Master Thesis

Minimization of Lennard-Jones clusters using multi-scaling methods from molecular dynamics

Author(s):
Gonnet, Pedro G.

Publication Date:
2003

Permanent Link:
https://doi.org/10.3929/ethz-a-004521822

Rights / License:
In Copyright - Non-Commercial Use Permitted

This page was generated automatically upon download from the ETH Zurich Research Collection. For more information please consult the Terms of use.
Diplomarbeit: Minimization of Lennard-Jones Clusters using Multi-Scaling Methods from Molecular Dynamics

Pedro G. Gonnet (pedro@vis.ethz.ch)

February 17, 2003
Chapter 1

Introduction

In this chapter we present a brief introduction to the problem of global energy minimization in Lennard-Jones clusters. Previous work is discussed and an overview of the methods presented herein is given.

1.1 Minimization of Lennard-Jones Clusters

Similar to pseudo-prime factorization in mathematics and computer algebra, Lennard-Jones cluster minimization is considered a “grand-challenge” problem in the domain of large-scale global minimization algorithms for problems with multiple local minima.

A Lennard-Jones cluster is a group of particles interacting with each other according to the pairwise potential energy function

$$E_{ij} = 4\epsilon \left( \left( \frac{\sigma}{r_{ij}} \right)^{12} - \left( \frac{\sigma}{r_{ij}} \right)^{6} \right)$$

where $r_{ij}$ is the Euclidian distance between the particles indexed with $i$ and $j$. The potential function has it’s minimum of $-\epsilon$ at $r = 2^{1/6}\sigma$. For practical purposes, we use the reduced units $\sigma = 1$ and $\epsilon = \frac{1}{4}$. The shape of the resulting interaction potential is shown in Figure 1.1.

The total energy of a particle cluster is given by the sum of all pairwise interaction energies between all particle pairs

$$E_{tot} = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} E_{ij}$$

where $N$ is the total number of particles in the cluster. A minimum of the total energy function is a set of particle positions for which the gradient with respect to the particle positions – i.e. the force on each particle – is zero.

The challenge associated with Lennard-Jones cluster minimization is the sheer number of local minima ($O(e^N)$ as is suggested in Tsai & Jordan (1993))
and the large energy barriers separating them. Although gradient descent over the energy landscape $E_{tot}$ finds local minima rather efficiently, there is no deterministic algorithm to find the global minimum for any given cluster.

1.2 Previous Work

Since the first systematic effort at Lennard-Jones cluster minimization (Northby 1987), much work has been published on the subject covering a number of different approaches for finding the global minimum of the potential energy landscape.

Since following the energy gradient of any given configuration toward the nearest local minimum is rather inexpensive, most methods focus on the construction of “good” initial configurations which lead to the global minimum (or a near-global minimum) once gradient descent is used.

In Northby (1987), initial configurations are created by placing particles randomly on an icosahedral lattice and consequently relaxing the cluster using gradient descent methods. This approach was applied to clusters ranging from 13 to 147 atoms, yielding results that have mostly remained unbeaten (Wales, Doye, Dullweber, Hodges, Naumkin, Calvo, Hernandez-Rojas & Middleton 2002).

An extension to the icosahedral lattice approach is given in Romero, Barrón & Gómez (1999), where clusters ranging from 148 to 309 particles are minimized using a combination of lattices and concepts from genetic programming, where two “good” solutions are crossed, mutated and minimized in the hope of getting a new, better solution.

In Foreman, Phillips, Rosen & Dill (1999) a method is presented which
interpolates a quadratic hyper-surface over a set of local minima. The global minimum is then assumed to lie near the minimum of the interpolated surface.

Locatelli & Schoen (2002a) present yet another method in which an extended potential function, which also satisfies some density and shape criteria, is minimized. The resulting minimal configurations are then used as initial conditions for the minimization of the “real” potential energy landscape. This approach is an extension of the modified potential approach presented in Locatelli & Schoen (2002b).

Almost all the above mentioned methods additionally use simulated annealing and Monte-Carlo type iterations or variations thereof to try to improve on minima once they have been found. This amounts to an extended exploration of the area surrounding a supposed global minimum and is not specific to any of the methods, yet in most cases necessary to achieve good convergence (Locatelli & Schoen 2002a).

1.3 New Approaches

In this diploma thesis we shall explore three new approaches to finding global minimal potential energy Lennard-Jones clusters:

- **Gaussian Process Interpolation**: Given a set of initial configurations, the potential energy surface is inferred by a Gaussian Process over which the minimum is then sought and added to the set of initial configurations.

- **SOM Coarse-Graining**: A coarse-grained model of the cluster is computed using a self-organizing map (SOM, described in Kohonen (1995)). The coarse-grained model is then minimized/updated from which a new, fine-grained model is then created.

- **Dimensionality Up-Scaling**: To avoid energy barriers separating local minima, the dimensionality of the problem is increased. Once a minimum has been found in a higher dimensions, the problem is “squeezed” back to three dimensions.

These three approaches all have one theme in common: the problem of Lennard-Jones clusters is abstracted to a model in which it can be solved efficiently. The solution in the model is then re-applied to the real problem in the hopes of getting a better configuration (Figure 1.2).
Figure 1.2: Iterative scheme for model-based problem solving.
Chapter 2

Gaussian Process Interpolation

In this chapter we describe Gaussian Processes in general and the strategy used for the potential energy surface interpolation. A Gaussian Process is used to model the potential energy landscape of a model problem consisting of three particles in two dimensions. The results of this experiment are discussed.

2.1 Introduction to Gaussian Processes

A Gaussian Process can be viewed as a regression method, given a set of \( N \) known vectors \( \mathbf{X} = \{ \mathbf{x}_i \}_{i=1}^N \) and their function values \( \mathbf{t} = \{ t_i = f(x_i) \}_{i=1}^N \) for some non-linear function \( f(\mathbf{x}) \). This interpolation is done by means of logical inference using

\[
P(t_{N+1} | \mathbf{t}, \mathbf{X}, \mathbf{x}_{N+1}) = \frac{P(t_{N+1}, \mathbf{t} | \mathbf{X}, \mathbf{x}_{N+1})}{P(\mathbf{t}, \mathbf{X})}
\]

This is done by creating a covariance matrix \( \mathbf{C} \) of the input vectors

\[
C_{ij} = C(x_i, x_j) = \theta_1 \cdot \exp \left[ -\frac{(x_i - x_j)^2}{4r^2} \right] + \theta_2
\]

where \( \theta_1, \theta_2 \) and \( r \) are the so-called hyper-parameters of the Gaussian Process. The predicted mean of \( t_{N+1} \) can then be calculated as

\[
\mu = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{t}
\]

where \( k_i = C(x_i, x_{N+1}) \). The hyper-parameters \( \theta_1, \theta_2 \) and \( r \) are chosen as to maximize the probability of these parameters given the set of known data vectors \( P(\{\theta_1, \theta_2, r\} | \{\mathbf{X}, \mathbf{t}\}) \), which amounts to minimizing the logarithm of the later:
\[-\frac{1}{2} \log \det(C) - \frac{1}{2} t^T C^{-1} t - \frac{N}{2} \log 2\pi\]

over \(\theta_1, \theta_2\) and \(r\).


### 2.2 Application to Lennard-Jones Cluster Minimization

The Gaussian Process can be used in a way similar to that in Foreman et al. (1999), yet with a more expressive interpolation scheme.

The search for the global minimum is performed iteratively: Given a set of initial random configurations, the Gaussian Process is trained – i.e. its hyper-parameters adjusted – from the initial particle positions and the resulting potential energies. From this trained Gaussian Process, the point with the minimum expected value is taken, evaluated and added to the set of initial points with it’s “real” potential energy (as opposed to the predicted potential energy).

The parameters of the Gaussian Process are simply the concatenation of the particle position vectors. For simplicity and faster convergence (and to avoid translational symmetries), the first particle was assumed to be at the origin. Furthermore, the system is bounded in space to avoid exploring too disperse configurations.

Three different objective functions were fed to the Gaussian Process interpolator/minimizer:

- **Bare potential energy:** The sum of the pairwise Lennard-Jones potential energies.

- **“Soft” potential energy:** Same as the Lennard-Jones potential, yet for the range 0…1 we use a third-degree polynomial approximation leveling off at some value \(E_{\text{max}}\) at \(r = 0\) and having the same derivatives as the Lennard-Jones potential with respect to \(r\) at \(r = 1\) (Figure 2.1). The resulting equation has the form:

  \[
  E_{\text{soft}}(r) = \begin{cases} 
  (2E_{\text{max}} - 6)r^3 + (6 - 3E_{\text{max}})r^2 + E_{\text{max}}, & \text{if } r \leq 1 \\
  \frac{1}{r} - \frac{1}{r}, & \text{otherwise}
  \end{cases}
  \]

- **Local well depth:** The potential energy at any configuration is the local minima found by following the steepest gradient from that point: \(E_{\text{local}}(x) = \min_x E(x)\). This is similar to the energy function used in the “basin-hopping” approach in Wales & Doyle (1997).
2.3 Results

The three different objective functions were tested on a system of three particles in two dimensions. The first particle being the origin, this results in four unknowns (x and y dimensions of the second and third particles) over which the potential energy is minimized.

Of the three objective functions, only the “soft” potential produced meaningful results. The “bare” Lennard-Jones potential failed due to the huge energy differences caused by the energy barrier at $r < 1$. This made the minima of the individual interactions disappear, mainly due to numerical precision.

The local well depth potential function failed due to the fact that it is not continuous and that different points could have the exact same function values. This caused singularities in the matrix operations involved in the Gaussian Process interpolation.

The “soft” potential energy objective function was used with $E_{max} = 10$. The convergence is much worst than what could even be expected by a random walk algorithm (Figure 2.2).

2.4 Discussion

The main reason behind the failure of the Gaussian Process interpolation to converge toward the global minimum (or any minimum) of the potential energy function is it’s tendency to spot false minima in areas where the potential energy surface is almost completely flat. These false minima are indeed real minima in the Gaussian Process, yet they are the result of attempting to interpolate a
vast almost flat surface with “non-flat” basis functions.

The problems encountered using Gaussian Process interpolation are very probably inherent to any multi-dimensional interpolation scheme. The biggest problem itself lies in the shape of the Lennard-Jones potential: the minimum of the potential function is stuck between an energy barrier and a rather flat surface, causing any interpolation scheme which is not lucky enough to sample the minimum at $2^{1/6}\sigma$ to infer a smooth, exponential-type curve ignoring the actual minimum.

Furthermore, it is the goal of most interpolation schemes to try to avoid noise and, unfortunately, in the case of the Lennard-Jones potential function, that’s just what the potential energy minima look like.
Chapter 3

SOM Coarse-Graining

In this chapter we describe the general framework of coarse-graining using a self-organizing map (SOM) to model certain characteristics of the underlying particles. The coarse-grained particles (SOM-nodes) are then updated as to minimize the underlying system. This approach is tested on a system of 100 particles approximated by 10 coarse-grained nodes and the results thereof are discussed.

3.1 General Framework

Coarse-Graining has always been a hot topic when it comes to large simulations of liquids or other homogeneous particle systems. The idea is quite simple: instead of simulating a system of, say, 100 microscopic particles, run the simulation using only 10 coarse-grained particles, each behaving as if they were 10 microscopic particles. The characteristics and interactions of the coarse-grained particles are derived either analytically or empirically to best reproduce the macroscopic characteristics of interest. A good overview of the topic is given in both Flekkøy, Coveyey & Fabritiis (2000) and Serrano & Español (2001).

Unfortunately, all the work on coarse-graining available refers to dynamic systems (i.e. describing equations of motion), whereas we are trying to use it for a static system. It is therefore somewhat difficult to build-up on previous work.

For coarse graining, we use a Self Organizing Map (SOM) which automatically adapts itself to the underlying data – i.e. the microscopic particle positions and some other quantities. We then use the SOM-nodes as our coarse-grained particles.

The SOM algorithm we use is the Neural Gas (NG) Model (Martinetz & Schulten 1991). This model was already used in Kern (2002) to coarse-grain the dynamic simulation of two Lennard-Jones droplets in 2D.

The NG nodes are adjusted in the following way (Fritzke 1997):

1. Select a “signal” vector $\xi$ at random from the microscopic particles.
2. Order all SOM-nodes according to their Euclidean distance to $\xi$ such that $w_i$ is the $i^{th}$ closest node to $\xi$ and $k_i$ the rank of the $i^{th}$ SOM-node in that order.

3. Adjust the position of each SOM-node by

$$\Delta w_i = \epsilon_i \cdot h_\lambda(k_i) \cdot (\xi - w_i)$$

where $h_\lambda$ is

$$h_\lambda(k) = \exp(-k/\lambda_t)$$

where $\lambda_t$ and $\epsilon_t$ are time-dependent decaying parameters defined by

$$\lambda_t = \lambda_i (\lambda_f/\lambda_i)^{t/t_{max}}$$
$$\epsilon_t = \epsilon_i (\epsilon_f/\epsilon_i)^{t/t_{max}}$$

and where $\lambda_i$ and $\epsilon_i$ are initial and $\lambda_f$ and $\epsilon_f$ are final values of these parameters.

for all $t$ up to $t_{max}$. The values for $\lambda$ and $\epsilon$ used are those given by Fritzke (1997): $\lambda_i = 10, \lambda_f = 0.01, \epsilon_i = 0.5, \epsilon_f = 0.005$. For all results generated, $t_{max}$ was set to 50'000.

Once the coarse-grained nodes have been positioned, we need to decide how to update/minimize them. Since we have no analytically derived energy potential for the coarse-grained particles (SOM-nodes) – which could be minimized over the coarse-grained particle positions – we need to somehow map the energy or forces to the SOM-nodes. This is done by viewing the microscopic particles as six-dimensional vectors containing both the particle positions and the force vector thereon:

$$[x, y, z; f_x, f_y, f_z]$$

where $f_x$, $f_y$ and $f_z$ are the forces in the dimensions $x$, $y$ and $z$.

The only adaptation necessary to the NG-algorithm described above is that the Euclidian distance for the neighbouring criteria and SOM-node rank is measured only in the first three dimensions, that is, the particle positions. The SOM-nodes are then updated in all six dimensions.

What we get after placing the SOM over the microscopic particles can be viewed as a decomposition of the space and an averaging of the forces therein. Figure 3.1 shows the extreme case of two disjoint clusters represented by two SOM nodes. The interactions between particles in proximity of the same SOM node get canceled-out once assigned to the later, whereas the interactions of particles assigned to different SOM nodes contribute to the interaction between the two nodes.
Once the SOM-nodes have been adjusted to their final positions and forces, the mass of each node \( M_i \) can be calculated as the sum of the masses of all microscopic particles for which the node is the nearest neighbour:

\[
M_i = \sum_{j=1}^{N} nn(p_j, w_i) \cdot m_j
\]

where \( nn(p_j, w_i) \) is the nearest-neighbour function which returns 1 if \( w_i \) is the node closest to the microscopic particle \( p_j \) or 0 otherwise, and \( m_j \) is the mass of the \( j^{th} \) microscopic particle (for the Lennard-Jones clusters we assume unit mass for all particles \( m_i = 1 \)).

Given the position, mass and force on each coarse-grained particle, we can step them as in a normal molecular dynamics simulation (much as in energy minimization through simulated annealing):

\[
w_i^{t+\Delta t} = w_i^t + \frac{f(w_i^t)}{2M_i} \cdot \Delta t^2
\]

where \( f(w_i) \) is the force vector part of \( w_i \) and \( \Delta t \) is the time-step, which is a parameter of the coarse-graining iteration.

Once the coarse-grained system has been updated, we need only to transform it back to the real space, that is, re-create microscopic particles from the coarse-grained particles. This is done by viewing the coarse-grained particles as centers of microscopic particle distributions. Each microscopic particle is set by:

1. Select a coarse-grained particle \( w_i \) with probability

\[
P(w_i) = \frac{M_i}{\sum_j M_j}
\]

2. Generate the microscopic particle positions from a Gaussian distribution \((G(\mu, \sigma))\):
Figure 3.2: The coarse-graining iteration applied to Lennard-Jones clusters.

\[ p_x = w_{ix} + \mathcal{G} \left( 0, M_i^{2/3} / 2 \right) \]

and so on for each dimension, where \( p_x \) is the \( x \)-dimension component of the particle \( p \) and \( w_{ix} \) the \( x \)-dimension component of the \( i \)th coarse-grained particle.

This might seem an odd strategy to re-create the microscopic space, since it does not take into consideration any positional constraints, such as particle proximity. There is, however, no intelligent way of doing this.

An alternative to re-creating the microscopic particles would be to update the original particle positions according to the forces on the SOM-nodes nearest to them – somewhat the inverse of the coarse-graining procedure. This however would only result in moving stiff portions of the cluster around and would not introduce the flexibility necessary to surmount the energy barriers between the different local minimum configurations.

In Summary, the coarse-graining iteration used herein for Lennard-Jones cluster minimization has the following form (Figure 3.2):

1. Create a SOM over the microscopic particle positions and the forces thereon.
2. Calculate the mass for each coarse-grained particle.
3. Move each coarse-grained particle according to it’s force vector and mass.
4. Re-create the microscopic particles from the coarse-grained particles.
3.2 Results

The SOM coarse-graining was tested on a system of 100 microscopic particles and 10 coarse-grained particles/SOM-nodes. The update of the coarse-grained particles was done with $\Delta t = 1$. Snapshots of the first 9 iterations can be seen in Table 3.1.

Although the SOM coarse-graining displays a high tendency to aggregate (that is, to form a compact cluster), the total energy values do not seem to converge toward a global minimum (Figure 3.3).

3.3 Discussion

Although the clusters generated by the SOM coarse-graining display anti-diffusive behavior, they are not especially helpful for finding minimal-energy configura-
Figure 3.3: Convergence of the SOM coarse-graining using 100 microscopic particles and 10 coarse-grained particles. The time-step used was $\Delta t = 1$. The $\log_{10}$ of the total energy was used for the plot.

The large fluctuations in total potential energy is due almost exclusively to the pairwise interactions resulting from the randomized re-creation of the microscopic system in the last step of the coarse-graining iteration. As mentioned earlier, this is hard to avoid. In general, if we had a method for creating low-energy microscopic configurations from a set of coarse-grained nodes, one could solve the problem of optimal clusters by simply creating them using that method to create the initial cluster.

The main problem, however, lies in the philosophy of coarse-graining: the coarse-grained model is created such as to represent some mesoscopic or macroscopic properties of the underlying system. In the case of Lennard-Jones clusters, however, the bulk of the potential energy results from the fine detail of the microscopic pairwise interactions.
Chapter 4

Dimensionality Up-Scaling

In this chapter we present a method for circumventing the energy barriers between the different minima of the potential energy surface of Lennard-Jones clusters by adding extra dimensions to the problem space. This approach is tested in two variants on clusters of sizes 20 to 40 and the results thereof are discussed.

4.1 Introduction to Dimensionality Up-Scaling

The main problem with global energy minimization on energy landscapes with many local minima is that there is no easy way to move from one minimum to another. This movement is impeded by so-called “energy barriers” which prevent gradient solvers from escaping one minimum and moving to another, better one (these barriers, in fact, create these different local minima in the first place). It therefore seems reasonable to adopt strategies and/or models which would allow us to avoid such barriers.

The “basin-hopping” approach used in Wales & Doyle (1997) is an example of a modified potential energy function that flattens the barriers between minima

\[ \tilde{E}(x) = \min_x E(x) \]

yet also removes any gradient from one minimum to another, therefore creating a piece-wise constant surface on which minimization is non-trivial.

A different approach is given in Laio & Parrinello (2002), where Gaussian functions are added to the potential energy term as to fill-in local minima and eventually escape them.

A third approach follows the logic of the Knight in the game of chess: if any other figure is in it’s way, it just “jumps” over it. How does this apply to Lennard-Jones cluster minimization? Chess is essentially a two-dimensional problem with two-dimensional barriers between different configurations in the form of figures impeding other figures’ movements. The Knight avoids these
two-dimensional barriers by “jumping” in a third dimension. This concept was first applied in Gonnet (2002) to avoid knots during energy minimization in protein structure problems.

How this concept works when applied to Lennard-Jones clusters can be shown with the following toy problem of ten particles in two dimensions. Figure 4.1 shows the potential energy landscape as a function of the coordinates of the first particle. The global minimum is clearly the particle’s original position in the center of the 2D-cluster, this position, however, is unreachable from outside of the cluster.

We now add an additional degree of freedom to the problem: we allow the first particle to move in three dimensions. Figure 4.2 shows snapshots of the same potential energy landscape for different values of the $z$-coordinate of the first particle (four-dimensional plots are unfortunately still out of reach for print media).

The global minimum in the center of the cluster, unreachable in two dimensions, suddenly becomes “visible” from anywhere on the potential energy surface once the particle “lifts off” in 3D. This is due to the energy barriers only existing as such in two dimensions. In three dimensions, one simply moves around them instead of trying to climb over them.

4.2 Moving between Dimensions

Although we can solve a 2D problem in 3D rather easily, we must also be able to move the solution back to two dimensions. This can be done by two different approaches: Homotopy modeling or penalization.

In the penalization approach (described in Gonnet (2002)), we use a modified total energy function containing terms for the additional dimensions.
Figure 4.2: Snapshots of the same potential energy surface as in Figure 4.1, yet with different values for the z coordinate of the first particle. Notice how the different local minima merge as the z-coordinate is increased.

\[
E_{\text{tot}} = \sum_{i}^{N-1} \sum_{j=i+1}^{N} \left( \frac{1}{r_{ij}^{12}} - \frac{1}{r_{ij}^{6}} \right) + \lambda \cdot \sum_{i=1}^{N} \sum_{d=4}^{D} p_{id}^{2}
\]

where \( N \) is the number of particles and \( D > 3 \) the (extended) number of dimensions. \( \lambda \) is the penalization factor for the added dimensions, where \( p_{id} \) is the \( d^{th} \) dimension component of the \( i^{th} \) particle. The pairwise distances \( r_{ij} \) are calculated in all \( D \) dimensions.

In this case, when \( \lambda = 0 \), the minimum is in \( D \) dimensions and when \( \lambda = \infty \) the minimum is in 3D (all \( p_{id} = 0 \)). If the problem is solved in \( D \) dimensions, the 3D solution can be found by “squeezing-out” the extra dimensions by successive iterations of increasing \( \lambda \) and minimizing over the particle positions until the extra dimensions are 0.

Another way of passing from one dimensionality to another is the use of a Homotopy

\[
E_{\text{tot}} = \lambda \cdot \sum_{i}^{N-1} \sum_{j=i+1}^{N} \left( \frac{1}{\bar{r}_{ij}^{12}} - \frac{1}{\bar{r}_{ij}^{6}} \right) + (1 - \lambda) \cdot \sum_{i}^{N-1} \sum_{j=i+1}^{N} \left( \frac{1}{r_{ij}^{12}} - \frac{1}{r_{ij}^{6}} \right)
\]

where \( \bar{r}_{ij} \) is the pairwise distance in three dimensions and \( r_{ij} \) the distance in \( D \) dimensions. The parameter \( \lambda \) is in the range 0...1. At \( \lambda = 0 \) we have the total energy in \( D \) dimensions and at \( \lambda = 1 \) in three dimensions. As with the penalization strategy, we first solve the problem in \( D \) dimensions and then iteratively increase \( \lambda \) step-wise and minimizing over the particle positions.
The energy minimization of the clusters themselves is done using the Polak-Ribiere variant (Polak & Ribiere 1969) of the Conjugate Gradients algorithm (Concus, Golub & O’Leary 1978). The Newton-Raphson algorithm was also implemented (this requires fixing all \( p_{d} = 0 \) where \( d \leq i \)), but failed to converge faster than Conjugate Gradients (calculating the Hessian matrix of the potential energy function and decomposing it resulted being far more expensive than line-searches in high dimensions).

## 4.3 Adding Dimensions

An iterative scheme for finding the global minimum of a Lennard-Jones cluster using additional dimensions can be formulated as follows:

1. Generate a random configuration and minimize it in 3D.
2. “Relax” the cluster into \( D \) dimensions.
3. “Squeeze” the cluster back to \( D \) dimensions.
4. If the total potential energy decreased, go to step 2.
5. If the total potential energy increased, go to step 1.

The initial random configuration is first minimized with some random-walk steps before being passed to the Conjugate Gradients solver. This is done to avoid particles being ejected from the cluster in the first few Conjugate Gradient steps due to their gradient being too large.

The “relaxation” in \( D \) dimensions is done using the Homotopy described above. The parameter \( \lambda \) is set \( = 1 \) and at every step until \( 2^{-1022} \) (minimum double-precision value) is halved and the system re-minimized using the Conjugate Gradients method (Concus et al. 1978). A final minimization step is performed with \( \lambda = 0 \).

Squeezing is also done using the Homotopy. The parameter \( \lambda \) is initialized at \( 2^{-1022} \). At every step until \( \lambda \geq 1 \), \( \lambda \) is increased by

\[
\min \left[ \lambda \cdot 0.01, \frac{-0.001 \cdot E_{tot}}{\frac{\partial E_{tot}}{\partial \lambda}} \right]
\]

and the system is minimized using Conjugate Gradients. The first term in the minimum is a multiplicative incremental step and the second term is to bound the increase of the total energy by a factor of 0.001. In a final step, the energy is simply minimized in three dimensions.

By stretching the system back and forth between three and four dimensions, we can circumvent many – if not all – energy barriers. This is similar to the approach used to solve the general N-Body problem in Darwin (Gonnet, Hallett, Korostensky & Bernardin 2000).
4.4 Maximum Dimensions

If we assume that, given an optimal configuration in $D$ dimensions, squeezing it down will result in an optimal configuration in $D' < D$ dimensions, we can reduce the problem of finding the optimal configuration in three dimensions to the problem of finding the optimal configuration in any number of dimensions $D > 3$.

This may seem a somewhat counterproductive approach – finding the global minimum in a problem with a higher number of degrees of freedom – until we consider that at $D = N - 1$ there exists only one – trivial – absolute minimum of the potential energy surface. This is the configuration where each particle interacts with every other particle at it’s optimal radius $2^{1/6} \sigma$ resulting in the maximum potential energy

$$
\epsilon \cdot \frac{N \cdot (N - 1)}{2}
$$

These configurations are, for example, two particles on a line in one dimension, an equidistant triangle of three particles in two dimensions and a pyramid of four particles in three dimensions and so on (Figure 4.3).

Therefore, if an optimal configuration in $D$ dimensions can be squeezed to the optimal configuration in $D' < D$ dimensions – which is not proven herein or elsewhere, but used as a working hypothesis –, then squeezing down from $D = N - 1$ dimensions to three dimensions should lead to the optimal configuration in three dimensions.

The squeezing is done in the same way as in the 3D-2D Iteration with a Homotopy.

4.5 Results

Three methods were applied to Lennard-Jones clusters of size 20 to 40: 3D-4D Iteration, 3D-5D Iteration and Maximum Dimensions. The results are shown in Table 4.1.

For the results of the iterative methods – which rely on random start positions – it would have been much more meaningful to present the number of times the iteration finds the global minimum on average. The time constraints on this Diploma Thesis, however, did not allow this.
<table>
<thead>
<tr>
<th>Size</th>
<th>3D-4D Iteration</th>
<th>3D-5D Iteration</th>
<th>Max. Dimensions</th>
<th>( E_{\text{tot}} )</th>
<th>( E_{\text{tot}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>1 1109s</td>
<td>1 445s</td>
<td>(-7.177 \times 10^{-9})</td>
<td>4399s</td>
<td>(-7.177 \times 10^{-9})</td>
</tr>
<tr>
<td>21</td>
<td>22 2126s</td>
<td>14 5707s</td>
<td>(-8.1651 \times 10^{-10})</td>
<td>1591s</td>
<td>(-8.1684 \times 10^{-10})</td>
</tr>
<tr>
<td>22</td>
<td>5 5405s</td>
<td>1 434s</td>
<td>(-8.5422 \times 10^{-10})</td>
<td>2467s</td>
<td>(-8.6809 \times 10^{-10})</td>
</tr>
<tr>
<td>23</td>
<td>5 7721s</td>
<td>1 531s</td>
<td>(-9.2844 \times 10^{-9})</td>
<td>2064s</td>
<td>(-9.2844 \times 10^{-9})</td>
</tr>
<tr>
<td>24</td>
<td>1 1280s</td>
<td>5 2831s</td>
<td>(-9.7313 \times 10^{-9})</td>
<td>9121s</td>
<td>(-9.7348 \times 10^{-9})</td>
</tr>
<tr>
<td>25</td>
<td>2 2880s</td>
<td>8 5254s</td>
<td>(-1.0237 \times 10^{-2})</td>
<td>3200s</td>
<td>(-1.0237 \times 10^{-2})</td>
</tr>
<tr>
<td>26</td>
<td>2 4174s</td>
<td>3 2252s</td>
<td>(-1.0831 \times 10^{-2})</td>
<td>2827s</td>
<td>(-1.0831 \times 10^{-2})</td>
</tr>
<tr>
<td>27</td>
<td>1 2430s</td>
<td>1 708s</td>
<td>(-1.1116 \times 10^{-2})</td>
<td>5113s</td>
<td>(-1.1287 \times 10^{-2})</td>
</tr>
<tr>
<td>28</td>
<td>6 17872s</td>
<td>1 1080s</td>
<td>(-1.1782 \times 10^{-2})</td>
<td>4850s</td>
<td>(-1.1782 \times 10^{-2})</td>
</tr>
<tr>
<td>29</td>
<td>1 466s</td>
<td>2 2057s</td>
<td>(-1.2358 \times 10^{-2})</td>
<td>5235s</td>
<td>(-1.2358 \times 10^{-2})</td>
</tr>
<tr>
<td>30</td>
<td>40 29938s</td>
<td>22 24131s</td>
<td>(-1.2588 \times 10^{-2})</td>
<td>7884s</td>
<td>(-1.2828 \times 10^{-2})</td>
</tr>
<tr>
<td>31</td>
<td>– 93932s</td>
<td>s (-1.3207 \times 10^{-2})</td>
<td>9035s</td>
<td>(-1.3358 \times 10^{-2})</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>– 97851s</td>
<td>s (-1.3815 \times 10^{-1})</td>
<td>28814s</td>
<td>(-1.3963 \times 10^{-1})</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>– 105056s</td>
<td>s (-1.4362 \times 10^{-2})</td>
<td>9551s</td>
<td>(-1.4482 \times 10^{-2})</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>– 110418s</td>
<td>s (-1.4672 \times 10^{-2})</td>
<td>12132s</td>
<td>(-1.5004 \times 10^{-2})</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>92 119531s</td>
<td>s (-1.5452 \times 10^{-2})</td>
<td>11385s</td>
<td>(-1.5575 \times 10^{-2})</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>– 135984s</td>
<td>s (-1.5927 \times 10^{-2})</td>
<td>40181s</td>
<td>(-1.6182 \times 10^{-2})</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>– 140538s</td>
<td>s (-1.6663 \times 10^{-2})</td>
<td>15569s</td>
<td>(-1.6703 \times 10^{-2})</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>46 73779s</td>
<td>s (-1.6997 \times 10^{-2})</td>
<td>36546s</td>
<td>(-1.7392 \times 10^{-2})</td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>s (-1.7502 \times 10^{-2})</td>
<td></td>
<td></td>
<td>(-1.8003 \times 10^{-2})</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>s (-1.8083 \times 10^{-2})</td>
<td></td>
<td></td>
<td>(-1.8524 \times 10^{-2})</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.1: Scores for different clusters using both the 3D-4D iteration and maximum dimensions methods. A “–” means that the minimum was not found in less than 100 iterations. The left most column is the minimum recorded in the Cambridge Cluster Database. All times were measured on an AMD Athlon™ MP running at 1800GHz.
4.6 Discussion

The results of both iterative runs are pretty impressive, considering the number of iterations used, as opposed to the wall-clock time needed to find the minimum. Although at a first glance they may seem to compare badly with the results presented by Locatelli & Schoen (2002a), one must consider that their “runs” include a random-walk search that takes up to 300 steps before aborting, whereas the iteration presented here makes no attempt to elaborate on the minima found at the end of each run.

The results for the deterministic Maximum Dimensions method are – although not all minima were found – quite encouraging, especially if we consider the probability of hitting such potential energies by chance (Locatelli & Schoen 2002b).

The major drawback of these methods is the amount of CPU time needed for the Homotopy, which involves thousands of Conjugate Gradient runs. Added to this comes the horrendous scaling factor of the Maximum Dimensions approach (number of unknowns for Conjugate Gradients grows with $O(N^2)$).

This lack of efficiency is due, in great part, to the regime used for adjusting the parameter $\lambda$ of the Homotopy. Choosing too wide a step increases performance dramatically, yet reduces the probability of getting a global minimum significantly. Halving the step-size for the 21-particle cluster, for instance, finds the global minimum of $-8.1684 \cdot 10^{-10}$, yet it roughly doubles the amount of time needed to find it. More work would be needed to find a more efficient way of moving through dimensionalities to make these methods practicable for larger clusters.
Chapter 5

Conclusions and Outlook

In this chapter, we review what progress has been made over previous work on Lennard-Jones cluster minimization. Open problems remaining after – or created by – this work are discussed.

5.1 Improvement on Previous Work

Although none of the methods presented herein managed to find previously unknown global minima, it still represents a contribution in terms of the models tested to solve the problem.

Although the Gaussian Process interpolation yielded no results in even a basic problem (three particles in two dimensions), it further pushes the point that interpolation schemes are not especially suited for Lennard-Jones cluster minimization. This point was shown already in Foreman et al. (1999), citing the large number of symmetries and the non-funnel layout of the local minima on the energy landscape as the main problems.

The SOM coarse-graining, although not yielding any useful results directly, provides a rather simple way of generating good initial conditions for larger Lennard-Jones clusters, although the main problem of re-constructing the microscopic space after updating the coarse-grained space still remains. It is, however, the first attempt at applying mesoscopic or macroscopic methods to Lennard-Jones clusters.

Dimensionality Up-Scaling fits in nicely within the methods aimed directly at reducing or eliminating energy barriers (Wales & Doyle 1997, Laio & Parrinello 2002), in introducing a new way of getting around them which has not yet been fully explored – probably due to it’s counter-intuitive approach of increasing the complexity of the problem to solve it. Although there are many issues to be solved concerning the efficiency and precision of the Homotopy, Dimensionality Up-Scaling represents a genuine contribution to Lennard-Jones cluster minimization (although this Diplomarbeit is not the first to use this approach in molecular dynamics problems (Gonnet 2002)).
All in all it should be mentioned that although most of the results are negative, they are results non-withstanding. When trying to solve such complex problems, it is usually just as important to know what won’t work as what might work.

5.2 Open Problems

Although his Diplomarbeit presents some interesting methods for Lennard-Jones cluster minimization, it probably creates more questions than it answers. A few of the more pertinent ones are:

- **Coarse-Graining Update:** What different methods can we use to update the coarse-grained particles? Can we empirically or analytically derive pairwise or many-body energy potentials over which we could minimize the coarse-grained system?

- **Coarse-Graining Reconstruction:** Is there a better way of re-construction the microscopic system than randomly distributing the particles?

- **Dimensionality Up-Scaling Homotopy:** Are there other ways, other than Homotopy or penalization, to move efficiently from one dimensionality to another? Is there a more efficient or precise strategy for adjusting the \( \lambda \) parameter of the Homotopy?

- **Dimensionality Up-Scaling Proof:** Can we prove that, given a global minimum configuration in \( D \) dimensions, we can reach the global minimum in three dimensions through a Homotopy or some other method?
Bibliography


