Algorithms in Structural Bioinformatics

Homework #1

Psallidas Kyriakos
7115152200033

All the results and graphs can be reproduced by running the python notebook included in the zip file

Problem 1

Problem statement: Find all optimal secondary structures of the RNA sequence

$$AAUACUCCGUUGCAGCAU$$

with the following crude energy minimization algorithm. Starting from the slides' algorithm, use the following initialisation: $j + 5 > i \Rightarrow E(i, j) = 100, i > j$ and bond energy $b(i, j) = 4, 0, 4$ for Watson-Crick bonds, GU, and all other possible pairs respectively. Implement your algorithm in Matlab, R, Python or other convenient system; submit your code. Print the filled-in table E. Draw (by hand) all optimal folds, show its bonds, and the corresponding backtrack path.

Solution: we first define a function called `bond_energy` which returns one of the appropriate bond scores mentioned before based on the bases of the bond:

- Watson-Crick bonds have energy = -4
- Unfavorable bonds have energy = 4
- GU bonds have energy = 0

Having created a function to assign energies when a bond is created, we create a second function, which takes as input an RNA sequence and outputs the crude energy minimization table E, along with the backtrack matrix.

1. First, beginning from the last row and the first column of the matrix, it places 100 following the rule $j + 5 > i \Rightarrow E(i, j) = 100$
2. Then for each remaining position it fills it with the minimum value of:

$$E(i,j) = min \begin{cases} E(i-1m, j), \\ E(i, j+1), \\ E(i-1, j+1) + bondenergy(r_i, r_j), \\ min_k\{E(i,k) + E(k-1, j) : j+1 < k < 1\} \end{cases}$$

3. The back-track matrix is filled based on the index of the minimum value in the cases above, if the minimum exists multiple times in the cases above we fill the backtrack table entry with all possible backtrack paths.

Problem 1 continued on next page. . .

The crude energy minimization table E produced after the execution of the functions is illustrated in (figure 1). Its corresponding backtrack table is at (figure 2). Finally, following the optimal backtrack paths from the top right entry of the backtrack matrix two distinct optimal folds are discovered and showcased in (figure 3), a photo design software was utilized rather than pen and paper (as requested) for the shake of presentational clarity.

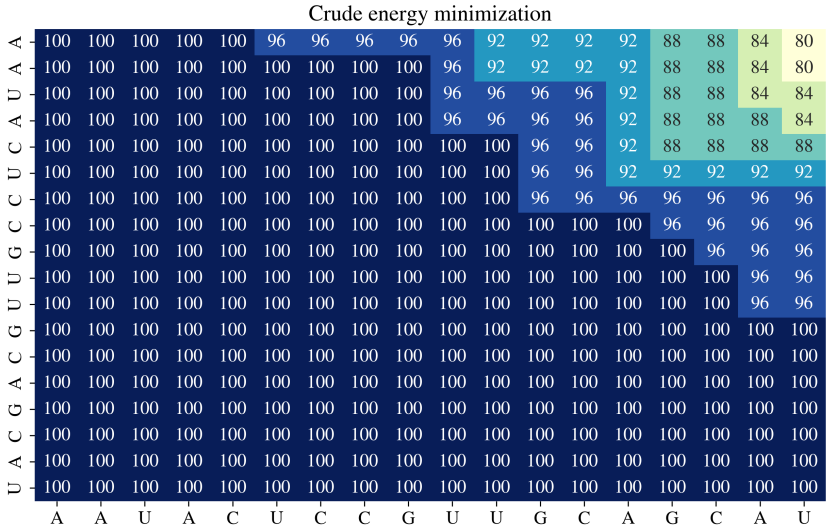

Figure 1: Crude energy minimization table E for sequence: AAUACUCCGUUGCAGCAU

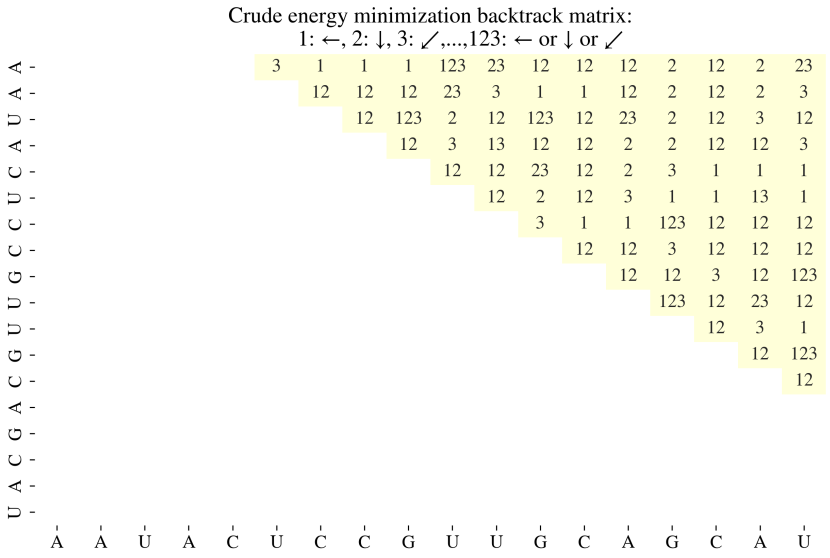

Figure 2: Backtrack table for sequence: AAUACUCCGUUGCAGCAU

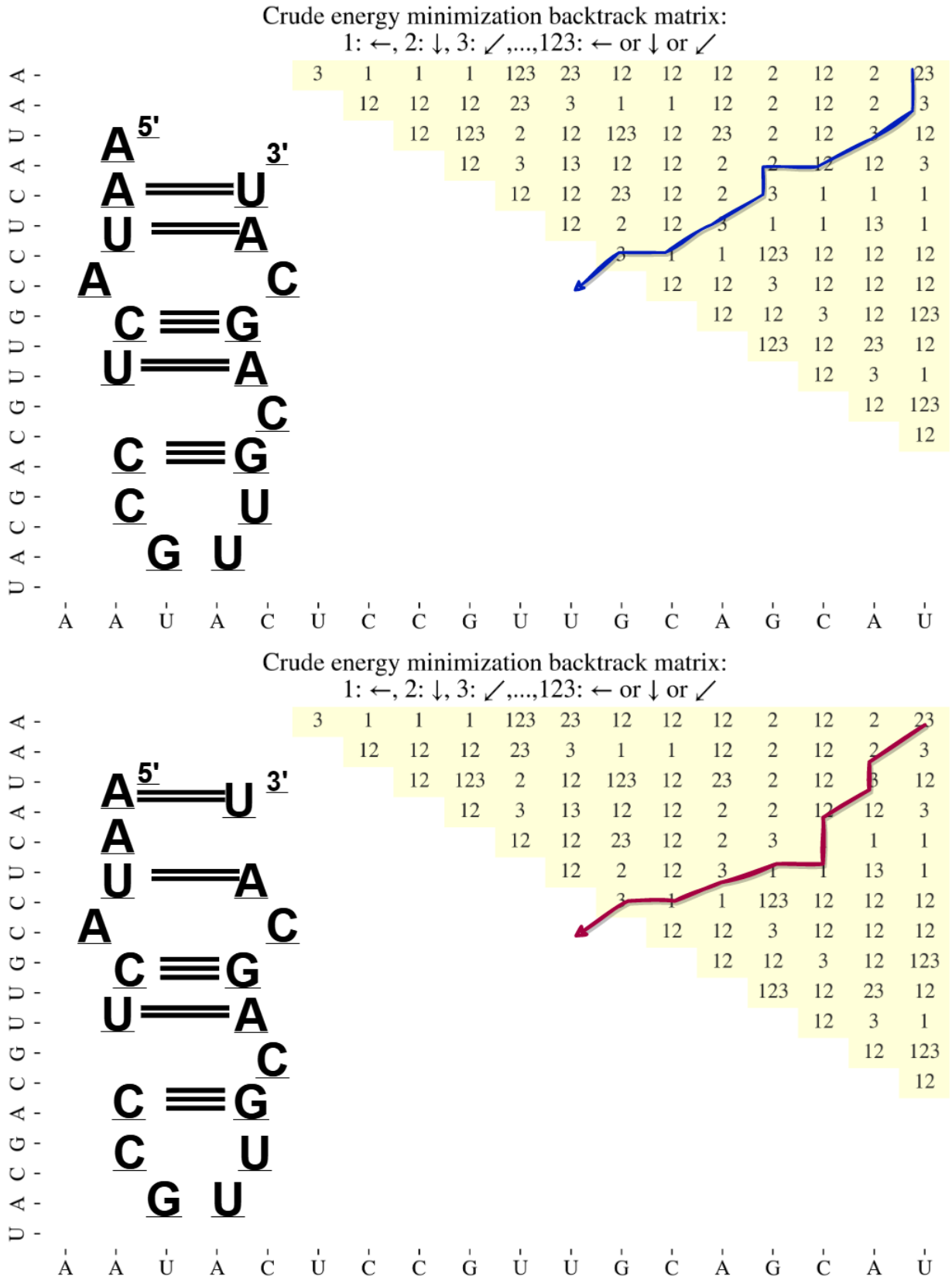

Figure 3: Optimal folds with their backtrack path

Problem 2

Problem statement: Given are 10 conformations of a specific molecule in file "10_conformations.txt" with $n = 369$ atoms on the backbone (hence in correspondence). The table starts with 2 lines containing 10 and $n = 369$. The rest of the tale uses tabs to deine 3 columns containnig n triplets $x\ y\ z$ of each conformation hence $2 + 10n = 3692$ rows. Implement c-RMSD and d-RMSD in Matlab, Mathematica, Maple or other system offering linear algebra (SVD); submit your code. If your system provides either of these functions, it is OK to just use it.

1. Compute the c-RMSD distances between all $\binom{10}{2}$ pairs of conformations. Compute the pairs of conformations. Use them to find the L1-centroid. conformatio.

2. Repeat (1) for d-RMSD using all $k = \binom{n}{2}$ distances within each conformation, or a random subset of $k = 3n$ distances.

3. Do all 3 approaches yield the same centroid? How do they compare in terms of speed.

Solution: We load the x, y, z coordinates from the file and store them in a data-frame. Since this dataframe contains 10 conformations of the molecule we reshape it into a an array of 10 conformations with 369 atoms each and their respective x, y, z coordinates. c-RMSD is given by the following formula:

$$c - RMSD = \sqrt{\frac{1}{n}\sum_{i=1}^{n} ||x_i - y_i||^2}$$

where x_i and y_i are the i-th carbon atom coordinates for the two conformations we are comparing. In order to be able to compare the conformations based on point distance in the 3D space of the carbon atoms we need to superimpose them. This requires that we translate and rotate the conformations so that they are as similar as possible, thus having the lowest possible c-RMSD. Optimal translation can be decoupled from rotation optimization, so we will first find the optimal translation and then the optimal rotation.

Optimal Translation

First of all we need to translate the coordinates of all the conformations to the origin. This can be done by subtracting the mean of the coordinates from each coordinate:

$$x = x - \bar{x},\ y = y - \bar{y},\ z = z - \bar{z}$$

Optimal Rotation

To find the optimal rotation matrix we first find the covariance matrix Σ of the two conformations. Since both conformations have a mean of $[0, 0, 0]$ This is done by multiplying the transpose of the first conformation matrix X^T with the second conformation Y and dividing with the number of atoms. Since we have three dimensions we get a $3x3$ covariance matrix Σ:

$$\Sigma = \frac{1}{n} X^T Y$$

Using Singular Value Decomposition (SVD) we can decompose the covariance matrix into:

$$\Sigma = USV^T$$

- where U and V are orthogonal matrices containing the left and right singular vectors of Σ respectively
- S is a diagonal matrix containing the singular values of Σ on the diagonal in descending order
An optimal rotation matrix Q should satisfy:

$$V^T Q^T U = I$$

Thus $Q = VU^T$ is the optimal rotation matrix.
In case $det(Q) = -1$, that means that the rotation matrix is a reflection, we need to correct this by multiplying the last column of U with -1 to get the optimal rotation matrix Q.

c-RMSD Results

We finally calculate the c-RMSD distances between all $\binom{10}{2}$ pairs of conformations and store them in a matrix, which we then pass in a dataframe and visualize them in a heatmap. Of course we dont compute the c-RMSD distance between a conformation and itself, as well as repeating pairs of conformations. Thus we get an upper triangular matrix with the c-RMSD distances between all $\binom{10}{2}$ pairs, the results are illustrated in (figure 4).

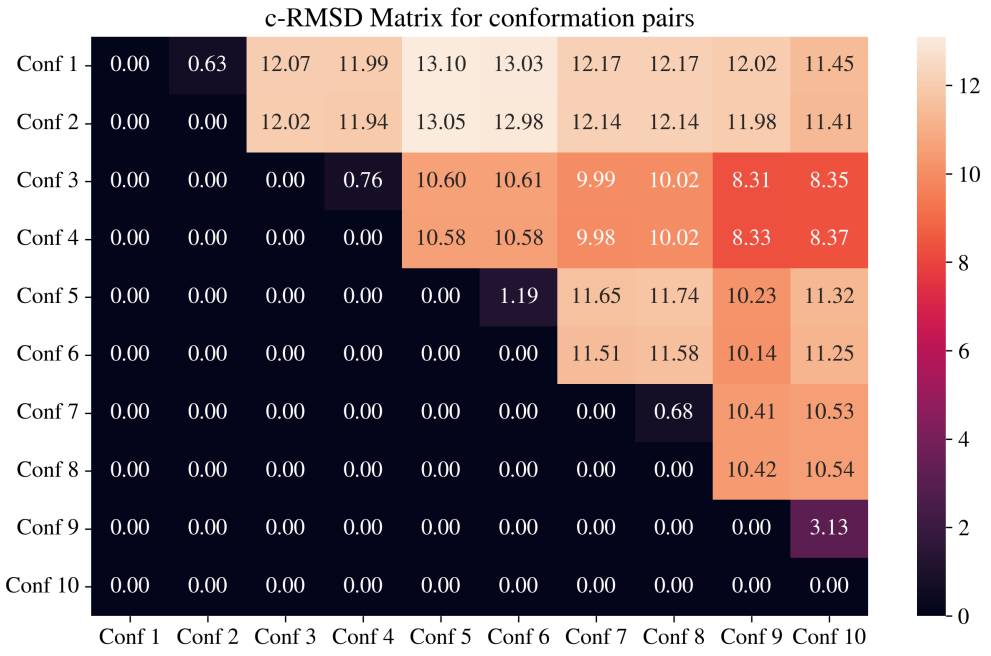

Figure 4: c-RMSD values between non-repeating conformation pairs

To find the L-1 centroid conformation, we find the conformation that has the lowest mean and median sum of distances from all other 9 confirmations. That is conformation 4 (counting from 1 not 0), with $mean = 9.17088$ and meadian $meadian = 10.015295$.

d-RMSD calculation

Rather the than coordinates of the carbon atoms we will now use the distances between the carbon atoms in each conformation under the assumption that if two molecules are similar the distances between their atoms will be similar as well. Thus we calculate all $\binom{369}{2}$ distances between all pairs of atoms in conformations and store them in a matrix. Again much like in the c-RMSD case, we don't compute the distance between an atom and itself, as well as repeating pairs of atoms since it does not scale well. We then calculate the d-RMSD for non-repeating conformation pairs following the equation:

$$d - RMSD = \sqrt{\frac{1}{k}\sum_{i=1}^{k} ||d_i - d_i`||^2}$$

where d_i and $d_i`$ are the i-th indexed atom distance in the distance matrix for each of the two conformations we are comparing. The results are presented in (figure 5). With the same

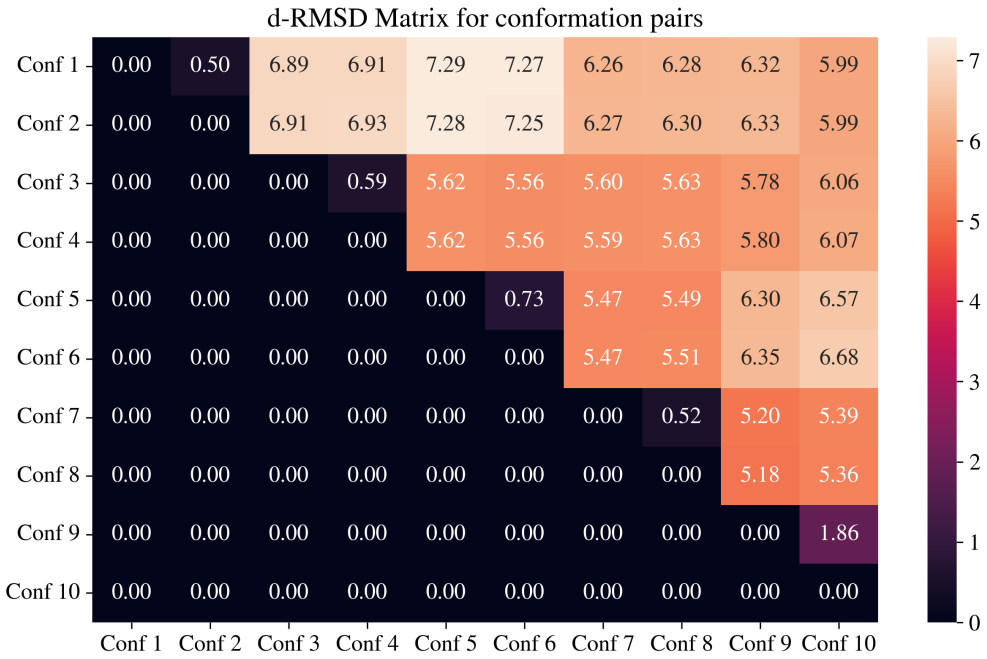

Figure 5: d-RMSD values between non-repeating conformation pairs

methodology as in the c-RMSD case, we find that for d-RMSD the L1-centroid is conformation 7 with $mean = 5.085135$ and $median = 5.470709$.

Comparing the speed and L1-centroids of c-RMSD and d-RMSD

In terms of speed as its evident below, c-RMSD is much faster since we need to compute $\binom{n}{2}$ distances, in this case $n = 369$ thus 45 distances. On the other hand in dRMSD we had to calculate $k = \binom{369}{2}$ thus $k = 67896$ distances.

- The speed of c-RMSD calculation for the 10 conformations is: 0.04 seconds
- The speed of d-RMSD calculation for the 10 conformations is: 30.67 seconds

Regarding the L1 centroids, they are different for c-RMSD and d-RMSD, this is to be expected since they represent different measures. C-RMSD measures similarity based on the center of mass between superimposed conformation and thus individual coordinate differences do not have a significant effect on the result if the mean is similar for both conformations. On the other hand, d-RMSD represents the similarity of inter conformation distances between atoms and thus is more affected by minor differences.

Problem 3

Problem statements: Consider 50 Ca atoms indexed $A102$ to $A152$ of main protease of SARS-COV-2 in complex with a peptide-like inhibitor (PDB id: 6LU7). Construct the 51 by 51 Cayley-Menger matrix B.

1. Compute $rank(B)$, explain why the obtained value is correct.
2. Perturb entries of B by 5%, an (maintaining symmetry, positive entries, 0's, 1's) then explain the new value of $rank(B)$. Compute Gram matrix G, apply SVD: $G = U\Sigma V^T$. Now S is the diagonal matrix containing the 3 largest singular values of G. Get the 3D coordinates as $\sqrt{S}U^T$, and report the c-RMSD against the original structure.

Solution: First of all, we download and open the relevant PDB file and keep the rows containing backbone atoms Ca,The 6th column contains the residue sequence number while columns 7-9 contain the x, y, z coordinates of the atoms. Thus we use the residue atom index to extract the coordinates of the 50 Ca atoms indexed 102-151 and store them in a data frame. The Cayley-Menger matrix is defined as the distance matrix of the atoms $M_{ij} = \frac{1}{2}dist(p_i, p_j)^2$ augmented with a row and column of ones with a $[0, 0]$ entry of 0. We calculate the Cayley-Menger matrix (figure 6) of the 50 Ca atoms and find that its $rank = 5$, this makes sense since the CM matrix rank is the rank of Matrix M plus 2. The rank of M is 3, since it spans the 3-D space.

To perturb entries of the CM matrix B by 5%, we can create a 50 by 50 matrix which contains values -1 and 1 randomly placed which then we multiplie with 0.05, this results in a matrix with elements -0.05 and 0.05, with elementwise multiplication of this matrix with the original M matrix, we can create a matrix containing either 5% or -5% os the values of the M matrix. We then add the

values of the noise matrix to the non-augmented columns of the Cayley Menger matrix, the resulting perturbed Cayley Menger matrix has a rank of 51, thus full ran. This is because the 5% pertrubation of the noise disturbed the linearity of the system and resulted in a non linear combination of all the rows or columns.

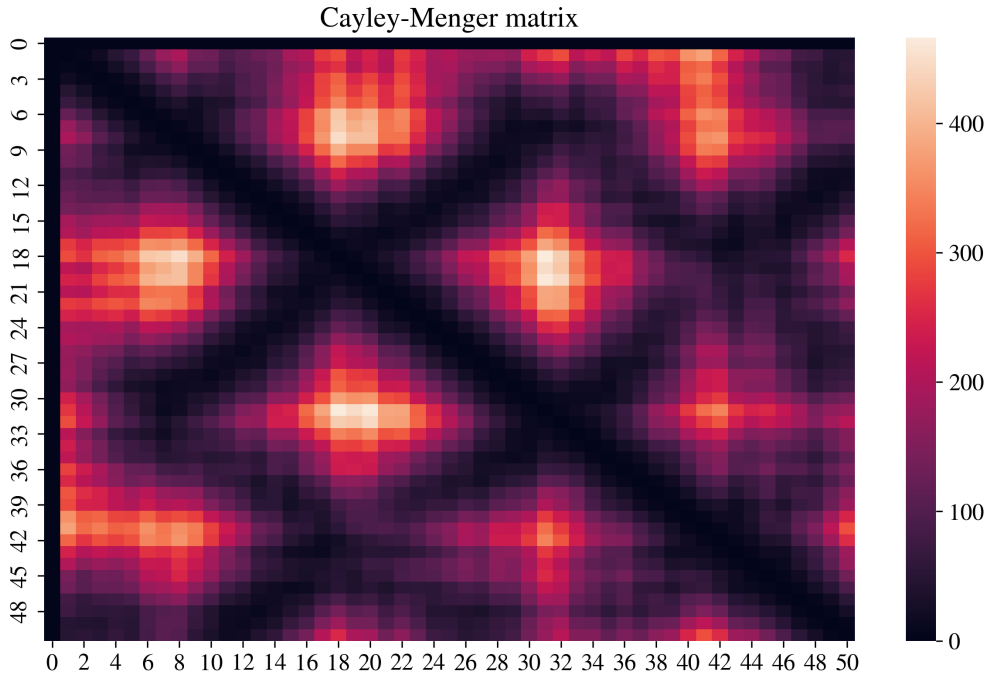

Figure 6: Cayley Menger matrix for the 50 Ca atoms indexed $A102$ to $A152$

To create the gram matrix of M We utilize the equation for each of the entries of a gram matrix based on the entries of the distance matrix M:

$$G_{ij} = \frac{d_{i0}^2 - d_{ij}^2 + d_{j0}^2}{2}$$

Utilising SVD, we decompose G into USV^T, then we pick the largest three singular values of S and create a 3 by 3 diagonal matrix. The corresponding left singular vectors of G are picked via the first 3 columns of the U matrix. Thus we retrieve the 3-D coordinates as:

$$M' = \sqrt{S}U^T$$

Finally we utilize the method explained in the second problem to find the c-RMSD of M against M' = 232.904.