Computational and Mathematical Modeling Seminar

Summer Semester 2022

And the memories bring back memories bring back you...

Anna Garoufali, Nima Mary Parekkadan, Quang Ninh Le,
Ruei-Bo Chen, Simon Paulus, Vasileios Polydoras

Advisors: Prof. Dr. Robert Spatschek, Roland Sandt

RWTH Aachen University

November 16, 2022

Contents

1 Introduction

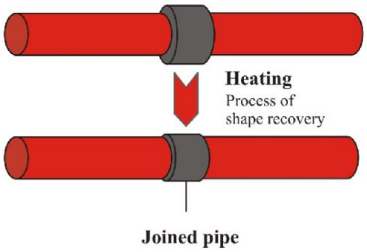
1.1 Shape Memory Alloys

In today's modern scenario Shape Memory Alloys(SMA) plays a very significant role due to its contribution towards Science and Technology. They can take the shape of a ribbon, wire, spring etc...An alloy is a mixture of materials. Shape Memory Alloys comes under a special class of alloys, which can be deformed(strain applied) at low temperatures but regains its original configuration at high temperatures(by heating). Thus called an alloy that remembers its past. They also have a property called superelasticity. For example when rubber is stretched, it gets back to its original position as soon as the mechanical force is removed. The most popular alloy constitutes the mixture of Nickel and Titanium.Also known as Nitinol. NiTi alloy is prominent because of its shape memory effect, corrosion resistance, durability, superelasticity. Moreover, it can also tolerate large amounts of shape memory strain.

1.2 Applications of Shape Memory Alloys

They are used in wide variety of applications from NiTi(Nickel-Titanium) used for eyeglass frames to Shape Memory Coupling for connecting of pipes.

(a) NiTi used for joining pipes



(b) Schematic representation of shape recovery process

Figure 1: Shape Memory Alloys

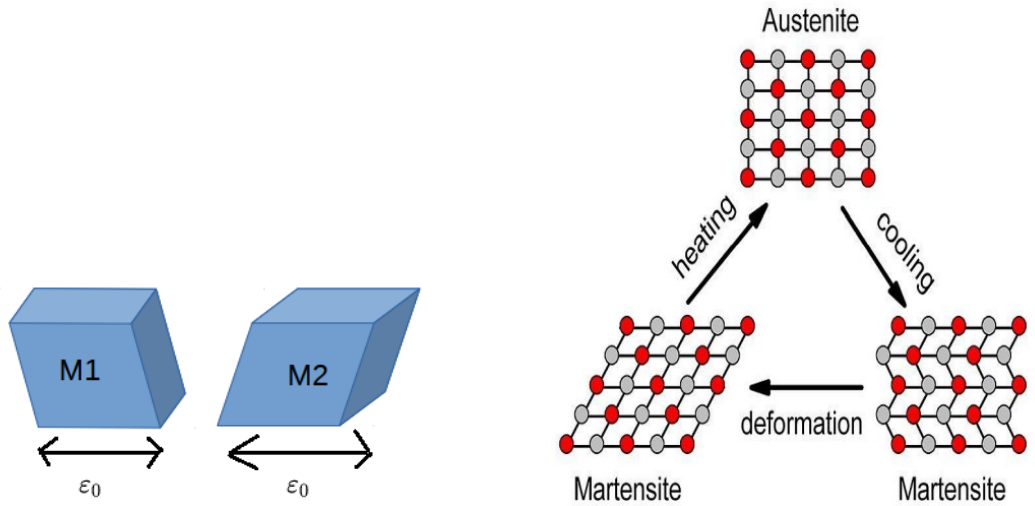
1.2.1 Shape Memory Coupling

In this technique, we connect pipes using Shape Memory Alloys. Here, we choose an internally ribbed sleeve of an alloy that has a diameter smaller than the other two red pipes seen in figure 1. The sleeve undergoes deformation at very low temperatures resulting in a diameter higher than the other two pipes. After fitting,the sleeve is heated, which results in shrinking back to its original position[1].

[1] [Wi]

1.3 Shape Memory Effect

The shape memory effect (SME) is a phenomenon, in which a material retains its original size and shape when heated above a certain characteristic transformation temperature. The temperature is material-specific i.e.they vary from material to material.
Shape Memory Effect consists of two stages:
a)Austenite Phase -High temperature phase
b)Martensite Phase-Low temperature phase
In this phase, there can be a positive flip called variant right or a negative flip called variant left.



(a) Left and right flip or Binary States

(b) Schematic representation of Shape Memory Effect

Figure 2: Change of microstructure in a material

1.3.1 Explanation of Shape Memory Effect using an example

Consider a paperclip in Martensite undergoing deformation at a low temperature.In this phase, there is a change in the microstructure of the material with a flip towards the right. Later, the insertion of deformed material in the hot water brings the paperclip in a unique phase called Austenite, where the material retains its original position. Then, allow the material to cool down which takes back to the Martensite with alternate flips to the right and left.

1.4 Scientific Background

In SMA, both changing temperature and mechanical deformations can change the microstructure of the material. Furthermore, to model the microstructure of SMA, elastic interaction between variants inside the material is considered. Elastic interaction coupled with thermodynamic description predicts in particular the equilibrium microstructure by minimizing the (strain) energy.

The challenge
This work aims to minimize the energy of the Shape Memory Alloy as well as to find the optimal configuration by using different model approaches by taking into account the computational time and accuracy.

2 The material structure in 1D and 2D

2.1 The material structure in 1D

For simplicity we analyze the material microstructure into cubes with the constant strain ε_0 for each variant, as shown in figure 3 If the cubes shift to the right, they have the value

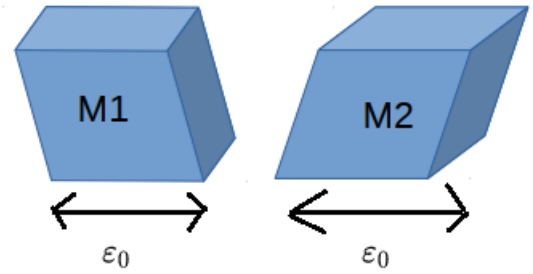


Figure 3: Two cubes of the material microstructure

$+1$, and if the cubes shift to the left, they have the value -1. As we can see in figure 4,

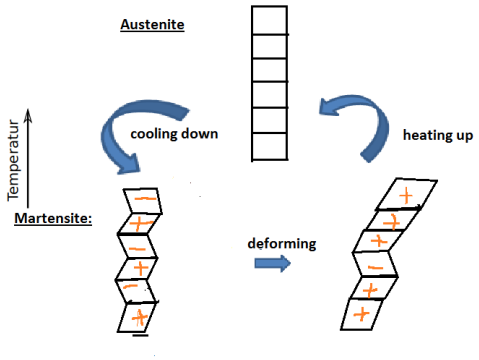


Figure 4: Austenite and martensite phase in 1D

in the austenite phase the material microstructure has only one structure in 1D. After we cool the material down , they will turn to martensite phase and have the zig-zag with the positive sign and the negative sign. And then when we deform the material , the cells in the material microstructure will take a spin and have the different signs. At the end,when we heat the material up, it will reshape immediately.

$$\begin{aligned} \text{Minimize} \quad & E(\sigma) = \frac{1}{2}\mu(x - \sum_{i=1}^{n} \sigma_i \varepsilon_0)^2 \\ & = \frac{1}{2}\mu x^2 - \mu\varepsilon_0 x \sum_{i=1}^{n} \sigma_i + \frac{1}{2}\mu\varepsilon_0^2 \sum_{i,j=1}^{n} \sigma_i\sigma_j \\ \text{s.t.} \quad & \sigma \in \{\pm 1\}^n; \end{aligned} \tag{1}$$

From now on, as shown in equation 1, the energy is expressed with the spin-spin interaction. In the energy equation σ has the value$\{\pm 1\}$ and it describes in which direction the cell takes a spin. Besides μ is shear modulus which is a measure of the elastic shear stiffness of a material. Additionally x is the start point and $x_{restpoint} := \sum_{i=1}^{n} \sigma_i \varepsilon_0$. Basically, the energy equation above bases on the energy formula which is given by equation 2.

$$\begin{aligned} E\text{–}E_0 &= \frac{1}{2}\mu\varepsilon^2 \\ \text{s.t.} \quad \varepsilon &= \frac{l\text{–}l_0}{l_0} \end{aligned} \tag{2}$$

We try to use the description to predict the equilibrium microstructure by minimization of energy, since in the nature every material tends to arrange itself in a way that minimizes the energy.

2.2 The material structure in 2D

Now we move to the 2D model and we see how the material can be presented as a 2D grid cells. In this case we have to consider the matrix J which is the pairwise interaction between the cells. In details, the element of the matrix J describes the interaction between two cells i, j of the material microstructure given by $J_{i,j} \in \mathbb{R}$.

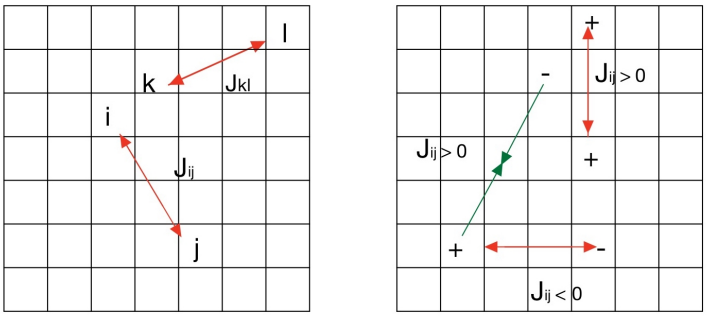


Figure 5: The interaction between two cells

As a result, with just two cells i, j the energy is given by

$$\sigma_i \sigma_j J_{i,j}, \tag{3}$$

and because in nature every material tends to arrange itself in a way that minimizes the internal energy, this means that

- $J_{i,j} > 0 \rightarrow \sigma_i$ and σ_j want to have different sign.

- $J_{i,j} < 0 \rightarrow \sigma_i$ and σ_j want to have the same sign.

We have to consider our problem in 2D with the matrices J and the state of cells σ, as shown in equation 4.

$$\begin{aligned} \text{Minimize} \quad & E(\sigma) = \frac{1}{2} \sum_{i,j=1}^{n} J_{i,j} \sigma_i \sigma_j \\ \text{s.t.} \quad & \sigma \in \{\pm 1\}^n \end{aligned} \tag{4}$$

To solve the current problem in 2D we have 4 approaches: Brute force algorithm, feed-forward neutral network, simulated annealing and quantum annealing.

3 Method 1: Brute Force Algorithm

In this approach we have to look at every single combination of σ to find out which one minimizes the energy. Therefore we have to find a way to iterate through all configurations. Take 2D case with 4 cells for example.

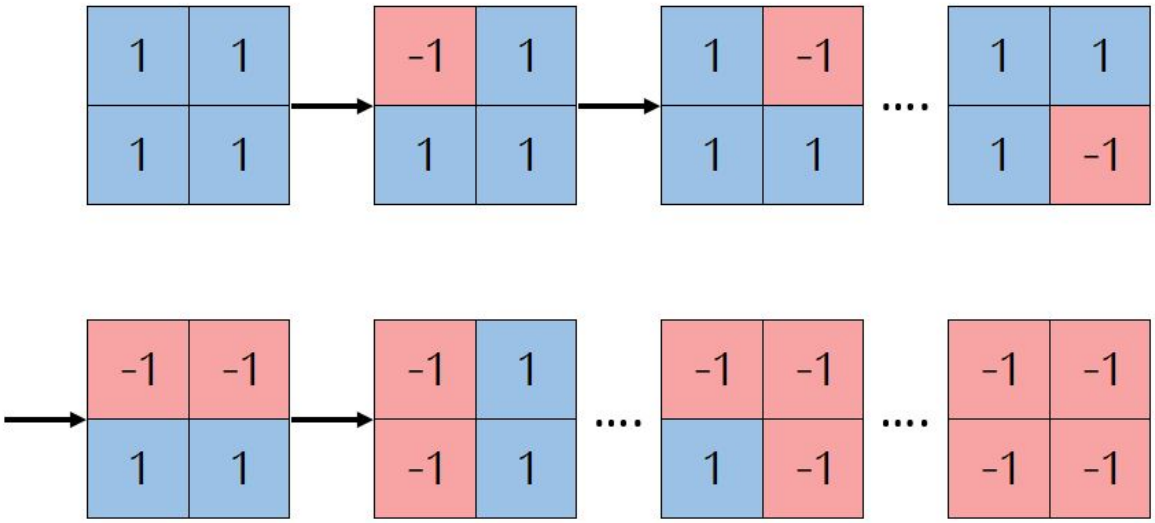


Figure 6: How brute force method works

Figure 6 visualizes how we use brute force method to find the global minimum. At each iteration, we change 1 spin, then calculate its corresponding energy and store it if this is the lowest energy we have ever obtained. Although it's simple and guarantees us the best solution, it takes very long time. The number of total possible combination grows exponentially as the number of total spins increases. In the example case, we have 4 spins, so it consequently brings totally $2^4 = 16$ different combination. Thus, the method's time complexity is $\mathcal{O}(2^n)$.

4 Method 2: Feedforward Neural Network

4.1 Topology

The initial energy minimization problem has also been solved using a deep learning algorithm that predicts the optimal spins of 4 cells that minimize the energy in the 2D case. In case of 4 cells equation 4 is reformed to equation 5.

$$\begin{aligned} &\text{Minimize} \quad E(\sigma) = \frac{1}{2}\left(J_{1,1}\sigma_1\sigma_1 + J_{2,2}\sigma_2\sigma_2 + J_{3,3}\sigma_3\sigma_3\right) + \\ &{}_{1,2}\sigma_1\sigma_2 + J_{1,3}\sigma_1\sigma_3 + J_{2,3}\sigma_2\sigma_3 + J_{3,3}\sigma_3\sigma_3 \\ &\text{s.t.} \qquad\qquad \sigma \in \{\pm 1\}^4 \end{aligned} \tag{5}$$

To tackle the problem, we implemented a feedforward neural network with 2 hidden layers of 20 neurons each and an additional Dropout layer. The addition of a Dropout layer is a regularization technique which helps reducing interdependent learning amongst the neurons. The network based on equation 5 accepts 6 inputs, which are arbitrary normalized coefficient interactions $\left|\mathrm{J}_{1,1},\ J_{1,2},\ J_{1,3},\ J_{2,2},\ J_{3,3}\ \right|$ and produces as output an 1 dimensional vector containing the spins of four cells $\left|\sigma_1,\ \sigma_2,\ \sigma_3,\ \sigma_4\ \right|$. The topology of the feedforward neural network can be shown in figure 7.

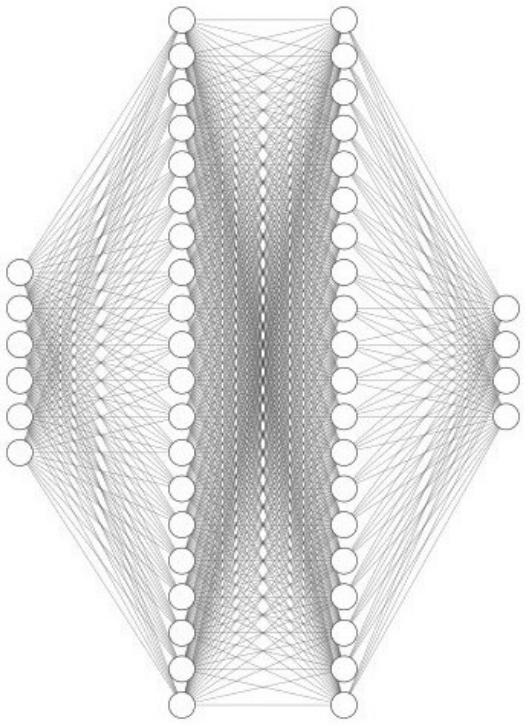


Figure 7: Feedforward Neural Network

In both hidden layers and in the output layer the activation function that has been used is the hyperbolic tangent function $tanh(x)$. Simply put, an activation function is a function that is added in the artificial neural network in order to help the network learn complex patterns in the data.

4.2 Results

For the training process we used 200 data samples extracted from the Brute Force algorithm for 4 cells. From this number of samples 75% is used for the training process, 15% for test data and 10% for validation data. The neural network was trained for 600 epochs with 85% accuracy and loss 0.0290. The validation data are used as an indicator of overfitting, which can happen when the model learns the details and noises of the training data, so that it affects negatively the performance of the model in different datasets. Therefore, validation data are used to check if we have overfitting or not during the training process. In 600 epochs we get a validation accuracy of 87% which is very good indicator that there is no overfitting in our model. Finally, test data are used in order to test our model performance. Test accuracy of 85% indicates a very good performance of the model. A visualization of the model's accuracy and loss both in training and test sets can be seen in figure 8.
This model has good performance but has a limitation in predicting the optimal spins of the cells for only four cells. The difficulty of generalization of the model in different number of cells comes from the difficulty of extracting the dataset for high number of cells because of the exponential increase of computational effort and runtime of the Brute force algorithm with respect to the number of cells. Another drawback of this model is that the optimal spins are predicted with a certain accuracy, i.e we are not always getting the correct results.

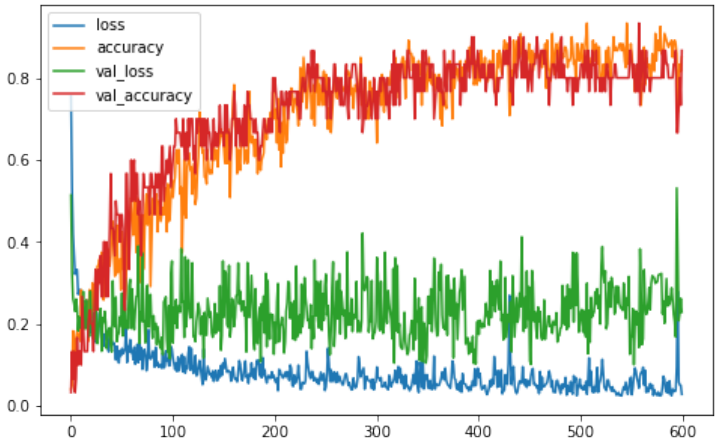


Figure 8: Comparison between accuracy, validation accuracy, loss and validation loss

5 Method 3: Simulated Annealing

Since brute force method is significantly time-consuming and the neural network method is not general enough for the case of different spins, we proposed another more efficient and general algorithm to find the global minimum, simulated annealing.

5.1 Annealing

To know more details about this method, it's a better way to know how it gets its name. Annealing is a heat treatment process to change the property of alloy. Generally, it can

enhance the ductility and reduce the hardness of the alloy. This process involves 3 phases: heat the alloy over recrystallization temperature, maintaining a suitable temperature for an appropriate amount of time and then cooling.
The atoms in the alloys stays at the position which keeps the internal energy at the local minimum. Increasing the temperature provides these atoms with the energy to randomly move from their own position to a new position, which might bring the internal energy to the global minimum.

5.2 Principle of simulated annealing

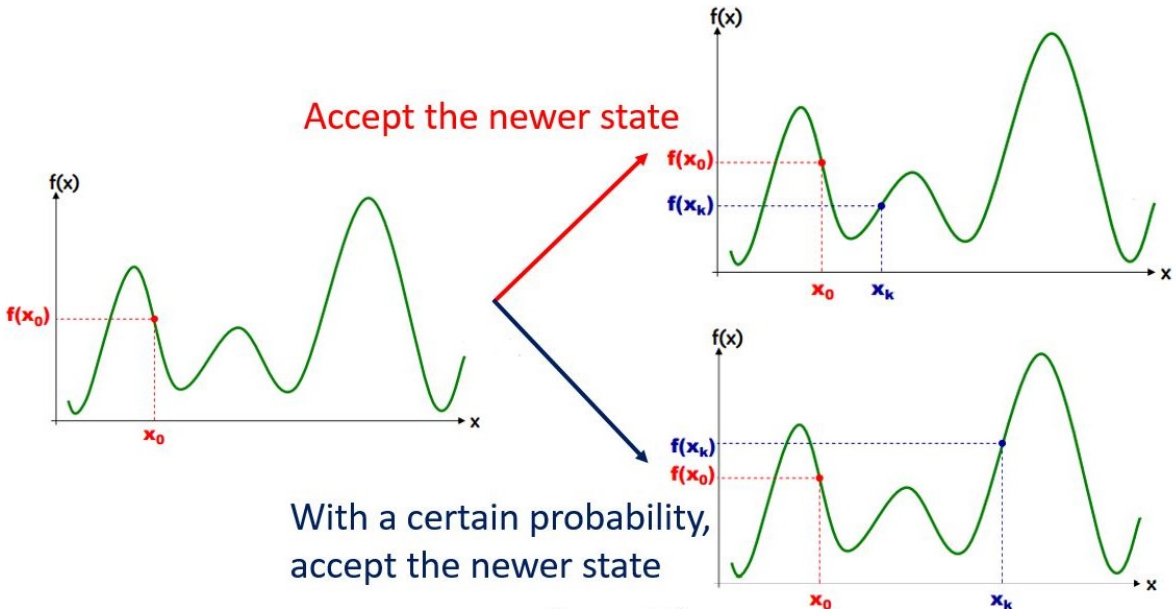


Figure 9: The principle of simulated annealing

The principle behind simulated annealing is similar to annealing. Take figure 9 as an example. The algorithm starts from one point in the solution space and calculate its corresponding function value with the given object function. Then we randomly change the state, that is, select other point and calculate its new function value. If the newer value is lower than previous value, like the red arrow in figure 9, we directly accept the new state. On the other hand, if the newer value is higher than the previous value, like in the blue arrow in figure 9, we don't update the value. However, by chance we still accept this change even the newer value is higher. The acceptance probability is given by

$$AP = min(1, e^{\frac{-\Delta f}{T}}) \tag{6}$$

while Δf is the difference of the energy, and T is the temperature.
Why do we accept the change even it is worse than previous trial? Because if we just give up all the worse result, we will consequently give up all of the possibility that can bring us to the global minimum.
As the temperature T will decrease as the iteration goes by, the acceptance probability will

decrease as well. This formula results in a higher acceptance probability at the beginning, explores more possibility for the global minimum. At the late phase of the iteration, as the acceptance probability decrease, the algorithm gradually focuses on the minimum which it is located.

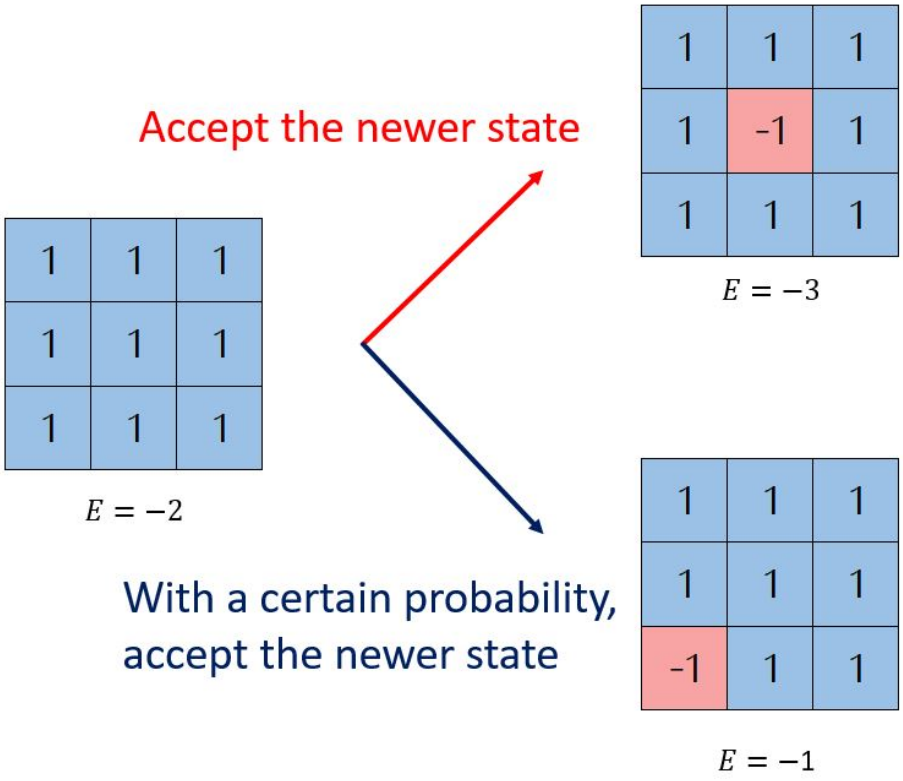


Figure 10: How does simulated annealing work to find the minimum configuration of spins

Figure 10 shows how simulated annealing is applied in our case. If the value of the newer state is lower, we just accept it. However, if the value of the newer state is higher than the previous value, we calculate the acceptance probability and decide whether we update the value.

5.3 Performance

In figure 11 we can see the difference between the performance of the brute force and simulated annealing. Note that y-axis is expressed in logarithm form. The curve of the brute force method is a straight line in this figure, which represents the time complexity of it is exponential.
The advantage of simulated annealing gets more and more obvious as the number of the spins increases. With the number of spins being 25, brute force took around 15 minutes, while simulated annealing requires just 17 seconds.

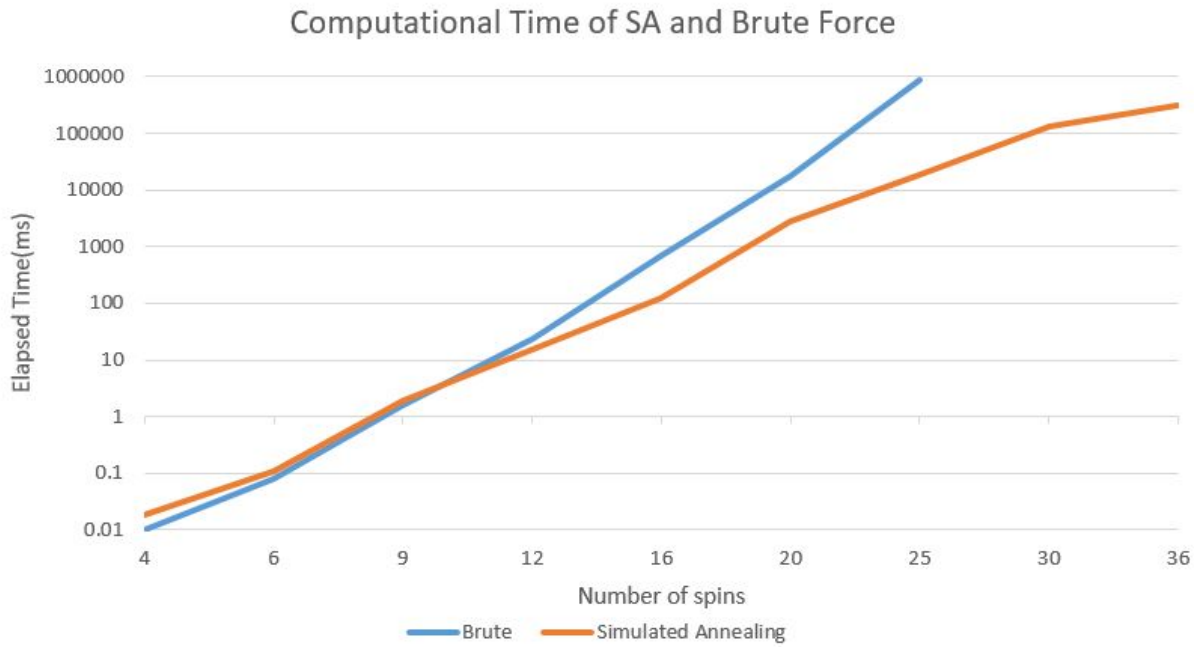


Figure 11: Performance between brute force and simulated annealing

5.4 Limitation

As simulated annealing is a probability method, it can not always guarantees the global optimum.

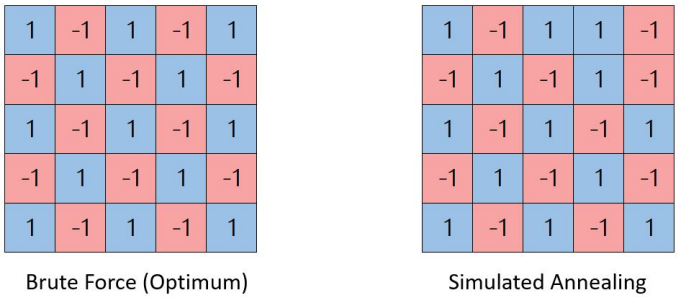


Figure 12: Simulated annealing can not always produce the global optimum

In figure 12, left side is the result of brute force method, which produces the global minimum result. The right side, which is the result of simulated annealing, however, produces the local minimum result. On the top right corner the reader can observe the difference. Nevertheless, the result is still tolerable given that the huge improvement of the computational time.

6 Method 4: Quantum Annealing

There is one last method we used to solve our problem of finding a minimum energy configuration. This method is quite distinct from the ones we have already seen. This is

because it does not even use a conventional computer to get to the desired result but uses quantum computer instead.
A quantum computer is different from a normal computer in that it uses so calles qubits instead of the bits we normally use. Qubits can take binary values $|0\rangle$ or $|1\rangle$ but they can also be in a superposition between these two

$$|\phi\rangle = \alpha|0\rangle + \beta|1\rangle. \tag{7}$$

with $\alpha, \beta \in \mathbb{C}$ and $|\alpha|^2 + |\beta|^2 = 1$.
This means, that it is not clear in which state the qubit will be until it is measured. If the qubit is in the state (7) then the probability of it being in state $|0\rangle$ is $|\alpha|^2$ and the probability of it being in state $|1\rangle$ is $|\beta|^2$.
Using this property a method called *quantum annealing* has been developed. With this method it is possible to find $\sigma \in \{\pm 1\}^N$ such that the energy

$$E(\sigma) = \sum_{i=1}^{N} h_i \sigma_i + \frac{1}{2} \sum_{i,j=1}^{N} J_{i,j} \sigma_i \sigma_j \tag{8}$$

is minimal. This is the so called *Ising model*.
If we set $h_i = 0$ for all $i \in \{1, \dots, N\}$ we recognize this to be exactly the problem we were dealing with before. Therefore quantum annealing may be a promising approach to solve the energy minimization problem from before.

6.1 Idea behind quantum annealing

Given the minimization problem (8) the quantum annealing method starts with N qubits which are all in the same superposition state with equal weights α, β. Then the energy landscape of the system gets transformed in such a way that at the end of the procedure the most likely resulting configuration of the qubits when measured is exactly the one which minimizes the energy of the Ising model (8).
To do this there are two modifications to the energy landscape we can do during the procedure. Namely adding biases to single qubits and coupling two qubits. A bias on a qubit makes it more likely for one state of the qubit to get observed over the other when measured. This corresponds to the weights h_i in the Ising model (8).
A coupling between two qubits makes it more likely for them to either show up in the same quantum state or to show up in the exact opposite state when measured. This corresponds to the weights $J_{i,j}$ in the energy formula (8), because as we have seen before (Section 2.2) if $J_{i,j} > 0$ then σ_i, σ_j want to be of opposite sign, whereas for $J_{i,j} < 0$ they want to be of the same sign.
So by introducing biases and couplings that correspond to h_i and $J_{i,j}$ we can assure that the most likely outcome when measured will be the minimum energy configuration.
However, we have to keep in mind that this method is still probabilistic. The measured result will not always be the minimum energy configuration. Therefore we have to run multiple trials to be confident we have found the optimal solution.

6.2 Quantum annealing in practice

We used the resources of the company D-Wave Systems to solve our problem in practice because D-Wave provides remote public access to their quantum computer. They have

an easy to use IDE which can be used inside the browser. To use the quantum annealing method to for the Ising model (8) a function for python is provided which takes a vector $h \in \mathbb{R}^N$ and a matrix $J \in \mathbb{R}^{N \times N}$ as well as the number of trials we want to do as input.

This means that all we had to do was to translate our function $J(r)$ which we used before into a matrix $J \in \mathbb{R}^{N \times N}$. We did this by setting the values of J to

$$\begin{aligned} J_{i,j} &:= J(d(i,j)) \\ \text{with} \quad d(i,j) &:= \sqrt{(i_x - j_x)^2 + (i_y - j_y)^2} \\ \text{where} \quad i_x &:= i \mod m \\ j_x &= j \mod m \\ i_y &= \left\lfloor \frac{i}{m} \right\rfloor \\ j_y &= \left\lfloor \frac{j}{m} \right\rfloor \end{aligned} \tag{9}$$

So we are using the euclidean distance d between two points i, j in a $n \times m$ grid. The grid points i, j are represented as a point in $\{1, \ldots, N\}$ for $N = n \cdot m$. We are reducing our two dimensional problem to just one dimension for it to be easier to work with. For this case, the sample output is formulated as in equation (10) and a number of 150 trials is displayed in figure 13.

$$J(r) := \frac{1}{r^3}, \; n = 3, \; m = 5 \tag{10}$$

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	energy	num_oc.
-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	-8.079623	17
+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-8.079623	53
+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-8.079623	1
+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	-1	-6.534274	2
-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	+1	-6.534274	1
+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	-1	-1	+1	-6.534274	1
+1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	+1	-1	-6.534274	2

Figure 13: Representation of 150 trials

In figure 13 the rows from left to right represent the 15 specific spins, the corresponding energy and the number of times this configuration was measured.
Configurations with a resulting energy of -8.08 were observed 71 times, so about halt of the time. We can check by brute force (Chapter 3) that this is really the minimum energy. This works here because $N = 15$ is still relatively small but for bigger N the runtime becomes so big that checking that would no longer be feasible. More on that in the next section.
Another thing we can notice is that the two states that got measured most often are exact opposites of one another, meaning that all the spins are flipped between the two configurations. For the energy functions we are considering, this will always be the case

because we are disregarding the linear terms in the Ising model (8) and for the quadratic terms it holds that

$$J_{i,j}\sigma_i\sigma_j = J_{i,j}(-\sigma_i)(-\sigma_j). \tag{11}$$

Thus, two configurations with flipped spins are equivalent in our model.

6.3 Runtime

The big advantage of this method, which justifies the struggle of using a quantum computer to solve the problem, is the runtime. If the trends on figure 14 in the last chapter continue we can expect a runtime of billions of years for the brute force method. For the simulated annealing method we can expect a runtime of more than an hour. But for this new method the runtime is just over 100ms which is a huge gain in performance.

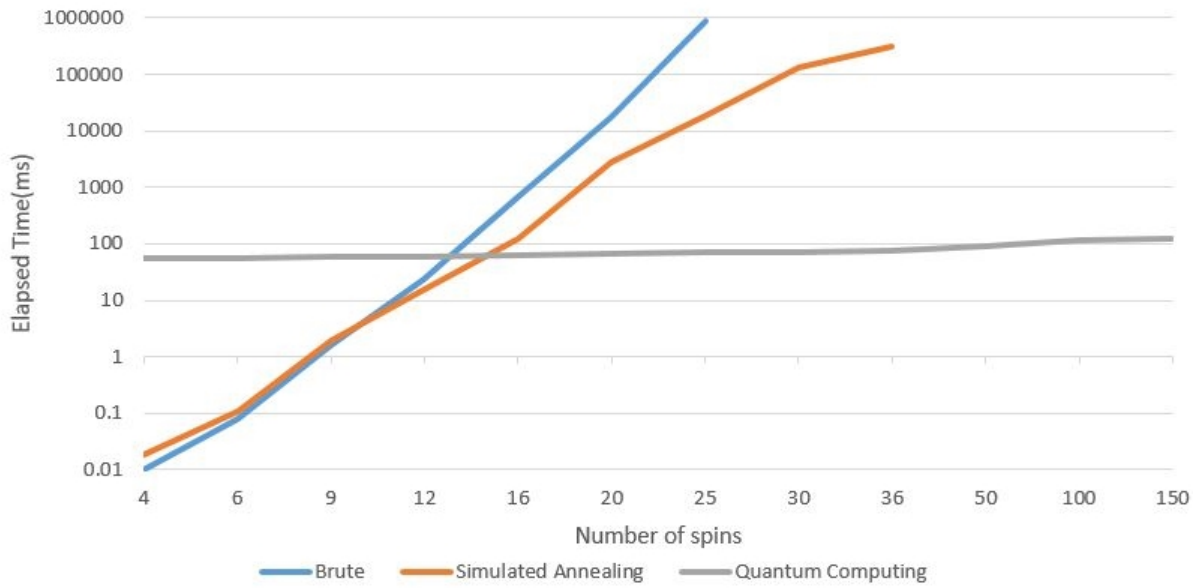


Figure 14: Computational Time of SA, Brute Force and Quantum Computing

Of course the simulated annealing method was limited to our hardware but it is assumed that quantum annealing has a slight edge over simulated annealing[2]

[2] [Ni]

7 Conclusion

In this paper we were looking for efficient ways of finding the interior structure of materials in which interior points are interacting with each other pairwise and in which these points can be in two opposite states that decide if the interaction with other points will be positive or negative. The state which will naturally occur will be the one that minimizes the interior energy. Therefore the aim of our research was to find efficient methods and algorithms to find the minimum energy configuration of a material, based on how its interior points are interacting with each other.
One of the methods that was followed in order to solve the initial problem was the implementation of a feedforward neural network. The model achieved a quite high performance of 85 % despite the main limitation of resources. Hence, a notable improvement in that model would come from a significant speedup of the Brute Force algorithm, which would lead to an increased number of inputs for the network and subsequently to a more generalized neural network.
The method whose runtime stood out the most was quantum annealing. Here we used an actual quantum computer instead of a normal computer to solve our problem. This method was also a probabilistic approach, similar to simulated annealing, the one we looked at before, but it outperformed it significantly in runtime.
However, given that quantum computer is not that easily accessible, simulated annealing is the more practical method. There are a lot of parameter and configuration for simulated annealing to perform better, such as the temperature scheduler, the stopping energy and the max iteration number. To get a better result with higher accuracy and the shorter computational time, we still need some time to tune those parameter. It could be a potential topic to optimize it.

8 Bibliography

References

[Wi] https://en.wikipedia.org/wiki/Shape-memorycoupling

[Ni] Nishimori, H. (5 February 2015). Comparison of quantum annealing and simulated annealing. Department of Physics, Tokyo Institute of Technology Oh-okayama, Meguro-ku, Tokyo 152-8551, Japan.