BULGARIAN ACADEMY OF SCIENCES
INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGIES
DEPARTMENT OF INFORMATION PROCESSES AND DECISION SUPPORT SYSTEMS

Rossen Mikhov Mikhov

MONTE CARLO APPROACH FOR OPTIMIZATION OF BIMETALLIC NANOSTRUCTURES

ABSTRACT

of Dissertation

for acquiring the educational and scientific degree "Doctor"

Professional field: 4.6. "Informatics and Computer Science"

Doctoral program "Informatics"

Scientific supervisor

Prof. Dr. Leoneed Kirilov

Sofia, 2025

The dissertation was discussed and admitted to defense at an extended meeting of the Department of Information Processes and Decision Support Systems at the IICT–BAS, held on 30.09.2025.

The dissertation is structured as follows: introduction, 6 chapters, conclusion, contributions, list of publications, list of discovered citations, declaration of originality of the results and bibliography. The dissertation has a total of 124 pages, 34 figures, 7 tables and 149 references.

The defense of the dissertation will take place on ..... from ..... in hall ..... of bl. 2 of IICT–BAS at an open meeting of the scientific jury composed of:

Scientific jury:

1. .....
2. .....
3. .....
4. .....
5. .....

The reviews and opinions of the members of the scientific jury and the abstract are published on the website of IICT–BAS.

The materials for the defense are available to those interested in room ..... of IICT–BAS, bl. 2, Acad. G. Bonchev Str., Sofia.

Author: **Rossen Mikhov Mikhov**

Title: **Monte Carlo approach for optimization of bimetallic nanostructures**

INTRODUCTION

Nanoparticles with a size between a few hundred and a few thousand atoms are among the most difficult to numerically model atomic systems. On the one hand, the nanoparticle is too small: established methods for modeling macroscopic metals are not applicable, because in a nanoparticle there is a very strong surface effect, and its properties strongly depend on its size and geometric shape. But on the other hand, the nanoparticle is too large: methods that are directly based on quantum physics are not applicable, because with such a large number of atoms, the calculations become unbearably complex even for the most powerful supercomputers. For this reason, the numerical modeling of nanoparticles remains an unsolved problem.

Various approaches exist for the acceptable modeling of nanoparticles, many of which are based on semi-empirical approximations. A lot of work is being done on this issue, but all available approaches are incomplete and have significant shortcomings. The present dissertation is not spared from these problems. It is simply part of the gradual improvement of one of the many existing approaches.

The main tool employed in this dissertation is Monte Carlo simulation, with the aim of finding stable atomic configurations. This is a rather natural approach, because in a certain abstract sense it is also the way in which nature finds stable configurations. Atoms move, try out a large number of configurations, and eventually the particle "freezes" in a shape in which it has low potential energy. We hope that the proposed approach represents a good balance and makes it possible to experiment widely, to investigate nanoparticles and to study some of their properties in a relatively short time, using computers with relatively modest capabilities.

The dissertation is structured into 6 chapters, as follows.

Chapter 1 provides an overview of existing methods for modeling and numerical optimization of the atomic configurations of metallic and bimetallic nanostructures.

In Chapter 2, a two-stage Monte Carlo approach for optimization of bimetallic nanostructures is proposed, including a mathematical model, three constituent algorithms, and some essential features of their software implementation. A two-stage method with five steps is defined, and the significance of the steps is described.

In Chapter 3, the proposed two-stage method is numerically tested. It is compared to the previous one-stage approach, verifying it achieves superior results. Next, the optimal proportion for distributing computational resources between the

two stages is investigated, and the chosen approach for tuning the parameters of the method is experimentally justified.

In Chapter 4, the influence of the initial temperature—a key parameter— on the performance of the wide-lattice Monte Carlo algorithm is investigated. A large number of tests are performed for nanoparticles of two chemical elements (silver and cobalt) with different sizes, on lattices of different geometries and sizes.

In Chapter 5, the two-stage method is applied to the study of the atomic ordering and of surface segregation processes in gold–silver nanocages with 3000 atoms. Some aspects of the method are adapted to the specifics of working with nanocages. A comparative analysis of the results for three AuAg ratios and two lattices with different symmetries shows how the interaction of these factors determines the local order and affects the macroscopic properties of the bimetallic nanocages.

Chapter 6 describes the software system developed as part of the work on the dissertation. A software architecture is proposed for the implementation of the two-stage method, which allows a high degree of optimizability for performance of the computations and flexibility for concurrent execution and combination of the constituent algorithms in different conditions. The developed software works on Linux and Windows operating systems and uses the standard XYZ format for input and output data for atomic configurations.

CHAPTER 1. ANALYSIS OF METHODS FOR MODELING AND NUMERICAL OPTIMIZATION OF ATOMIC CONFIGURATIONS OF METALLIC AND BIMETALLIC NANOSTRUCTURES

The important theoretical and practical significance of studying the structural characteristics and transformations of metal nanoparticles and nanoscale heterostructures is associated with the broad prospects for their application in various fields. For example, they can serve as nanoelectrodes/nanoconductors, as sensors or as catalysts. New possible applications are constantly being proposed and tested. The success of these technologies and the speed of their development crucially depends on a good understanding of the processes at the nanoscale, the behavior of metal atoms in structures of such sizes and their specific properties and effects, which sometimes greatly differ from the behavior of the same metals in classical macro-configurations.

1.1. The stability of metal nanostructures as an optimization problem

The first important step when studying metal nanoclusters is to identify their stable geometric structures. One of the most commonly used approaches in the numerical search for stable nanostructures is to define the potential energy of the system in some way as a function of the interatomic distances, and then use numerical algorithms to find a global minimum of this function. The problem here is that no matter how it is defined, the potential energy is inevitably a function with a huge number of local minima, which makes finding a global minimum extremely difficult.

The appropriate potential function plays an important role for the computer simulation of metallic structures. A large number of studies have been devoted to the problem of defining metallic potentials [Cleri and Rosato, 1993; Guevara et al., 1995]. Most often, the analytical form of the function is constructed in such a way that it contains a certain set of independent parameters. Using these parameters, the potentials are fitted to experimental data or to data from *ab initio* calculations.

A relatively simple way to express the atomic and electronic structure is the Gupta potential [Gupta, 1981] and its variants—the so-called tight binding (TB) potential [Cleri and Rosato, 1993]. Despite its simple analytical form, the TB model describes elastic properties, defect characteristics and melting properties quite well for a wide range of metals.

1.2. Methods for optimization of metallic nanostructures

There are many optimization approaches that aim to solve the problem of finding a global minimum of the potential energy and to explore stable atomic configurations. It is worth mentioning the molecular dynamics (MD) method [Liu et al., 2017], basin-hopping [Wales and Doye, 1997], fast annealing evolutionary algorithms (FAEA) [Cai and Shao, 2002], random tunneling algorithms (RTA) [Jiang et al., 2002], genetic algorithms (GA) [Gregurick et al., 1996], simulated annealing (SA) algorithms [Ma and Straub, 1994]. There are also promising algorithms based on machine learning [Chen et al., 2020].

1.3. Methods for optimization of bimetallic nanostructures

Due to the specific difficulty of the problem in bimetallic nanostructures, targeted algorithms are often used for their optimization. Here again, methods based on basin hopping [Rossi and Ferrando, 2017], genetic algorithms [Paz-Borbón et al.,

2007], as well as other types of Monte Carlo simulations [Giménez and Schmicker, 2011; Shao et al., 2017] are widely used, and in some more limited cases, density functional theory (DFT) calculations can also be performed [Kovács et al., 2017].

1.4. Modeling of diffusion in bimetallic nanostructures

In the contact region of two dissimilar metals, a process of mutual diffusion occurs and a transition region is formed, the so-called diffusion zone [Li et al., 2019]. As a result, there is a diffusional movement of atoms, which ultimately leads to the establishment of a phase composition in the entire volume of the sample, which is determined by a phase diagram. However, in nanoscale systems, it is of particular importance to take into account the effects of size, specifically at the interface between the components.

Surface diffusion plays a crucial role in determining the shape and morphology of growing nanoparticles and nanofilms. Understanding the mechanism of this process helps to achieve desired properties of the materials and avoid undesirable ones.

1.5. Nanocages

Nanocages are nanoparticles that have a cavity in the middle. They are one of the relatively new objects of study, possessing unique properties. Due to their specific structure, the problem of thermal stability and the change in the properties of their local structure are among the central considerations in the selection of temperature conditions and particle sizes when using nanocages in medicine, catalysis and other areas. Moreover, thermoinduced action on mono- and bimetallic nanocages leads to differences in the identifiable temperature regions of healing of the cavities (pores) on the surfaces and in the inner region (core) of the nanocage, as well as to the structural collapse of the nanocage. For bimetallic nanocages, of interest is also the study the patterns of surface segregation before and after the collapse.

1.6. Conclusion

The geometric shape and structure of atomic clusters of metal nanoparticles of a given size are crucial for determining their properties, and in bimetallic nanostructures the atomic ordering is of utmost importance. Finding stable atomic configurations is a basic research problem that can be formulated as a computational problem for global optimization. A configuration is stable when its potential energy is minimal.

After analyzing the various methods and approaches described in the literature, it can be concluded that the TB potential of Gupta gives good results with high flexibility, which makes it popular. Its parameters will likely continue to be updated and refined in the future with the advancement of the experimental base and the possibilities for *ab initio* calculations. For this reason, the present dissertation focuses on working with this potential.

Finding a global minimum is a difficult problem, especially for bimetallic nanoalloys, therefore it is necessary to search for approximate solutions using effective optimization strategies (metaheuristics). There are many approaches for this, a significant number of which based on Monte Carlo simulations. Due to the large variety of possible nanostructures, of great importance is the search for new ways for improving the results of such Monte Carlo simulations and for facilitating their use under various conditions.

1.7. Purpose and objectives of the dissertation

Based on the analysis of existing methods for modeling and numerical optimization of atomic configurations of metallic and bimetallic nanostructures, the purpose of the present dissertation is formulated as follows: to develop a Monte Carlo approach with simulated annealing (SA), using the tight-binding (TB) potential, for the optimization of different types of bimetallic nanostructures, including nanoparticles, nanowires and nanofilms. To achieve this goal, the following tasks need to be completed:

- to propose a method for optimization of bimetallic nanostructures, including nanoparticles, nanowires and nanofilms;
- to investigate the effectiveness of the proposed method;
- to propose an appropriate approach for choosing and adjusting the parameters of the method;
- to determine which of the following factors most significantly affect the optimal choice of initial temperature for simulated annealing: chemical element, nanoparticle size, lattice type, lattice size;
- to propose a software architecture and to develop a software system implementing the new method that allows a high degree of optimizability for performance of the computations, flexibility for varying the algorithms and their parameters and good compatibility with external applications for analysis and visualization of the results;

- to conduct an investigation applying the proposed method to the study of a specific class of gold–silver nanocages with 3000 atoms, which are of interest for many applications, in order to establish how the differences in the Au:Ag ratio and the symmetry of the lattice affect the atomic ordering and the processes of surface segregation.

CHAPTER 2. TWO-STAGE MONTE CARLO APPROACH FOR OPTIMIZATION OF BIMETALLIC NANOSTRUCTURES

In the present dissertation, a two-stage lattice Monte Carlo approach is proposed for the optimization of bimetallic nanoalloys [Mikhov et al., 2020; Mikhov et al., 2022]. The two stages consist of simulated annealing on a larger lattice, followed by simulated diffusion.

2.1. Mathematical model of the bimetallic nanostructure

The bimetallic nanoalloy is modeled as a collection of a certain number of atoms of two metal elements. The atoms are arranged on a lattice consisting of empty nodes. The first goal is to find a configuration of the atoms on this lattice so that the resulting nanoparticle is stable. i.e., to minimize the potential energy of the system.

To calculate the interaction between atoms, the many-body strong-binding (TB) potential of Cleri—Rosato [Cleri and Rosato, 1993] was used. The total potential energy of the system has the following form:

$$E = \sum_i \sum_{j \neq i} E_{ij,ab} - \sum_i \sqrt{\sum_{j \neq i} B_{ij,ab}} \tag{2.1}$$

$$E_{ij,ab} = A_{ab} \exp\left(-p_{ab} \left(\frac{r_{ij}}{r_{0,ab}} - 1 \right) \right) \tag{2.2}$$

$$B_{ij,ab} = \xi_{ab}^2 \exp\left(-2\, q_{ab} \left(\frac{r_{ij}}{r_{0,ab}} - 1 \right) \right), \tag{2.3}$$

where i lists all atoms in the system; j lists all atoms other than i but at a distance from atom i not greater than the predefined global parameter R_{cut}; a and b denote the chemical element of atoms i and j; $E_{ij,ab}$ and $B_{ij,ab}$ are, respectively, the repulsive and attractive components of the potential energy generated by the atomic pair (i, j); r_{ij} is the distance between the atoms; $r_{0,ab}$, A_{ab}, p_{ab}, ξ_{ab} and q_{ab} are constants specific to the chemical elements considered. The value of R_{cut} used here corresponds to five coordination sphere distances, beyond which we assume that the interaction between the atoms becomes zero.

The modeled nanoalloy can have nano-dimensions along some of the coordinate axes and macro-dimensions along the remaining axes. A distinction is made between 3 cases (illustrated in Fig. 2.1): a nanoparticle, where all three axes are nano-dimensions; a nanowire, where one of the axes is macro-dimensions; a

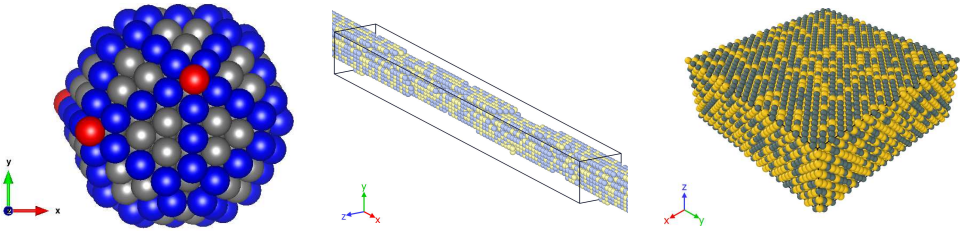


Fig. 2.1. Model of a nanoparticle, nanowire and nanofilm. The configurations are obtained with the algorithms of 2.2.

nanofilm, where two of the axes are macro-dimensions.

When it comes to a nanowire or nanofilm, the model is analogous to that for a nanoparticle, but periodic boundary conditions are used. This means that a nano-sized lattice is used along all three axes, but a periodic window is introduced along the axes where macro-sizes are modeled. The distance between atoms is defined as follows:

$$r_{ij} = \sqrt{|\Delta x_{ij}|^2 + |\Delta y_{ij}|^2 + |\Delta z_{ij}|^2} \tag{2.4}$$

$$|\Delta x_{ij}| = \min\{|x_i - x_j|, L_x - |x_i - x_j|\}, \tag{2.5}$$

where x_i and x_j are the x-coordinates of atoms i and j inside the periodic cell, and L_x is the x-size of the periodic cell. If macro-sizes along the y and z axes are modeled, then $|\Delta y_{ij}|$ and $|\Delta z_{ij}|$ are also calculated in a similar way. The non-periodic case corresponds to $L_x = \infty$.

The current model, as well as its algorithms, are suitable for optimization of bimetallic nanoparticles with sizes between 100 and several tens of thousands of atoms.

2.2. Constituent algorithms

2.2.1. The wide-lattice Monte Carlo algorithm

The first constituent Monte Carlo algorithm, which we will call the "wide-lattice" algorithm, is based on simulated annealing and was developed in a series of papers by a team with the author's participation [Myasnichenko et al., 2019].

The algorithm starts by generating a lattice with a specific geometric structure, such as face-centered cubic (fcc), icosahedral, decahedral, etc. Each lattice consists of a certain number of nodes in space, given by their Cartesian coordinates (x, y, z). For the purposes of this algorithm, a wide lattice is used, which contains several times more nodes than the total number of atoms in the system.

Initially, atoms are placed at randomly selected nodes. Each node is defined as either empty or containing an atom of one of the two chemical elements. Immediately after the atoms are placed, a series of precomputations are performed and stored in memory in order to remove as many computational operations as possible from the main cycle of the algorithm. This step, described in 2.2.2, is crucial for the speed of the algorithm.

The basic loop consists of a series of iterations. In each iteration, one atom and one adjacent empty node are randomly selected and it is decided whether the atom will jump to the empty node. If the potential energy of the system would decrease, the jump is made unconditionally. Otherwise, the jump can still be made, but with a probability determined by the following formula:

$$P = \exp\left(-\frac{\Delta E}{kT}\right), \tag{2.6}$$

where ΔE is the difference in the energies of the two configurations; T is the current temperature of the system; k is the Boltzmann constant. In the case where there are no empty nodes in the vicinity of the selected atom, the jump is not performed.

Each iteration ends either with a jump of one atom or without a jump.

The temperature is set high at the beginning and then gradually decreased. A linear cooling formula is used:

$$T = \max\{1, T_0 + s\,\Delta T\}, \tag{2.7}$$

where T is the current temperature; T_0 is the initial temperature; s is the iteration number; ΔT is a negative constant. This temperature change occurs once every few thousand iterations.

The algorithm ends when the system has reached equilibrium, the current temperature has reached 1 K, or for a certain (pre-set) number of iterations the energy has not decreased.

2.2.2. Implementation of the algorithm, guaranteeing its speed

In a simulation of this type, the ability to work with a huge number of iterations is an important condition for achieving better results. This algorithm is purposefully designed so that the main cycle is as simple as possible. The speed of the algorithm

is guaranteed by the use of appropriate data structures and a series of preliminary calculations.

The nodes are stored in two arrays `N` and `A`, where `N` gives the index of a given node in `A`, and `A` gives the index of a given node in `N`. The arrays are sorted in a way that allows for constant time to search for information about nodes and atoms, to select atoms randomly, to add, remove, and move atoms.

Before the main loop begins, lists of its neighbors and neighborhoods are computed and stored for each node. In addition, for each node i, each node j (within a distance $\leq R_{\text{cut}}$) and each possible combination of chemical elements a and b, $E_{ij,ab}$ and $B_{ij,ab}$ are calculated using (2.2)–(2.3) and stored. From this point on, the algorithm forgets about the Cartesian coordinates of the nodes and starts working only with the node indices, together with the values of the energy components calculated here.

For each atom i the sum under the radical in (2.1) is calculated and stored. This value is updated in due time during the main cycle of the algorithm when atoms are moved. This allows to avoid enumerating the R_{cut}-neighborhoods of R_{cut}-neighborhoods of nodes during the main cycle, which would otherwise be necessary and would increase the number of nodes enumerated per square.

As a result, the algorithm can run billions of iterations in minutes on a standard personal computer. The algorithm is not parallelized, since it is a Monte Carlo simulation using random numbers and it is assumed that in order to find stable results, it will be run multiple times under the same initial conditions. These runs can run in parallel.

2.2.3. The diffusion Monte Carlo algorithm

The second constituent Monte Carlo algorithm, which we will call the diffusion algorithm, is described in [Myasnichenko et al., 2022] by a team including the author. The main purpose of this algorithm is simulation of diffusion processes in bimetallic nanostructures, but in the present dissertation work we use it as the second stage of a combined approach for optimization of bimetallic nanostructures.

The algorithm starts with a lattice pre-filled with atoms of the two different chemical elements, plus **a small number** of empty nodes ($\sim$4 for a structure of 200 atoms). Preliminary calculations are performed, analogous to those in 2.2.2, in order to significantly speed up the algorithm. The main cycle is again a series of iterations, but here at each iteration an empty node is chosen at random and

the iteration always ends with a jump of a neighboring atom at this empty node. Which atom will jump is again determined by (2.6), but here this value is calculated separately for each candidate atom and is used as a relative probability in choosing one of these candidates.

Another important difference is in the definition of the neighborhood. In the wide-lattice algorithm, the neighborhood is understood as the radius of one coordination sphere, while here the neighborhood is taken as an atom within a radius of three coordination spheres. This difference is due to the small amount of empty nodes in the diffusion process.

The temperature is controlled analogously to the wide-lattice Monte Carlo algorithm, according to a linear formula (2.7).

2.2.4. Relaxation with molecular dynamics

It is existing practice [Myasnichenko et al., 2020] to perform another relaxation step using molecular dynamics (MD) after the Monte Carlo simulation is complete to further improve the energy of the system. In the MD simulation, the time trajectories of all atoms in the system are followed in space by integrating the Newtonian equations of motion. The forces are extracted as the negative gradient of the interatomic potential.

For the purposes of this dissertation, the so-called modified Verlet velocity algorithm was chosen, where the positions, velocities and accelerations at iteration $t+\Delta t$ are calculated as follows:

$$\vec{r}\,(t+\Delta t) = \vec{r}\,(t) + \vec{v}\,(t)\,\Delta t + \frac{1}{2}\,\vec{a}\,(t)\,(\Delta t)^2 \tag{2.8}$$

$$\vec{v}\,(t+\frac{1}{2}\,\Delta t) = \vec{v}\,(t) + \frac{1}{2}\,\vec{a}\,(t)\,\Delta t \tag{2.9}$$

$$\vec{a}\,(t) = -\frac{\operatorname{grad} U(\vec{r}\,(t))}{m}, \tag{2.10}$$

where $U(\vec{r}\,(t))$ is the potential energy and m is the mass. An advantage of this method is that it is self-starting, i.e. it does not require special treatment of the numerical integration scheme around $t=0$.

2.3. The two-stage method, combining the constituent algorithms

The starting point of this research is the observation that combining the two Monte Carlo algorithms above can lead to better results than following a single-stage approach.

In this regard, a combined method is proposed, which includes the following steps:

STAGE I

Step 1. Parameter tuning.

The wide-lattice Monte Carlo algorithm is run repeatedly from random initial configurations, for N_1 ($\sim 40\times10^6$) iterations in each trial, to determine the optimal initial temperature T_2 and scaling factors (along each of the x, y, and z axes) for the lattice. Because of the smaller number of iterations, each trial completes quickly, saving time for the more important subsequent steps.

Step 2. Shape fixing.

The wide-lattice Monte Carlo algorithm is run again, repeatedly from random initial configurations, but this time for N_2 ($\sim 1\times10^9$) iterations in each trial, using the best values from **Step 1** for the initial temperature T_2 and for the scaling factors. The cooling here is much slower than in **Step 1**: since a linear cooling schedule is used, the cooling rate is (in short) the initial temperature divided by the number of iterations.

STAGE II

Step 3. Parameter tuning for diffusion.

The diffusion Monte Carlo algorithm is run repeatedly on the results of **Step 2**, for N_3 ($\sim 10\times10^6$) iterations in each trial, to determine the optimal initial temperature T_4 for diffusion. Because of the smaller number of iterations, each trial completes quickly, saving time for the more important subsequent steps.

Step 4. Diffusion.

The diffusion Monte Carlo algorithm is run again, once on each resulting configuration of **Step 2**, for N_4 ($\sim 100\times10^6$) iterations in each trial, using the best value from **Step 3** for the initial temperature T_4.

Step 5. Relaxation.

The molecular dynamics algorithm is run once on each resulting configuration of **Step 4**, to further improve the energy of the system. The MD simulations are carried out in the canonical ensemble at 0.1 K, with a time step of 0.5 fs. As the final result, the configuration with the lowest energy among those observed is selected.

For **Step 2**, the number of atoms initially placed is β (~ 4) more than the desired number of atoms in the final result (these β atoms are distributed proportionally to each chemical element). After **Step 2** is completed, all empty nodes are deleted, then β atoms are converted into empty nodes, which will be used

as vacancies during diffusion (these β atoms are chosen randomly, but in such a way as to preserve the proportion between the chemical elements).

Since the proposed method is not deterministic, each step involves running several parallel trials to ensure stable behavior. Ultimately, one configuration is selected, but more than one can be visualized and analyzed, depending on what aspects of the configurations are of interest for the purposes of a given study.

2.3.1. Purpose of the steps of the proposed method

The main work of the method is concentrated in **Step 2** and **Step 4**.

The goal of **Step 2** is to obtain energetically favorable geometric shapes suitable for the nanoparticle, nanofilm or nanowire of interest on the given lattice. Therefore, a wide lattice is used and the atoms are allowed to move freely along it. The configurations obtained from **Step 2** also have to some extent a low-energy atomic ordering of the two metal elements, which is the main step of the previous one-stage approach, which the present dissertation aims to improve by adding **Step 4**.

In **Step 4**, the geometric shape is fixed and does not change any more. The movement here is carried out exclusively by diffusion, with the number of empty nodes β is very small ($\sim$4). The goal is to achieve an energetically favorable ordering of the atoms of the two chemical elements within the already fixed geometric shape. The influence of adding **Step 4** on the quality of the results is studied in detail in Chapter 3, from which it is clearly seen that distributing the computational resources in a certain proportion between **Step 2** and **Step 4** leads to significantly better results than using the same amount of resources only on **Step 2**.

Step 1 and **Step 3** aim to determine the appropriate starting temperature for the respective algorithm used under the specific conditions of the nanoparticle under study. Since the cooling rate is (approximately) the starting temperature divided by the number of iterations, if we use the number of iterations as a measure of the amount of computational resources used, then the starting temperature is the main parameter regulating how efficiently these resources are used. The results in Chapter 3 and Chapter 4 show that determining this parameter in **Step 1** and **Step 3** is useful and important.

The purpose of **Step 5** is to achieve relaxation of the atoms in the system. The lattice is removed and the atoms, which until now have been confined in a lattice, are freed to roam in their surroundings according to a numerical integration of

Newton's equations of motion, in order to fall to the bottom of their local minimum. A very low temperature is used so as not to disturb the already achieved atomic configuration, which is globally energetically favorable. **Step 5** is considered an additional step in this method, because the atomic configuration, which is decisive for the properties of the particle, practically does not change. The only thing that matters is that the numerical value of the potential energy is changed, which allows for correct comparison of results between lattices with different structures or results achieved by other methods.

2.4. Conclusion

This chapter describes the chosen mathematical model, the used constituent algorithms and the proposed two-stage method for searching for stable configurations of bimetallic nanostructures. The description gives exemplary values of some of the parameters, except for the initial temperature. The following chapters investigate the behavior of this method under different conditions, which justifies the specific formulation in 5 steps, the appropriate distribution of computational resources between the steps, as well as the influence of various factors on the appropriate initial temperature.

CHAPTER 3. NUMERICAL TESTING OF THE TWO-STAGE METHOD FOR OPTIMIZATION OF BIMETALLIC NANOSTRUCTURES

This chapter describes the results of numerical experiments with the proposed in 2.3 two-stage method for optimization of bimetallic nanostructures.

3.1. Testing conditions

Unless otherwise stated, all experiments were performed on an $Au_{100}Ag_{100}$ nanoparticle (total 200 atoms; gold and silver in a ratio of $1:1$), on a double bi-pyramid lattice consisting of 309 nodes. An exemplary configuration of this particle is illustrated in Fig. 3.1.

Since all tests are performed on the same lattice to simplify the workflow, instead of relaxation with MD (**Step 5**), all comparisons of final solutions in this chapter are made after applying additional lattice scaling (with separate coefficients for each coordinate axis x, y and z).

It is expected that for other chemical compositions and other lattices these results will remain valid, although not with the same specific temperature values.

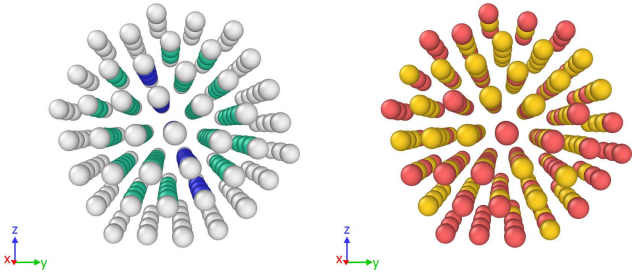


Fig. 3.1. An example configuration of an $Au_{100}Ag_{100}$ nanoparticle. Left image: blue – fcc-atoms, green – hcp-atoms, gray – unspecified atoms. Right image: yellow – Au atoms, red – Ag atoms.

This was observed in the preliminary tests carried out in the process of planning the parameters of this study, which is why it was preferred to carry out the main part of the experiments in this chapter on one lattice. However, in 3.5 one of the experiments was repeated on another lattice, for additional confirmation.

3.2. Comparing the two-stage method to only using the wide-lattice algorithm

The first task is to make a correct comparison between the proposed method and the previous approach using only the wide-lattice algorithm.

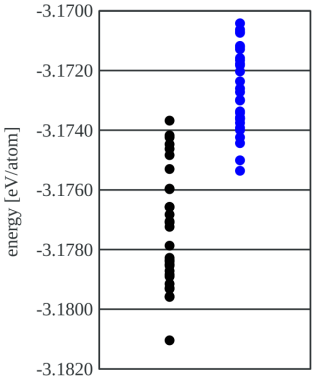
The results are presented in Fig. 3.2. The points on the left are achieved potential energy values from runs of the combined method (**Steps 1–4**). The points on the right are runs only at the first stage of the method (**Steps 1–2**), but for a larger number of iterations and at $\beta = 0$. The number of iterations is chosen so that the average duration of one trial (detected by an independent clock) in both cases is approximately the same.

It can be seen that the two-stage approach leads to a significant improvement. There is some overlap, where the more successful runs of the one-stage method give similar results to the less successful runs of the two-stage method, but it is not large. Both the average and the lowest values are better with the two-stage method, and 30 runs are more than enough to guarantee finding a lower minimum with the two-stage method.

3.3. Finding the optimal proportion for distributing computational resources between the two stages

If we assume that **Step 1** and **Step 3** take only a small but constant amount of

time, then the problem of allocating computational resources between the two stages reduces to determining the optimal proportion between the number of iterations N_2 for **Step 2** and N_4 for **Step 4**.

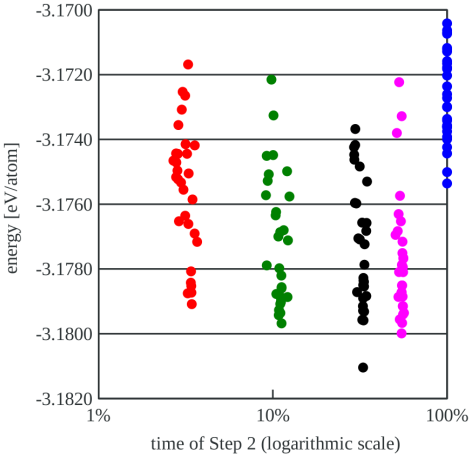


(black) dots on the left side: the combined method 30 trials	(blue) dots on the right side: only the wide-lattice algorithm 30 trials
Step 1: $N_1 = 40 \times 10^6$ **Step 2**: $N_2 = 1226 \times 10^6$; $T_2 = 2500\,\mathrm{K}$; $\beta = 4$ **Step 3**: $N_3 = 17 \times 10^6$ **Step 4**: $N_4 = 128 \times 10^6$; $T_4 = 1400\,\mathrm{K}$	**Step 1**: $N_1 = 40 \times 10^6$ **Step 2**: $N_2 = 4 \times 10^9$; $T_2 = 2500\,\mathrm{K}$; $\beta = 0$ (so that the total duration is is the same as on the left side)

Fig. 3.2. Comparison of the two-stage method (**Steps 1–4**) with the use of only the wide-lattice algorithm (**Steps 1–2**).

Fig. 3.3 shows the results of tests with different values of N_2 and N_4. To make the comparison correct, the values were chosen so that the total execution time for each attempt was approximately the same. The figure shows the actual values of the time spent (measured by an independent clock), presented in the form of a proportion: the time t_2 spent in **Step 2** divided by the time t_{2+4} spent in **Steps 2+4**.

The average total time spent t_{2+4} was indeed approximately the same for the different data columns, within $\pm 5\%$, with the third (black) column actually taking less time (-2%), although it clearly produced the best solutions. This gives further reason to assume that under the tested conditions the values $N_2 = 1226 \times 10^6$ and



1st (red), 2nd (green), 3rd (black), 4th (pink) columns of dots: the combined method 30 trials in each column
Step 1: $N_1 = 40 \times 10^6$ **Step 2**: $N_2 = 125 \times 10^6$, 400×10^6, 1226×10^6, 2052×10^6; $T_2 = 2500\,\mathrm{K}$; $\beta = 4$ **Step 3**: $N_3 = 17 \times 10^6$ **Step 4**: $N_4 = 184 \times 10^6$, 170×10^6, 128×10^6, 86×10^6; $T_4 = 1400\,\mathrm{K}$
rightmost (blue) column of points: only the wide-lattice algorithm 30 trials
(same as in Fig. 3.2, shown here for comparison)

Fig. 3.3. Comparison of different time distributions between the two stages, i.e. different proportions of time spent in **Step 2** relative to the total time spent in **Steps 2+4** for one trial.

$N_4 = 128 \times 10^6$ (corresponding to $t_2/t_{2+4} = 30\%$) were the most successful.

The figure clearly shows that the overlap between the second, third, and fourth columns is large. This means that there is no acute shortage of computational resources in one of the two algorithms (as there is, say, in the leftmost and rightmost columns). Moreover, it seems that the chosen total time t_{2+4} is sufficient for both algorithms to reach a kind of saturation, where investing more computational resources in a single attempt does not lead to a significant improvement in the results.

In conclusion, it can be said that in general it is a good strategy to allocate 30% of the computing resources to **Step 2** and 70% to **Step 4**.

3.4. Testing the approach for determining the initial temperature

This numerical experiment aims to verify whether the approach to setting the initial temperature in **Step 1** and **Step 3** is correct. In these preliminary steps, the values of T_2 and T_4 are determined at high cooling rates to be used later in **Step 2** and **Step 4** at much lower cooling rates. This is correct only if the optimal initial temperatures at high rates continue to be optimal at low rates.

Fig. 3.4 and Fig. 3.5 show the results of the operation of the wide-lattice algorithm and the diffusion algorithm, respectively, at different initial temperatures and at different cooling rates. In the figures, the points and hicks correspond to the same temperatures, but for better visibility, they are shifted slightly to the left and slightly to the right, respectively.

It can be seen that the initial temperatures giving the best results are $T_2 = 2500\,\mathrm{K}$ and $T_4 = 1400\,\mathrm{K}$. What is important here is not so much these specific values as the fact that these temperatures are optimal despite the large difference in the speed of cooling.

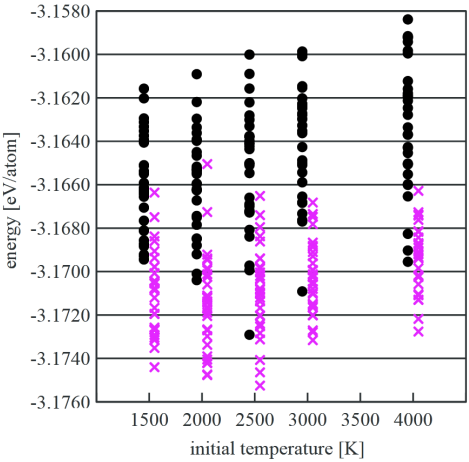
3.5. Testing on an alternative lattice

To further confirm the effect of the proposed method, the experiment from 3.2 was repeated on an alternative lattice: an icosahedral lattice with 309 nodes. The chemical composition $Au_{100}Ag_{100}$ remains the same.

The results are shown in Fig. 3.6. Similar to the previous experiments, there is some overlap between the more successful runs of the one-stage method and the less successful runs of the two-stage method, but the advantage of the two-stage method for finding the minimum is clearly expressed.

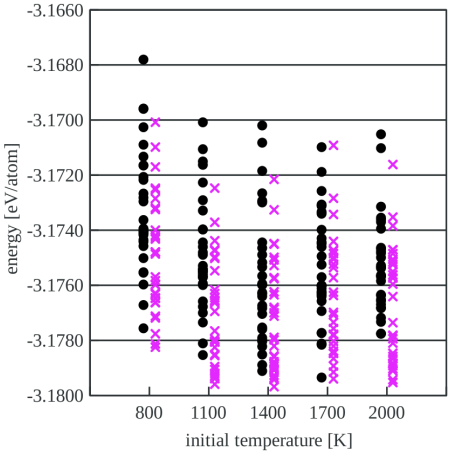
3.6. Conclusion

As a result of the experiments, it can be argued that the proposed two-stage method for optimization of bimetallic nanostructures represents a significant improvement over following a single-stage approach, and its specific formulation in 5 steps on page 13 is appropriate and leads to successful parameter adjustment. The most appropriate proportion for distributing computational resources between the two stages is 30% for the wide-lattice algorithm and 70% for the diffusion algorithm.



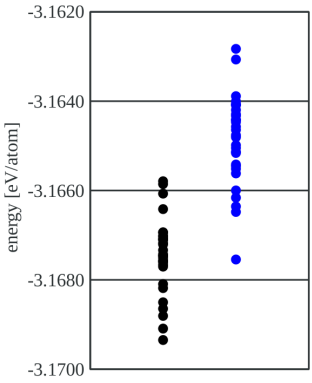
The wide-lattice algorithm. For each initial temperature:	
(black) dots on the left side: rapid cooling (40×10^6 iterations) 30 trials	(pink) hicks on the right side: slow cooling (400×10^6 iterations) 30 trials

Fig. 3.4. Comparison of the optimal initial temperature for the wide-lattice algorithm under fast cooling and under slow cooling.



The diffusion algorithm. For each initial temperature:	
(black) dots on the left side: rapid cooling (17×10^6 iterations) 30 trials	(pink) hicks on the right side: slow cooling (170×10^6 iterations) 30 trials

Fig. 3.5. Comparison of the optimal starting temperature for the diffusion algorithm under rapid cooling and under slow cooling.



(black) dots on the left side: the combined method 30 trials	(blue) dots on the right side: only the wide-lattice algorithm 30 trials
Step 1: $N_1 = 40 \times 10^6$ **Step 2**: $N_2 = 1200 \times 10^6$; $T_2 = 2500\,\mathrm{K}$; $\beta = 4$ **Step 3**: $N_3 = 13 \times 10^6$ **Step 4**: $N_4 = 130 \times 10^6$; $T_4 = 1400\,\mathrm{K}$	**Step 1**: $N_1 = 40 \times 10^6$ **Step 2**: $N_2 = 4 \times 10^9$; $T_2 = 2500\,\mathrm{K}$; $\beta = 0$ (so that the total duration is is the same as on the left side)

Fig. 3.6. Results for an icosahedral lattice: comparison of the two-stage method (**Steps 1**–**4**) with using only the wide-lattice algorithm (**Steps 1**–**2**).

CHAPTER 4. INFLUENCE OF THE INITIAL TEMPERATURE ON THE WIDE-LATTICE SIMULATED ANNEALING ALGORITHM

This chapter pays special attention to the initial temperature of the wide-lattice algorithm described in 2.2.1. In simulated annealing algorithms, one of the most important parameters is the value of the initial temperature. On the one hand, it must be high enough to allow the system to explore different global configurations without falling into a local minimum too early. But on the other hand, setting the temperature too high is a waste of resources: a lower initial temperature allows finding a solution with fewer iterations.

An important factor is the cooling rate. Furthermore, the algorithm is not deterministic and is expected to be run multiple times with different sequences

of random numbers so that the system can explore different transition paths and finally choose the best solution. This means that a trade-off is needed: with the same number of computational resources, whether to use them to set a higher initial temperature, to set a slower cooling rate, or to run more trials.

A certain fixed initial temperature value may be high for some input data, but not high enough for others. In nanoparticle optimization, factors that can have an impact are the type and size of the particle, as well as the type and size of the lattice.

The results of this chapter are the basis for the decision to formulate the two-stage method proposed in 2.3 specifically in the given manner in 5 steps. They also provide a guideline for the range of temperatures suitable for starting **Step 1**. Furthermore, they have an application separate from the two-stage method, as they can be used in studies related to the optimization of monometallic nanostructures.

The results of this chapter are published in [Mikhov et al., 2021].

4.1. Selection of chemical elements

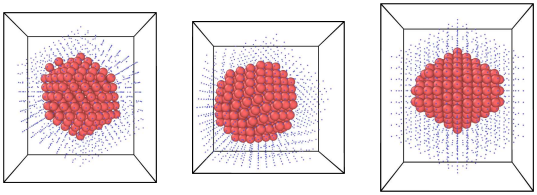
This dissertation does not aim to focus on specific metals, but rather seeks to achieve results applicable to a variety of conditions. For this reason, silver and cobalt—chemical elements with very different characteristics—were chosen for testing.

4.2. Testing conditions

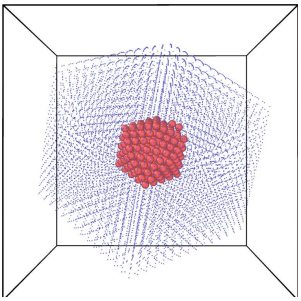
Three types of lattices were selected (icosahedral, bi-cuboctahedral and decahedral), each with two variants of different sizes (a "small" variant with 1415 nodes and a "large" variant with 10179 or 17885 nodes). Silver and cobalt nanoparticles were placed separately on each of these six lattices, varying the number of atoms from Ag_{150} to Ag_{310} and from Co_{150} to Co_{200} with a step of 10. In addition, 10 larger Ag particles (from Ag_{620} to Ag_{3000}) were tested on the three large lattices. The initial temperature (T_0) was tested in the range 1000–5000 K for the smaller lattices and 2000–5000 K for the larger ones. For each variant, 30 runs of the wide-lattice Monte Carlo algorithm were made and the average of the potential energies of the obtained final configurations was accepted as the result of the given variant. The total number of runs amounted to 24,300.

4.3. Results

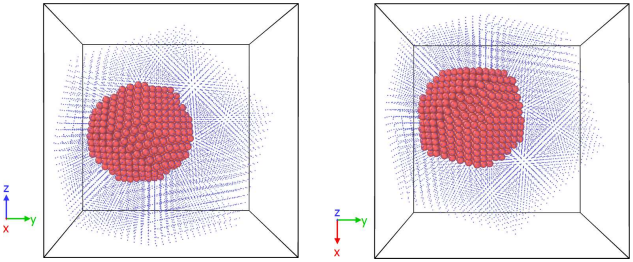
Some of the solutions obtained are shown in Fig. 4.1, where the red spheres represent Ag atoms and the blue dots are empty nodes. The results for the potential



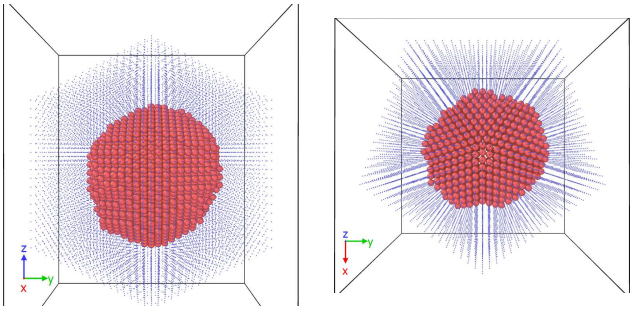
a) Ag_{310} on the small icosahedral, bi-cuboctahedral and decahedral lattices



b) Ag_{310} on the icosahedral lattice with 10179 nodes



c) Ag_{1100} on the bi-cuboctahedral lattice with 10179 nodes, viewed from two different angles



d) Ag_{3000} on the decahedral lattice with 17885 nodes, viewed from two different angles

Fig. 4.1. Some of the solutions obtained.

energy at different initial temperatures are shown for the different nanoparticles and lattices in Fig. 4.2–4.3.

For small Ag nanoparticles on small lattices (Fig. 4.2 a) the results for $T_0 =$ 5000 K, 4000 K and 3000 K completely overlap with those for 2000 K. The difference between 2000 K and 1500 K is quite small, and between 1500 K and 1000 K there is already a large difference. This means that the optimal starting temperature here is either 2000 K or 1500 K.

Is the difference between 2000 K and 1500 K significant? Note that each point in Fig. 4.2 represents the average energy value of 30 trials. In actual use of the algorithm, it is expected to be run multiple times to find the minimum solution, and then the more successful runs are of interest, not the average itself. In the present experiment the average values at $T_0 = 2000$ K and 1500 K are quite close, but $T_0 = 2000$ K consistently finds better minima than $T_0 = 1500$ K. This is partially illustrated in Fig. 4.4, where all the starts are plotted. Therefore, here the most appropriate starting temperature is 2000 K.

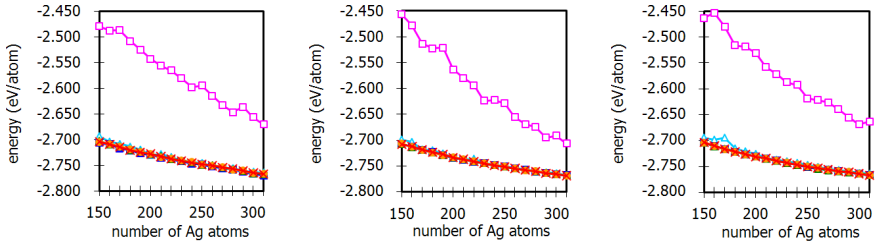
Similar reasoning applies to all other combinations of chemical elements, sizes, and lattices.

When comparing the best values of T_0 under different conditions, the data show that the most important factor considered is the lattice size. For small nanoparticles on small lattices (Fig. 4.2 a, c), the optimal temperature is around 2000–3000 K, with cobalt requiring higher temperatures than silver. For the same nanoparticles on larger lattices, however, 4000 K or even 5000 K become necessary (Fig. 4.2 b, d). The highest temperature is required for the largest lattice tested, the decahedral lattice with 17885 nodes.

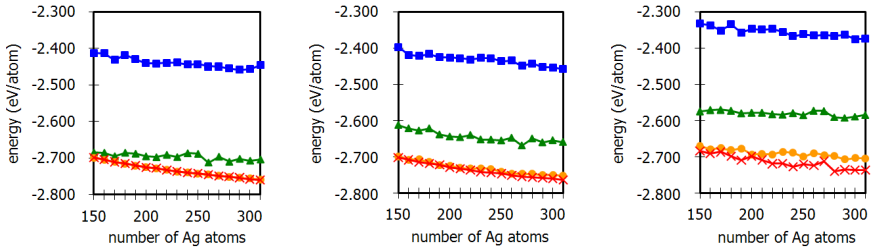
It is important to note that the lattice size factor here is relative to the size of the nanoparticle. As the size of the nanoparticle increases, it begins to occupy a significant part of the lattice and then a high initial temperature is no longer needed (Fig. 4.3). It becomes easier to find a minimum because the possible atomic configurations are fewer.

4.4. Conclusion

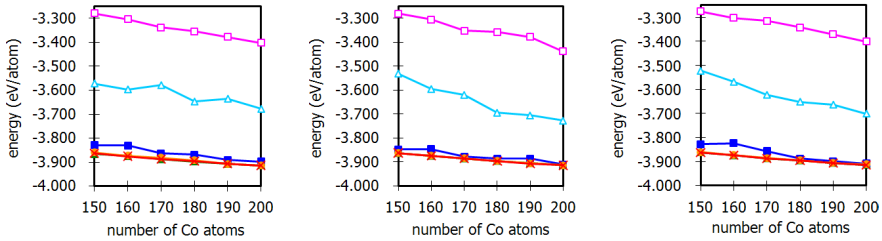
As a result of the experiments, it was found that the most important factor in determining the appropriate initial temperature is the size of the lattice, and the temperature should be especially high when placing a small particle on a large



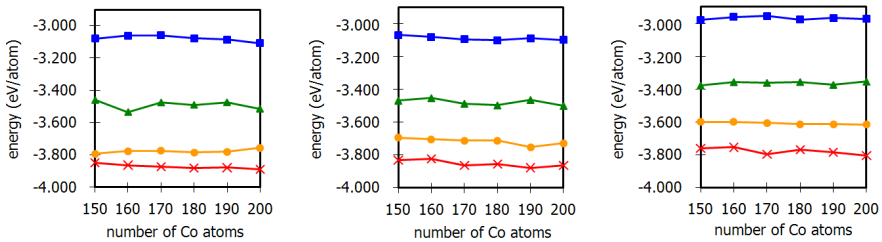
a) Ag on the small icosahedral, bi-cuboctahedral and decahedral lattices



b) Ag on the large icosahedral, bi-cuboctahedral and decahedral lattices



c) Co on the small icosahedral, bi-cuboctahedral and decahedral lattices



d) Co on the large icosahedral, bi-cuboctahedral and decahedral lattices

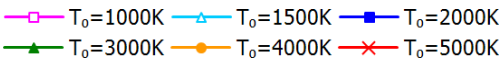


Fig. 4.2. Effect of initial temperature on the quality of the solution for small sizes of silver and cobalt.

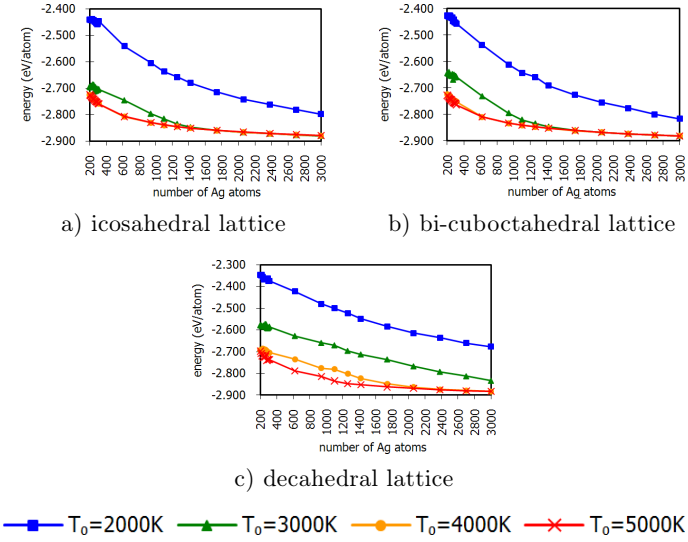


Fig. 4.3. Influence of initial temperature on the quality of the solution for large-sized silver.

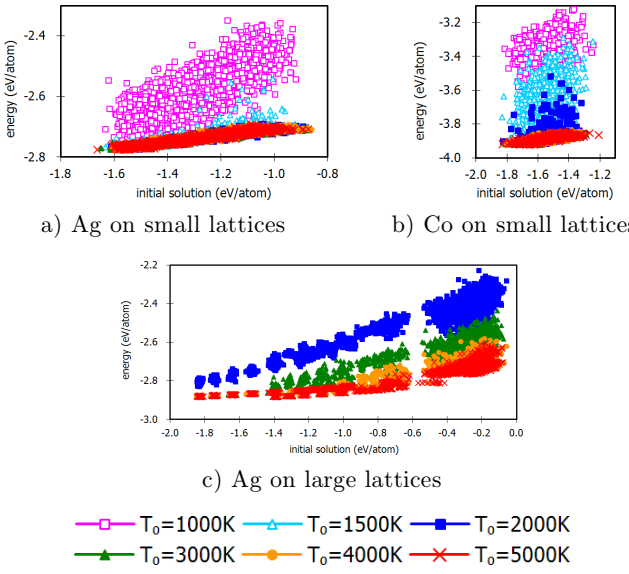


Fig. 4.4. All the data points from the experiment in this chapter. The energy of the final configuration versus the energy of the initial configuration.

Table 4.1. Suitable starting temperature for the tested particles.

	Ag_{150}–Ag_{310}	Ag_{620}–Ag_{1740}	Ag_{2060}–Ag_{3000}	Co_{150}–Co_{200}
lattices with 1415 nodes	2000 K	—	—	3000 K
lattices with 10149 nodes	4000 K	4000 K	3000 K	5000 K
lattices with 17885 nodes	5000 K	5000 K	4000 K	5000 K

lattice. The type of chemical element also has some importance. The type of lattice does not have a significant effect. The optimal initial temperature for the studied variants is summarized in Table 4.1.

CHAPTER 5. APPLICATION OF THE METHOD TO THE OPTIMIZATION OF GOLD–SILVER NANOCAGES

In this chapter, the proposed two-stage method is applied to study the atomic ordering and the processes of surface segregation and cavity healing in gold–silver nanocages of 3000 atoms [Myasnichenko et al., 2025]. The method is adapted for nanocages, using a new approach for generating lattices with variable levels of node compression. For the analysis of the results, a modification of the Warren–Cowley short-range order (SRO) parameter is proposed, in which the atomic concentrations are calculated locally, taking into account the coordination numbers of the atoms. This makes the parameter applicable to nanoparticles with pronounced surface segregation.

5.1. Selection of lattice geometry and alloy composition

In this chapter, gold–silver nanocages of 3000 atoms at ratios of Au:Ag = 1:1, 1:3 and 3:1 on fcc and icosahedral lattices are studied.

5.2. Lattice generation

To simulate an environment in which it is energetically advantageous for atoms to avoid the central zone of the lattice, we propose to use a compressed lattice with a compression coefficient that varies with the distance to the center of gravity of the system.

Specifically, two lattices (fcc and icosahedral) were generated for the present study, each containing 4500 empty nodes. In these, for a core of 1000 nodes at the center of the lattice, a compression of 9% was applied to the basis length of the coordinate vector of each node. At the boundary of the core, the compression is

8%. From there outward, the compression coefficient decreases linearly with radius until it reaches a 0.1% expansion at the outer boundary of the lattice. The exact formula is as follows:

$$\kappa_{\text{shell}}(r) = 0,92 + (1,001 - 0,92)\,\frac{r - r_{\text{core}}}{r_{\text{out}} - r_{\text{core}}}, \tag{5.1}$$

where $\kappa_{\text{shell}}(r)$ is the local scaling factor at radius r ($r_{\text{core}} \leq r \leq r_{\text{out}}$); r_{core} is the radius of the core; r_{out} is the outer radius of the lattice.

5.3. Parameters of the two-stage method

For optimization of the nanocages, the two-stage method from Chapter 2 was used, with the following adaptations. In both stages of the method, the main cycle ends after a predetermined number of iterations N. Cooling is performed once every 10,000 iterations, with the cooling rate depending on N as follows:

$$T = T_0 - (T_0 - 1)\,\frac{s}{N - 10\,000}, \tag{5.2}$$

where T is the current temperature; T_0 is the initial temperature; $s$is the current iteration number. This is a linear cooling schedule scaled so that the last 10,000 iterations are performed at 1 K.

5.3.1. Plan of the numerical experiments

For the numerical experiments, the two lattices generated in 5.2 (fcc and icosahedral, 4500 nodes each) were used. Nanocages of 3000 atoms in three compositions were tested on them: Au:Ag = 1:1, 1:3, 3:1.

In **Step 1**, for each of the two lattices and each of the three compositions, several initial temperatures T_1 were selected, for each of which 30 independent trials of the wide-lattice Monte Carlo algorithm were performed. Each trial started from a new random initial arrangement of 3000 atoms and was performed for $N_1 = 40 \times 10^6$ iterations. The lowest energy achieved in each trial was recorded. The initial temperature for which this value (averaged over the 30 trials) was the lowest was chosen as the best initial temperature T_2. No scaling was performed in this step along the x, y, z axes.

In **Step 2**, 30 independent trials of the wide-lattice Monte Carlo algorithm were performed for each lattice/composition. Each trial started from a new random initial arrangement of 3004 atoms ($\beta = 4$) and was run for $N_2 = 1 \times 10^9$ iterations. At the end of this step, but before the additional $\beta = 4$ atoms are deleted, a global

rescaling of the atomic coordinates (separately along the x, y, z axes) is performed in order to further minimize the energy function. This fixes the geometric shape of the nanocage and from now on, basically only the arrangement of the atoms changes.

In **Step 3**, for each lattice/composition, several initial temperatures T_3 are selected, for each of which 30 trials of the diffusion Monte Carlo algorithm are performed. Each trial starts from the result of the correspondingly numbered trial from **Step 2** and is performed for $N_3 = 8 \times 10^6$ iterations. The lowest energy achieved in each trial is recorded. The initial temperature for which this value (averaged over the 30 trials) is the lowest is chosen as the best initial temperature T_4. No scaling was performed in this step along the x, y, z axes.

In **Step 4**, 30 trials of the diffusion Monte Carlo algorithm were performed for each lattice/composition. Each trial started from the result of the correspondingly numbered trial from **Step 2** and was run for $N_4 = 80 \times 10^6$ iterations. At the end of this step, a final global rescaling of the atomic coordinates (separately along the x, y, z axes) was performed in order to further minimize the energy function. The configuration that achieves the lowest energy here was selected as the final result for each lattice/composition.

The final configurations achieved are shown in Fig. 5.1.

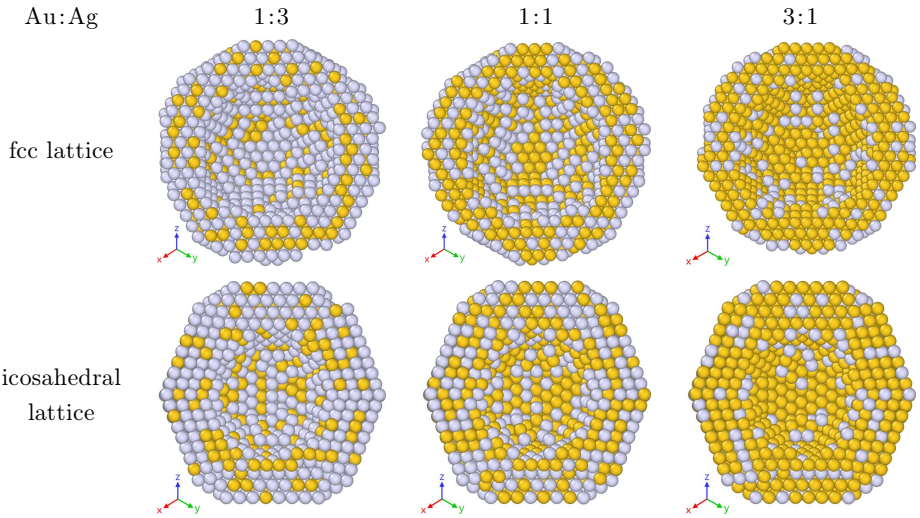


Fig. 5.1. Cross-sections of the resulting optimized nanocages with 3000 atoms. Yellow – Au atoms; gray – Ag atoms.

Step 5 is not performed because in this chapter the goal is to model nanocages without relaxation. Removing the lattice, depending on the conditions, would lead to the collapse of the nanocage.

5.4. Adaptation of the short-range order parameter

To quantitatively assess the local chemical ordering in the resulting nanocages, it is appropriate to perform a short-range order (SRO) analysis. In this chapter, we use the Warren–Cowley SRO parameter [Polgreen, 1985] as a basis, noting that the different SRO parameters are correlated with each other [Gahn and Pitsch, 1989].

According to the classical definition, the value of SRO α_i for the i-th coordination sphere around a given atom of type a is calculated as follows:

$$\alpha_i = 1 - \frac{P_{ab}(i)}{x_b}, \tag{5.3}$$

where $P_{ab}(i)$ is the proportion of neighbors (in the i-th sphere) of type b around the atom of type a, and x_b is the global proportion of atoms of type b in the system. This value α_i is averaged over all atoms in the system, regardless of their type. When $\alpha_i = 0$, this is an indicator of random mixing of chemical elements; $\alpha_i < 0$ indicates a preference for bonds between atoms of different elements; $\alpha_i > 0$ signals a preference for clustering of atoms of the same type.

The Warren–Cowley SRO parameter thus defined is appropriate for quantitatively assessing chemical ordering in macroscopic crystals, but fails to describe surface segregation in nanoparticles, as these are different phenomena governed by different energetic effects (e.g. bond tension versus surface energy minimization). To address this issue, SRO should be calculated separately for different zones (interior/subsurface/surface) and local concentrations x_b should be used, separating the internal ordering from segregation artifacts.

For this purpose, we modify the above definition as follows. For each atom of type a, we calculate x_{ab}^{local} – the local (within three coordination spheres) proportion of atoms of type b around the given atom of type a. Then, the atoms are classified into 3 zones: internal atoms, subsurface atoms, and surface atoms. The formula for SRO (5.3) is retained, but instead of x_b, an average value of x_{ab}^{local} is used, averaging over all atoms of type a in the same zone as the given one. Only α_1 is considered, and the average value of α_1 for the entire nanoparticle is not considered, but only the average value for each zone separately.

When it comes to a nanocage, due to the presence of a central cavity, we have two surfaces (external and internal), respectively two subsurfaces. This further subdivides the 3 zones into 5 subzones, which we will call "layer groups". In this case, x_b is calculated for each of the 3 zones, as explained above, but an average value of α_1 is considered for each of the 5 layer groups separately.

5.4.1. Classification of atoms by zones and layer groups

For the classification of atoms by zones, we consider the number of neighboring positions within three coordination spheres. In a macroscopic fcc lattice, each atom would have 12 neighbors in the first sphere, 6 in the second, and 24 in the third (a total of 42). If the atom is in the subsurface of a nanoparticle, some of the neighboring positions in the third sphere would remain empty. On the surface, even neighboring positions in the first sphere remain empty. In the interior, most neighboring positions would be occupied, but not completely; for particularly small nanoparticles (such as the thin-walled nanocages discussed in this chapter), it is possible that there may not even be an atom with a completely filled third sphere. Therefore, for the classification by zones, we will use the following rule:

$$\begin{array}{rcl} 38 \leq n_3 \leq 42 & \Longrightarrow & \text{interior} \\ 30 \leq n_3 \leq 37 & \Longrightarrow & \text{subsurface} \\ n_3 \leq 29 & \Longrightarrow & \text{surface,} \end{array} \tag{5.4}$$

where n_3 is the total number of occupied adjacent positions within three coordination spheres.

The subdivision of the surface group of atoms into an outer and inner surface (and especially the subdivision of the subsurface group of atoms into an outer and inner subsurface) requires the use of a geometric approach. In our case, for each atom we measure the distance to the center of gravity of the nanocage. If it is below a certain value, we classify the atom as belonging to the inner (sub)surface layer group, otherwise to the outer one.

5.5. Results

The atomic configurations obtained after optimization (Fig. 5.1) were analyzed for radial composition profiles, surface segregation indices, and pore-healing metrics (average cavity size).

5.5.1. Structural outcomes

Fig. 5.2 shows the coordination numbers and the classification by layer groups of the atoms in one of the optimized configurations for each lattice.

From the cross-sections of the optimized cages (Fig. 5.1–5.2), it is seen that the three fcc nanocages have thinner walls and the cavity radius is larger. In addition, there are more low-coordinated atoms (Fig. 5.2, top left) compared to the icosahedral cages. A significant number of atoms with coordination numbers 6 and 8 are observed on the surface, both on the outer and inner surfaces. The number of atoms in the layer group of the interior (Fig. 5.2, top right) is very small.

In the icosahedral lattice, atoms occupy more layers, including more compressed layers located closer to the center of the lattice (Fig. 5.2, bottom left). On the inner surface there are also single atoms with a coordination number less than 7. Both surfaces are formed by (111) facets and therefore the predominant coordination number on the surface is 9. In the icosahedral lattice, the layer group of the interior (Fig. 5.2, bottom right) contains a sufficiently large number of atoms.

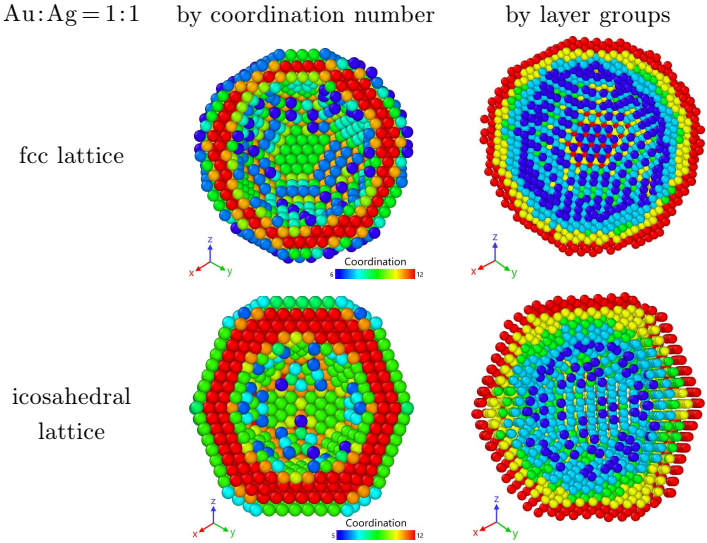


Fig. 5.2. Cross-sections of the resulting optimized nanocages in the composition Au:Ag = 1:1. Left: colored by coordination number. Right: colored by layer groups (red – outer surface; yellow – outer subsurface; green – interior; light blue – inner subsurface; blue – inner surface).

5.5.2. Local concentration of the metals and short-range order

Fig. 5.3 shows the local Au:Ag ratios at different radii in the nanocages. It can be seen that Au preferentially occupies subsurface positions, while Ag dominates in the surface layers. This is consistent with the lower surface energy of Ag [Vollath et al., 2018]. There is a pronounced surface segregation of Ag and since the total surface area (external plus internal) of the nanocages is large, most of the silver atoms end up on the surface. In the Au-enriched (3:1) composition, there are almost no silver atoms in the internal layers.

Fig. 5.4 shows the SRO values for the different layer groups in the nanocages. The influence of the composition on the SRO is more pronounced for the fcc cages due to their thinner walls.

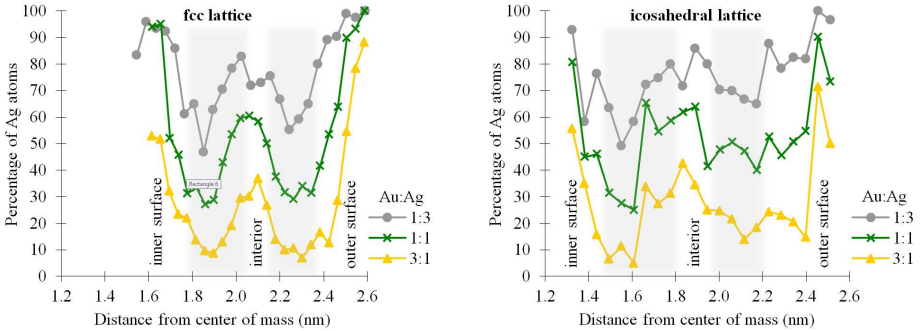


Fig. 5.3. Local Ag concentration at different radii in the nanocages. The light gray background shows the areas with a large number of subsurface atoms, according to the classification of 5.4.1.

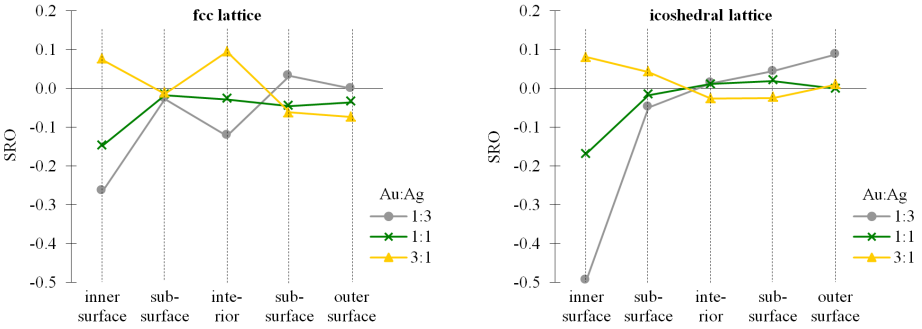


Fig. 5.4. SRO values for different layer groups in nanocages.

The icosahedral cages show more uniform mixing on the facets, but a stronger accumulation of Ag on the fivefold vertices. The Au-enriched (3:1) composition contains the highest number of mixed Au–Ag bonds on both lattices, due to the sufficient number of Ag atoms in the inner layers. The Au-enriched systems form almost close-packed shells of Au, with the Ag atoms filling the inner cavities, a situation that can improve mechanical stability.

5.5.3. Energetic outcomes

The total potential energy for the systems is summarized in Table 5.1, where E_{opt}

Table 5.1. Energy results (eV/atom).

	fcc lattice			icosahedral lattice		
Au:Ag	1:3	1:1	3:1	1:3	1:1	3:1
E_{opt}	−2.984	−3.194	−3.392	−3.024	−3.231	−3.426
$E_{\text{rand}} - E_{\text{opt}}$	0.011	0.015	0.013	0.005	0.006	0.004

is the energy of the resulting optimized configuration, and E_{rand} is the energy of a random alloy with the same geometry and composition. E_{rand} is obtained by randomly shuffling the atoms in the corresponding optimized configuration 30 times and taking the arithmetic mean of the energy. It is confirmed that the ordered configurations are more stable by 5–10 meV/atom compared to random alloys. These energy differences, although modest, can dictate thermal activation barriers in catalysis or affect melting point depressions in nanoscale materials.

5.6. Implications for applications

The results show that by appropriately selecting the Au : Ag ratio and lattice geometry, the properties of the nanocage can be regulated and tailored to specific applications:

- Catalysis: silver-rich equiatomic cages with a distinct A–B ordering can provide synergistic active centers for selective oxidation reactions.
- Plasmonics and sensing technologies: gold-rich fcc-based nanocages with surface silver can support hybridized plasmon modes and enhanced electromagnetic “hot spots”.
- Drug delivery and photothermal therapy: icosahedral gold-rich cages combine high stability with tunable pore sizes for drug cargo encapsulation and efficient light-to-heat conversion.

5.7. Limitations of the study and directions for future work

Although the current two-stage lattice Monte Carlo approach captures key phenomena in structural ordering and pore healing, it cannot account for vibrational entropy and dynamic effects. In the future, kinetic Monte Carlo on the lattice or MD sampling with machine-learned potentials could be integrated to investigate temperature-dependent diffusion processes and nanocage collapse mechanisms.

5.8. Conclusion

In this chapter, the two-stage method is applied to study the atomic ordering and related phenomena in gold–silver nanocages. Some aspects of the method are adapted to the specifics of working with nanocages. A new approach for generating lattices with variable levels of node compression is proposed, which allows modeling cavities in nanocages. A modified version of the Warren–Cowley near-order parameter is defined, which is applicable even for nanoparticles with pronounced surface segregation.

Using this approach, 3000-atom nanocages in three compositions (Au:Ag = 1:1, 1:3, 3:1) on two lattices (fcc and icosahedral) have been studied. The following conclusions can be drawn from the results for this class of nanocages:

- In fcc systems, the nanocage walls are thinner and the cavity radius is larger. There are noticeably more atoms with low coordination numbers (CN = 6–8) on the surfaces. Icosahedral structures form denser atomic layers (both inner and outer), and atoms with CN = 9 predominate on the surface.
- In the diffusion process, Ag atoms preferentially migrate to both surfaces of the nanocage, reflecting the strong surface segregation of silver.
- In Au-rich alloys (Au:Ag = 3:1) the inner layers are practically devoid of Ag, with Au–Au bonds dominating. Ag-rich alloys (Au:Ag = 3:1) have the maximum number of mixed Ag–Au bonds and a more uniform local order in both crystal lattices. The effect of composition on SRO is most noticeable in fcc-nanocages due to their thinner walls.

These effects clearly demonstrate how the interaction between the lattice symmetry and the alloy element ratio determines the local order and influences the macroscopic properties of bimetallic nanocages. The proposed two-stage Monte Carlo approach is efficient for such studies and can be easily adapted for other alloy systems, multicomponent combinations, and alternative potentials.

CHAPTER 6. SOFTWARE ARCHITECTURE AND SYSTEM FOR THE IMPLEMENTATION OF THE TWO-STAGE METHOD FOR OPTIMIZATION OF BIMETALLIC NANOSTRUCTURES

This chapter describes the requirements for the software system for the implementation of the two-stage method for optimization of bimetallic nanostructures. A software architecture is proposed and the various components of the developed system are described.

6.1. Requirements for the software system

The following requirements for the system are formulated:

- to be optimized for processors with x86-64 architecture, achieving performance that permits the execution of billions of Monte Carlo iterations within minutes, enabling the optimization of nanoparticles with size between several hundred and several thousand atoms on standard and widely-available hardware;
- to use the standard XYZ format for input and output data of atomic configurations, compatible with a large number of existing software packages, such as for 3D visualization, analysis of the results, etc.;
- to be compatible with Linux and Windows operating systems;
- to enable the formulation of complex plans for numerical experiments, which combine several processing steps with different combinations of algorithms, variation of parameters, saving of intermediate data for animated 3D visualization of the behavior of the methods, performance statistics, etc.;
- to synchronize the execution of a plan for numerical experiments concurrently using several processor cores of one computer or several computers with different operating systems, with the ability to resume normally even after unexpected interruptions such as power failures, restarting of the operating system, etc.;

- to enable a high degree of flexibility, for example the ability to modify a plan for numerical experiments while in the process of performing those experiments, where all results obtained up to that point are not lost and can be used in the execution of the modified plan.

6.2. Software architecture

The software architecture of the system contains 3 components: computational core; template for the master plan for numerical experiments; supplementary functionalities for analysis of the results, testing and visualization.

6.2.1. Computational core

The computational core is concentrated in one executable command, developed in the C programming language. The algorithms of Chapter 2 are implemented here, as well as some variations. Each execution of this command constitutes one execution of one of the steps of the proposed two-stage method.

6.2.2. Master plan for numerical experiments

The master plan for numerical experiments describes what nanoparticles will be studied, what trials will be performed, what combination of algorithms will be employed, whether parameters will be varied, what results will be collected, where they will be saved and how they will be combined and summarized for visualization. Essentially, the master plan is implemented as one Makefile with a specific structure. However, we don't just use a standard Make utility: we combine several of the less well-known modern functionalities of GNU Make with new extensions, written specifically for the present software system. In this way, Make is transformed from a clumsy instrument to an expressive domain-specific language, especially suitable for the fulfillment of the requirements of Section 6.1.

6.2.3. Supplementary functionalities for analysis of the results and testing

The supplementary functionalities are a library of GNU Make functions, implemented with the help of Perl scripts, which provide the necessary instruments for expressing the master plan for numerical experiments. In addition, a system for automated regression testing for the implementation of the algorithms is included.

6.3. Conclusion

A software architecture is proposed for the implementation of the two-stage method for optimization of bimetallic nanostructures, and a corresponding software system is developed. The architecture contains a computational core, a template for the master plan for numerical experiments, and supplementary functionalities for analysis of the results, testing and visualization. The developed software works on Linux and Windows operating systems and uses the standard XYZ format for input and output data for atomic configurations.

CONCLUSION – SUMMARY OF THE RESULTS

The present dissertation proposes a two-stage lattice Monte Carlo approach for the optimization of bimetallic nanoalloys, combining two main algorithms: simulated annealing on a wide lattice and simulated diffusion. Numerical experiments show convincingly that the proposed combined method achieves better results compared to the use of a single-stage approach. Experimentally is established that the specific formulation of the method in 5 steps is sound and leads to successful tuning of the parameters. The most appropriate proportion for distributing the computational resources between the two stages is 30% for the wide-lattice algorithm and 70% for the diffusion algorithm.

A more thorough investigation of the influence of the initial temperature on the performance of the wide-lattice algorithm for simulated annealing has been performed. A large number of nanoparticles of silver and cobalt have been tested on lattices with different shapes and sizes, with a total of 810 different combinations of conditions. The results show that the most important factor for determining the appropriate initial temperature is the size of the lattice, with especially high temperatures required when a small particle is placed on a large lattice.

In order to demonstrate the applicability of the proposed two-stage approach, the method has been adapted for nanocages and has been used for studying the atomic ordering and the processes of surface segregation in gold–silver nanocages with 3000 atoms. A new approach is proposed for lattice generation, with varying levels of compression of the nodes, which allows modeling of cavities in the nanocages. A modified version of the Warren–Cowley short-range order parameter is defined, which is applicable even for nanoparticles with pronounced surface

segregation. A comparative analysis has been performed on the results for three compositions (Au:Ag = 1:1, 1:3, 3:1) and two lattices (fcc and icosahedral), which shows that fcc nanocages have thinner walls, cavities with larger radii and more low-coordinated atoms on the surface compared to icosahedral cages. In the process of diffusion, Ag atoms show a tendency to migrate to the two surfaces of the nanocage, as a result of which the interior layers of the Au-enriched alloys are practically devoid of Ag, making Au–Au bonds dominant. Ag-enriched alloys have a maximum number of mixed Ag–Au bonds and a more uniform local order on both lattices.

The two-stage method and its variations have been implemented in a software system. A software architecture is proposed, which allows a high degree of optimizability for performance of the computations, flexibility for concurrent execution and combination of the constituent algorithms in different conditions, and good compatibility with external applications for analysis and visualization of the results. The developed software works on Linux and Windows operating systems and uses the standard XYZ format for input and output data for atomic configurations.

As a future development of this research, it is planned to recalibrate the constants of the TB potential function based on newly available results from DFT calculations of specific particles, so that the two-stage method reproduces important aspects of their configurations and allows experimentation with similar particles with explicit confidence level data.

CONTRIBUTIONS

The main results presented in this dissertation can be summarized as follows:

1. A two-stage lattice Monte Carlo method is proposed for optimization of bimetallic nanostructures, including nanoparticles, nanowires and nanofilms. The first stage is simulated annealing on a wide lattice and the second stage is simulated diffusion. The method is implemented with the help of data structures and a preprocessing strategy, which significantly increase its efficiency, and allow the optimization of nanostructures from several hundred to several thousand atoms on a standard personal computer.
2. Experimentally is established that an effective strategy for distributing computational resources between the two stages of the method is to use 30% of the time on the first stage and 70% of the time on the second stage.

3. Experimentally is established that the specific way in which the method is formulated in 5 steps is sound and leads to successful tuning of the parameters.

4. The influence of the initial temperature on the performance of the wide-lattice Monte Carlo algorithm is investigated for different lattices and chemical elements. Experimentally is established that the most important factor for choosing appropriate initial temperature is the lattice size, with the highest temperatures being required when placing a small particle on a large lattice. The type of chemical element is also of certain importance, while the type of lattice does not have a significant effect.

5. An adaptation of the two-stage method is made for nanocages. It is used for studying the atomic ordering and the processes of surface segregation in gold–silver nanocages with 3000 atoms. A comparative analysis of the results for three compositions ($\mathrm{Au{:}Ag} = 1{:}1, 1{:}3, 3{:}1$) and two lattices (fcc and icosahedral) shows that fcc nanocages have thinner walls, cavity with a larger radius, and more low-coordinated atoms on the surface in comparison to icosahedral cages. Ag atoms show a tendency to migrate to the two surfaces of the nanocage, as a result of which the interior layers of the Au-enriched alloys are practically devoid of Ag and Au–Au bonds dominate. Ag-enriched alloys have a maximum number of mixed Ag–Au bonds and a more uniform local order on both lattices.

6. A software architecture is proposed for the implementation of the two-stage method, which allows a high degree of optimizability for performance of the computations, flexibility for concurrent execution and combination of the constituent algorithms in different conditions, and good compatibility with external applications for analysis and visualization of the results.

LIST OF PUBLICATIONS ON THE DISSERTATION

1. R. **Mikhov**, V. Myasnichenko, L. Kirilov, N. Sdobnyakov, P. Matrenin, D. Sokolov, S. Fidanova (**2020**). A two-stage Monte Carlo approach for optimization of bimetallic nanostructures. In: *Proceedings of the Federated Conference on Computer Science and Information Systems: FedCSIS 2020* (M. Ganzha, L. Maciaszek, M. Paprzycki, eds.), pp. 285–288. Annals of Computer Science and Information Systems, vol. 21. IEEE. ⟨https://doi.org/10.15439/2020F135⟩

2. R. **Mikhov**, V. Myasnichenko, S. Fidanova, L. Kirilov, N. Sdobnyakov (**2021**). Influence of the temperature on simulated annealing method for metal nanoparticle structures optimization. In: *Advanced Computing in Industrial Mathematics: BGSIAM 2018* (I. Georgiev, H. Kostadinov, E. Lilkova, eds.), pp. 278–290. Studies in Computational Intelligence, vol. 961. Springer International Publishing, Cham. ⟨https://doi.org/10.1007/978-3-030-71616-5_25⟩
3. R. **Mikhov**, V. Myasnichenko, L. Kirilov, N. Sdobnyakov, P. Matrenin, D. Sokolov, S. Fidanova (**2022**). On the problem of bimetallic nanostructures optimization: an extended two-stage Monte Carlo approach. In: *Recent Advances in Computational Optimization: WCO 2020* (S. Fidanova, ed.), pp. 235–250. Studies in Computational Intelligence, vol. 986. Springer International Publishing, Cham. ⟨https://doi.org/10.1007/978-3-030-82397-9_12⟩
4. V. **Myasnichenko**, R. Mikhov, N. Sdobnykov, L. Kirilov, A. Bazulev (**2025**). Lattice Monte Carlo simulation of atomic ordering in gold-silver nanocages. *WSEAS Transactions on Electronics*, vol. 16, pp. 157–167. ⟨https://doi.org/10.37394/232017.2025.16.16⟩

LIST OF DISCOVERED CITATIONS OF THE PUBLICATIONS

The following papers cite Mikhov et al. (2020):

1. M. **Dai**, X. Feng, H. Yu, W. Guo (**2022**). A novel spectral ensemble clustering algorithm based on social group migratory behavior and emotional preference. In: *Knowledge Science, Engineering and Management: KSEM 2022*, vol. 3 (G. Memmi, B. Yang, L. Kong, T. Zhang, M. Qiu, eds.), pp. 316–328. Lecture Notes in Computer Science, vol. 13370. Springer International Publishing, Cham. ⟨https://doi.org/10.1007/978-3-031-10989-8_25⟩
2. M. **Dai**, X. Feng, H. Yu, W. Guo, X. Li (**2023**). A Monte Carlo manifold spectral clustering algorithm based on emotional preference and migratory behavior. *Applied Intelligence*, vol. 53(16), pp. 19742–19764. ⟨https://doi.org/10.1007/s10489-023-04484-w⟩
3. V. **Todorov** (**2022**). An overview of lattice and adaptive approaches for multidimensional integrals. In: *Recent Advances in Computational Optimization: WCO 2021* (S. Fidanova, ed.), pp. 333–348. Studies in Computational Intelligence, vol. 1044. Springer International Publishing, Cham. ⟨https://doi.org/10.1007/978-3-031-06839-3_19⟩

The following papers cite Mikhov et al. (2022):

4. D. **Rapetti**, C. Roncaglia, R. Ferrando (**2023**). Optimizing the shape and chemical ordering of nanoalloys with specialized walkers. *Advanced Theory and Simulations*, vol. 6(9), pp. 2300268-1–2300268-13. ⟨https://doi.org/10.1002/adts.202300268⟩
5. C. **Roncaglia** (**2024**). *Development and application of computational methods for the investigation of the structures of metal nanoparticles*. PhD thesis, Physics Department, University of Genoa, Italy. ⟨https://hdl.handle.net/20.500.14242/67950⟩

6. V. **Todorov** (**2022**). An overview of lattice and adaptive approaches for multidimensional integrals. In: *Recent Advances in Computational Optimization: WCO 2021* (S. Fidanova, ed.), pp. 333–348. Studies in Computational Intelligence, vol. 1044. Springer International Publishing, Cham. ⟨https://doi.org/10.1007/978-3-031-06839-3_19⟩
7. V. **Todorov**, I. Dimov, R. Georgieva (**2022**). Advanced biased stochastic approach for solving Fredholm integral equations. In: *Recent Advances in Computational Optimization: WCO 2021* (S. Fidanova, ed.), pp. 349–371. Studies in Computational Intelligence, vol. 1044. Springer International Publishing, Cham. ⟨https://doi.org/10.1007/978-3-031-06839-3_20⟩

BIBLIOGRAPHY

1. W. **Cai**, X. **Shao** (**2002**). A fast annealing evolutionary algorithm for global optimization. *Journal of Computational Chemistry*, vol. 23(4), pp. 427–435. ⟨https://doi.org/10.1002/jcc.10029⟩
2. C. **Chen**, Y. Zuo, W. Ye, X. Li, Z. Deng, S. P. Ong (**2020**). A critical review of machine learning of energy materials. *Advanced Energy Materials*, vol. 10(8), pp. 1903242-1–1903242-36. ⟨https://doi.org/10.1002/aenm.201903242⟩
3. F. **Cleri**, V. **Rosato** (**1993**). Tight-binding potentials for transition metals and alloys. *Physical Review B*, vol. 48(1), pp. 22–33. ⟨https://doi.org/10.1103/PhysRevB.48.22⟩
4. U. **Gahn**, W. **Pitsch** (**1989**). Correlations between short-range order parameters during short-range order reactions. *Acta Metallurgica*, vol. 37(9), pp. 2455–2462. ⟨https://doi.org/10.1016/0001-6160(89)90042-4⟩
5. M. C. **Giménez**, W. **Schmicker** (**2011**). Monte Carlo simulation of nanowires of different metals and two-metal alloys. *The Journal of Chemical Physics*, vol. 134(6), pp. 064707-1–064707-6. ⟨https://doi.org/10.1063/1.3549900⟩
6. S. K. **Gregurick**, M. H. Alexander, B. Hartke (**1996**). Global geometry optimization of $(Ar)_n$ and $B(Ar)_n$ clusters using a modified genetic algorithm. *The Journal of Chemical Physics*, vol. 104(7), pp. 2684–2691. ⟨https://doi.org/10.1063/1.470990⟩
7. J. **Guevara**, A. M. Llois, M. Weissmann (**1995**). Model potential based on tight-binding total-energy calculations for transition-metal systems. *Physical Review B*, vol. 52(15), pp. 11509–11516. ⟨https://doi.org/10.1103/PhysRevB.52.11509⟩
8. R. P. **Gupta** (**1981**). Lattice relaxation at a metal surface. *Physical Review B*, vol. 23(12), pp. 6265–6270. ⟨https://doi.org/10.1103/PhysRevB.23.6265⟩
9. H. **Jiang**, W. Cai, X. Shao (**2002**). A random tunneling algorithm for the structural optimization problem. *Physical Chemistry Chemical Physics*, vol. 4(19), pp. 4782–4788. ⟨https://doi.org/10.1039/B206251H⟩
10. G. **Kovács**, S. M. Kozlov, K. M. Neyman (**2017**). Versatile optimization of chemical ordering in bimetallic nanoparticles. *The Journal of Physical Chemistry C*, vol. 121(20), pp. 10803–10808. ⟨https://doi.org/10.1021/acs.jpcc.6b11923⟩

11. X.-Y. **Li**, B. Zhu, R. Qi, Y. Gao (**2019**). Real-time simulation of nonequilibrium nanocrystal transformations. *Advanced Theory and Simulations*, vol. 2(1), pp. 1800127-1–1800127-8. ⟨https://doi.org/10.1002/adts.201800127⟩
12. W. **Liu**, P. Chen, R. Qiu, M. Khan, J. Liu, M. Hou, J. Duan (**2017**). A molecular dynamics simulation study of irradiation induced defects in gold nanowire. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, vol. 405, pp. 22–30. ⟨https://doi.org/10.1016/j.nimb.2017.05.016⟩
13. J. **Ma**, J. E. **Straub** (**1994**). Simulated annealing using the classical density distribution. *The Journal of Chemical Physics*, vol. 101(1), pp. 533–541. ⟨https://doi.org/10.1063/1.468163⟩
14. R. **Mikhov**, V. Myasnichenko, L. Kirilov, N. Sdobnyakov, P. Matrenin, D. Sokolov, S. Fidanova (**2020**). A two-stage Monte Carlo approach for optimization of bimetallic nanostructures. In: *Proceedings of the Federated Conference on Computer Science and Information Systems: FedCSIS 2020* (M. Ganzha, L. Maciaszek, M. Paprzycki, eds.), pp. 285–288. Annals of Computer Science and Information Systems, vol. 21. IEEE. ⟨https://doi.org/10.15439/2020F135⟩
15. R. **Mikhov**, V. Myasnichenko, S. Fidanova, L. Kirilov, N. Sdobnyakov (**2021**). Influence of the temperature on simulated annealing method for metal nanoparticle structures optimization. In: *Advanced Computing in Industrial Mathematics: BGSIAM 2018* (I. Georgiev, H. Kostadinov, E. Lilkova, eds.), pp. 278–290. Studies in Computational Intelligence, vol. 961. Springer International Publishing, Cham. ⟨https://doi.org/10.1007/978-3-030-71616-5_25⟩
16. R. **Mikhov**, V. Myasnichenko, L. Kirilov, N. Sdobnyakov, P. Matrenin, D. Sokolov, S. Fidanova (**2022**). On the problem of bimetallic nanostructures optimization: an extended two-stage Monte Carlo approach. In: *Recent Advances in Computational Optimization: WCO 2020* (S. Fidanova, ed.), pp. 235–250. Studies in Computational Intelligence, vol. 986. Springer International Publishing, Cham. ⟨https://doi.org/10.1007/978-3-030-82397-9_12⟩
17. V. **Myasnichenko**, L. Kirilov, R. Mikhov, S. Fidanova, N. Sdobnyakov (**2019**). Simulated annealing method for metal nanoparticle structures optimization. In: *Advanced Computing in Industrial Mathematics: BGSIAM 2017* (K. Georgiev, M. Todorov, I. Georgiev, eds.), pp. 277–289. Studies in Computational Intelligence, vol. 793. Springer International Publishing, Cham. ⟨https://doi.org/10.1007/978-3-319-97277-0_23⟩
18. V. **Myasnichenko**, N. Sdobnyakov, L. Kirilov, R. Mikhov, S. Fidanova (**2020**). Structural instability of gold and bimetallic nanowires using Monte Carlo simulation. In: *Recent Advances in Computational Optimization* (S. Fidanova, ed.), pp. 133–145. Studies in Computational Intelligence, vol. 838. Springer International Publishing, Cham. ⟨https://doi.org/10.1007/978-3-030-22723-4_9⟩
19. V. **Myasnichenko**, R. Mikhov, L. Kirilov, N. Sdobnykov, D. Sokolov, S. Fidanova (**2022**). Simulation of diffusion processes in bimetallic nanofilms. In: *Recent Advances in Computational Optimization: WCO 2020* (S. Fidanova, ed.), pp. 221–233. Studies in Computational Intelligence, vol. 986. Springer International Publishing, Cham. ⟨https://doi.org/10.1007/978-3-030-82397-9_11⟩

20. V. **Myasnichenko**, R. Mikhov, N. Sdobnykov, L. Kirilov, A. Bazulev (**2025**). Lattice Monte Carlo simulation of atomic ordering in gold-silver nanocages. *WSEAS Transactions on Electronics*, vol. 16, pp. 157–167. ⟨https://doi.org/10.37394/232017.2025.16.16⟩
21. L. O. **Paz-Borbón**, T. V. Mortimer-Jones, R. L. Johnston, A. Posada-Amarillas, G. Barcaro, A. Fortunelli (**2007**). Structures and energetics of 98 atom Pd–Pt nanoalloys: potential stability of the Leary tetrahedron for bimetallic nanoparticles. *Physical Chemistry Chemical Physics*, vol. 9(38), pp. 5202–5208. ⟨https://doi.org/10.1039/B707136A⟩
22. T. L. **Polgreen** (**1985**). Monte Carlo simulation of short range order relevant to CuAu. *Acta Metallurgica*, vol. 33(2), pp. 185–189. ⟨https://doi.org/10.1016/0001-6160(85)90136-1⟩
23. G. **Rossi**, R. **Ferrando** (**2017**). Combining shape-changing with exchange moves in the optimization of nanoalloys. *Computational and Theoretical Chemistry*, vol. 1107, pp. 66–73. ⟨https://doi.org/10.1016/j.comptc.2017.01.002⟩
24. G.-F. **Shao**, M. Zhu, Y.-L. Shangguan, W.-R. Li, C. Zhang, W.-W. Wang, L. Li (**2017**). Structural optimization of Au–Pd bimetallic nanoparticles with improved particle swarm optimization method. *Chinese Physics B*, vol. 26(6), pp. 063601-1–063601-9. ⟨https://doi.org/10.1088/1674-1056/26/6/063601⟩
25. D. **Vollath**, F. D. Fischer, D. Holec (**2018**). Surface energy of nanoparticles – influence of particle size and structure. *Beilstein Journal of Nanotechnology*, vol. 9, pp. 2265–2276. ⟨https://doi.org/10.3762/bjnano.9.211⟩
26. D. J. **Wales**, J. P. K. **Doye** (**1997**). Global optimization by basin-hopping and the lowest energy structures of Lennard-Jones clusters containing up to 110 atoms. *The Journal of Physical Chemistry A*, vol. 101(28), pp. 5111–5116. ⟨https://doi.org/10.1021/jp970984n⟩