prakash, A. *J. Chem. Phys.* **1973**, *58*, 567.

(37) Inorganic Crystal Struture Database (ICSD): http://www.fiz-karlsruhe.de/ecid/Internet/en/DB/icsd/ or: http://icsdweb.fiz-karlsruhe.de/

(38) Ponder, J. W. *TINKER*, Software Tools for Molecular Design; Washington University: St. Louis, MO; http://dasher.wustl.edu