

Polymer micelle farm

W. Nowicki

The project concerns the aggregation of amphiphilic/amphipathic polymer chains, namely the block copolymers containing the lyophilic and lyophobic fragments, into polymeric micelles. Realization of the project applies the Metropolis Monte Carlo (MC) method on the 3D cubic lattice implemented in the attached Pascal program **metropolisPC.pas** (similar to that from Exercise “The effect of solvent on conformation of a linear macromolecule. Metropolis-MC method”) and accompanied with the dynamic link library **AmphipaticEq.dll**. The choice of the method is dictated by an extremely long relaxation time of micelle formation and equilibration. Global modifications of chains allowed in the MC simulations are much more effective than the standard Molecular Dynamics methods.

Analytical approach

Let's consider a solution of linear amphipathic chains of N segments in the bulk (3D). The chains are di-block copolymers containing two types of segments – lyophobic and lyophilic – characterized by different Flory interaction coefficients χ . In the solution, these blocks form coils of radii of gyration scaling with the number of beads with different Flory exponents μ for lyophobic and ν for lyophilic parts of the chain. When the polymer concentration is high enough, the chains may aggregate into micelles with the core formed by lyophobic sub-chains and the shell formed by lyophilic heads. As a consequence of the difference in the scaling of dimensions of both sub-coils, the core of the micelle is denser than its shell.

For simplicity, let's assume that micelles are spherical and that the coils formed by lyophobic and lyophilic parts of chains are relaxed and not disturbed by each other. The sub-chains forming the shell of the micelle should cover completely the core but, at the same time, their number is limited by the available space. Therefore, for the micelle in the equilibrium state, when all parts of chains are relaxed, the number of heads n (equivalent to the number of chains) is fully determined by geometrical restrictions. The volume of the core of the micelle is equal to the sum of volumes occupied by all lyophobic sub-chains forming the dense core

$$V_{\text{core}} = n \left(\frac{b}{\sqrt{6}} ((1-\alpha)N)^\mu \right)^3 \quad (1)$$

where α is the ratio of the number of lyophilic beads to the total number of beads in a single chain. Let's assume for simplicity that the shell is filled with lyophilic heads in such a way that the surface area S_0 of the sphere located between the surface of the core and the whole micelle is fully covered with sections of heads, i.e. curved polygons of the area equal to circles of the radius equal to the radius of gyration of lyophilic sub-chain. The radius R_0 of the sphere is equal to the average of the radius of the core and the whole micelle, as shown in Fig. 1. Thus, the geometrical constraint of the shell requires that

$$\frac{n \left(\frac{b}{\sqrt{6}} (\alpha N)^\nu \right)^2}{S_0} = 1 \quad (2)$$

On the basis of the above assumptions, the aggregation number of the micelle can be calculated from the equation:

$$n = \frac{9\pi}{16} \left(\frac{((1-\alpha)N)^\mu + (((1-\alpha)N)^{3\mu} + (\alpha N)^{3\nu})^{1/3}}{(\alpha N)^\nu} \right)^6 \quad (3)$$

The dimensions of the polymer micelle are shown in equations below: for the core radius

$$\frac{R_{\text{core}}}{b} = \frac{1}{2} 6^{-\frac{1}{6}} \pi^{-\frac{1}{3}} n^{1/3} ((1-\alpha)N)^\mu \quad (4)$$

and the radius of the whole micelle:

$$\frac{R_{\text{micelle}}}{b} = \left(\frac{3}{4\pi} \right)^{1/3} \frac{1}{\sqrt{6}} n^{1/3} \left(((1-\alpha)N)^{3\mu} + (\alpha N)^{3\nu} \right)^{1/3} \quad (5)$$

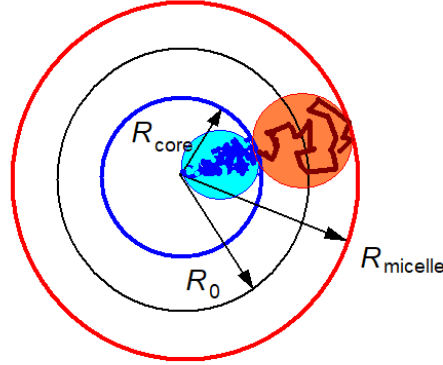


Fig. XX. Schematic structure of polymeric micelle.

Values of exponents μ and ν are fully determined by the Flory interaction coefficient:

$$\chi = \frac{Z \left(\epsilon_{\text{PS}} - \frac{\epsilon_{\text{SS}} + \epsilon_{\text{PP}}}{2} \right)}{kT} \quad (6)$$

where ϵ_{PS} , ϵ_{SS} , and ϵ_{PP} are the energies of interaction between the polymer segment and the solvent molecule, between two solvent molecules and between two polymer segments, respectively, k – the Boltzmann constant, T – temperature, Z – the effective coordination number of the lattice (for the 3D cubic lattice $Z=4$). Assuming that the solvent of the solution studied is water, the ϵ_{SS} energy is equal to the energy of the hydrogen bond (about $-10 kT$). Assuming additionally that the lyophilic parts of a chain can be treated as immersed in the athermal solution (in a good solvent), the Flory coefficient takes values 0 and 20 for lyophilic and lyophobic parts of the chain which results in $\mu=1/3$ and $\nu=3/5$.

Assuming additionally that $\alpha=0.5$, the aggregation number of the polymer micelle should be approximately equal to 12.

Program features

At the very beginning of the program **metropolisPC** the procedure **start** is called. The procedure looks for the file **_init.dat** which contains parameters of simulation. If the file is absent in the current directory, **start** invokes **screeninit()** (**MMC.inc**) and asks for the initial data in the program window. The initial parameters are:

- the number of molecules (1-1000),
- the number of steps (for example $1e8$ for a single simulation),
- the Monte Carlo step number at which the current simulation results are written to files (usually assumed as equal to 1). One Monte Carlo step means the number of elementary steps equal to the product of the number of molecules and the number of segments
- the number of simulations (it seems that 1 can be enough)
- the number of segments
- the number of segments in the lyophilic fragment of the chain
- the energy of interaction between solvent molecules (kT) (a negative value)
- the energy of interaction between shell segments (kT) (equal to 0 for athermal solution)
- a half-length of simulation box edge (usually 10 - 100)

- the reduced temperature (for typical simulation equal to 1)
- the character (switch) which indicates whether the simulation starts (S) or to be continued (C). Initially, the switch should be set to S. When a set of results is already written, the switch should be changed to C. Now the program can be stopped and run many times. Each new program instance will continue with a previously generated and equilibrated structure. In the end, **start** reads the **_strategy.dat** file which defines ratios of probabilities of different move algorithms applied by the program.

The main body of the program contains the two main of iterations located one in the other. The first one (with the control variable **jj=1**) starts the new simulation by initialization of the free space and location of all macromolecules in it. These actions are performed by **initSys** procedure in **MC.inc**. The procedure fills the whole lattice with zeros and then generate **surfN** macromolecules of **N** segments with **headN** lyophobic segments employing the SAW algorithm (**_SAW** from **AmphipaticEq.dll**). All generated segments are represented in the lattice array **L** as numbers 1 or 2. Additionally, each segment coordinates are stored in the **C** array.

After some technical calls of functions/procedures (switching the graphics window, reading data needed to continue interrupted simulation, etc.), another iteration starts with the control variable **ii**. The iteration is the main part of the program – it performs the thermodynamic equilibration of the system. Namely, it changes the conformations and positions of molecules in the system striving to the minimum energy of the whole system. The iteration starts with the random choice of the number of the segment and the number of the molecule to be moved (the number of the segment is sometimes ignored by some global move algorithms). The current location of all segment positions of the chosen chain is saved in the additional array by **save** procedure and the total interaction energy of the chain is stored in variable **EnergyT1** (calculated by **energy** function in **MC.inc** file). Then the **_move** function (from **AmphipaticEq.dll** tries to change the conformation of the system using some algorithms described below. If it can do it successfully (i.e. without violation of the excluded volume; if nevertheless it happens, the variable **moved** takes value different from zero) the energy of the new conformation is calculated and stored in variable **EnergyT2** (also calculated by **energy** function in **MC.inc** file). On the basis of the initial and final conformation energies, the Metropolis algorithm decides whether the new conformation is accepted or not. If not, the previous conformation is restored by **restore** procedure. If the Metropolis algorithm accepts the conformation, the **ii** variable is increased by the number of modified segment positions and the simulation continues. When **ii** value is equivalent to the next MC step, the current results are written to several files described below. Additionally, the new line containing information about the system energy and the biggest micelle found in the simulation box is added to the file **results.txt**. Then the iteration loop realizes some actions important for data service but not significant from the physical point of view.

The operation of two procedures called by the inner loop described above is worth analyzing.

The first procedure (which are stored in **AmphipaticEq.dll** but which can be replaced directly by any combination of move procedures by the student) is the function **_move**. On the basis of the contents of the **_strategy.dat** file which assumes probabilities of each set of modifications (1 – 8), it realizes the chosen modification using a set of algorithms:

- (1) the set of the local or bilocal Verdier-Stockmaer elementary algorithms: **_End_Move**, **_Reptation**, **_Kink_Jump**, and **_Crankshaft**
- (4, 5) the global algorithms: **_Cut_and_Paste** (including Inversion and Reflection) and **_Pivot**
- (2, 3) the neutral position change algorithms: **_Shift** (the shift of the whole macromolecule along with one of the system axes), **_Rotate** (the rotation of the whole macromolecule by the angle of $\pi/2$, π , $3\pi/2$ around the randomly chosen axis of the system), **_Inverse** (the exchange in positions of lyophobic and lyophilic segments)
- (6, 7, 8) the “initial” algorithms which make it easier to start the micelle creation: **_RotateTowards0** (the rotation of the whole macromolecule in the way that the final position of the lyophobic end is closer to the center of the simulation box than the lyophilic one), **_InverseTowards0** (the exchange of positions of lyophobic and lyophilic segments for the same purpose) and **_ToCenter** (the shift of the whole macromolecule the center of the simulation box). It should be stressed here that these algorithms do not lead to the equilibrium conformations. The initial dense structure has to be relaxed in many steps to the situation when the number of adjoining and disconnecting macromolecules achieves the “kinetic” equilibrium.

The other important procedure is the procedure **key()** (**keypressed.inc** file). The procedure checks whether a key was pressed in the course of the simulation. If a key was pressed, the program takes action:

- x, y, z – the simple graphic window presents the projection of the contents of the simulation box along the assumed axis
- t – performs the test of consistency of the data

- **f** – finds coordinates of the biggest micelles in the simulation box (the Gaussian density of core segment distribution is assumed, the algorithm seeks for the densest core objects). The procedure collects coordinates of the micelles, their aggregation numbers (**R_micelle_data.txt**), the average conformational entropy and internal energy of macromolecules currently forming the micelle (**R_micelle_data_E.txt**), the radial segment distributions of 5 biggest micelles found in the simulation box (**Radial_LocDistrXXXX.txt**) and the geometrical properties of current biggest micelles (**R_micelle_data_Rg_3D.txt**). Columns in the file contain: the ordinal number of the micelle, its aggregation number, the radius of gyration of the micelle core, the radius of gyration of the whole micelle, the monomer density in the core and the monomer density in the shell of the micelle.
- **r** – calculates the average radial distribution of all species incorporated into the system (**Radial_GlobDistrXXXX.txt**)
- **s** – creates the Surface Evolver file which gives the possibility of spatial view of the system (**_SurEv**). The function uses the **Find** function.
- **d** – dumps all current system data to appropriate files: the contents of **L** and **C** files, the current number of iteration, the blueprint of the simulation, the copy of **_init.dat** file, arrays of segment coordinates and the lattice node content (**Coord_XXXX.txt** and **Lattice_XXXX.txt**)

The names of result files contain information about the simulation number **S=**, the micelle number **M=**, and optionally the iteration number **I=** (when called by **key** procedure)

All the procedures described above are realized automatically after each end of the current MC step.

It is also worth noting that the initial conformation of macromolecules can be changed by proper changes in parameters of **initSys** procedure. After modification, the program should be restarted.

Since some changes in the initial parameters can produce some errors in the whole simulation system, the **AmphipaticEq.dll** file offers the procedure **_Test** for verification of data consistency.

The *.fe file generated by the **_SurEv** function (**AmphipaticEq.dll** file) is the source file for Surface Evolver (SE). The interpretation of the file by SE produces a 3D view of the simulation box with periodic boundaries (**TORUS** model) and gives some text information about the number of macromolecules, the number of segments, the simulation box dimensions, the number of elementary steps, the aggregation number of the micelle and the coordinates of its mass center. Besides the modifications of graphic screen offered by SE, the view of the system can be modified by some redefined one letter commands:

r/c – the switch of the mode of the display from raw (original coordinates of segments) to clipped (the apparent positions of segments in the simulation box)

s/h – the switch of the mode of displaying the sphere (green) pointing the core of the biggest micelle found

b – cuts off all the macromolecules which do not belong to the found micelle (the command cannot be undone)

Z – converts the picture to the pearl-necklace view – it allows assessment of the occupancy of the space – it produces the most realistic picture of the micelle. It is recommended to use after **b** command (the command cannot be undone).

I – executes the cross-section of the simulation box. The sectional plane passes through the largest micelle.

All irreversible commands mentioned above can be canceled by the restart of SE.

Computations

The aim of the project is to grow a micellar system that is in the thermodynamic equilibrium, starting with the randomly positioned copolymer macromolecules located in the simulation box.

All initial data and strategy of equilibration gathered in **_init.dat** and **_strategy.dat** files can be changed in order to obtain the system of interest. However, it is necessary to take into account the limitations of the program (e.g. size of tables) and the computation time needed to implement the project. It is advisable to try different starting datasets in short simulations before deciding on the main calculations. The program can also be changed by i/ modifying the parameters of the **initSys** procedure, ii/ replacing it by directly calling the **_SAW** procedure, iii/ replacing the **_move** function by direct calls of the procedures that implement particular chain modifications, and so on. All such modifications should be controlled for data consistency by the procedure **_Test**.

The finally obtained system with the “bred micelle” should be visualized and analyzed using the values of the system or micelle energy, the aggregation number, the local and global radial distribution functions, etc.

The simulation results should be compared with the theoretical predictions mentioned in the Analytical Approach section and discussed in terms of both the correctness of the analytical model and the correctness and effectiveness of the simulation. How could the theoretical model be improved (e.g. by introducing the concept of blobs) as well as how to improve the simulation algorithm?

Exemplary graphical results

The set of visualizations and distributions obtained from exemplary simulation are collected below.

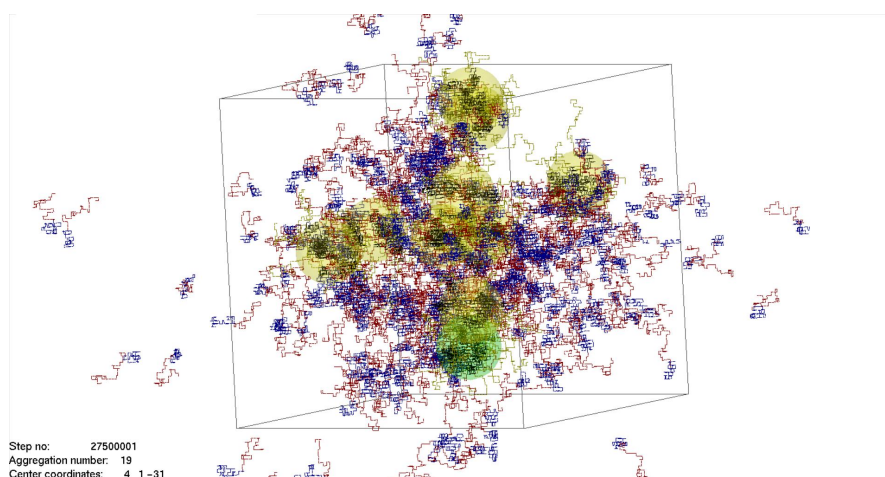


Fig. 1. Raw_cells

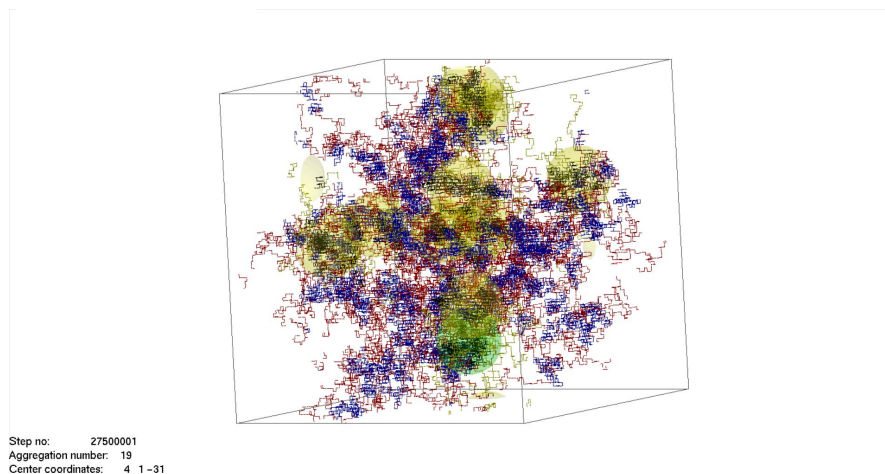


Fig. 2. Clipped_cells.

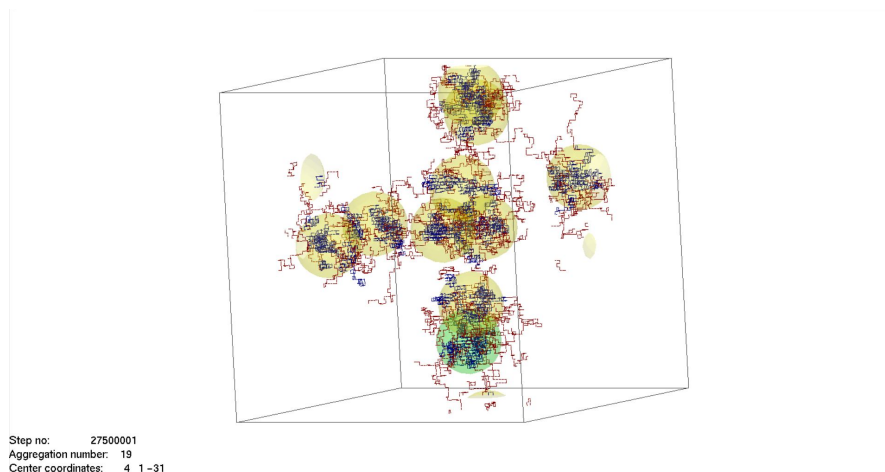


Fig. 3. Bulk macromolecules removed.

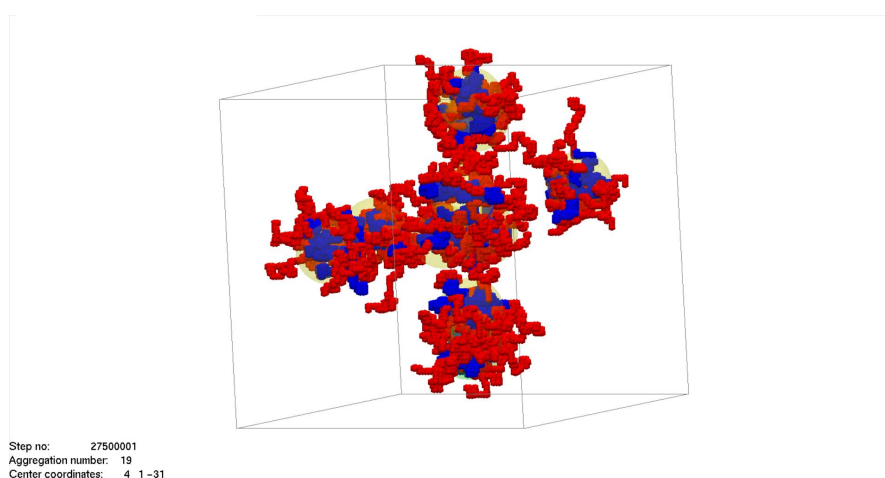


Fig. 4. The pearl necklace presentation (raw_cells)

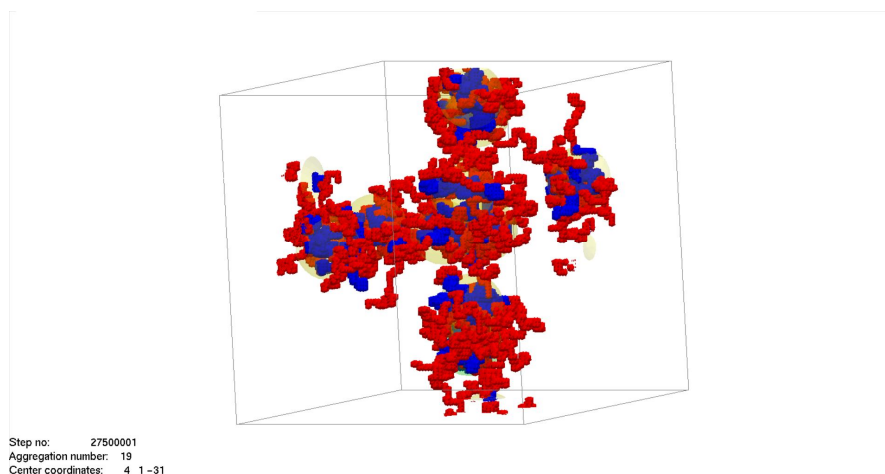


Fig. 5. The pearl necklace presentation (clipped_cells)

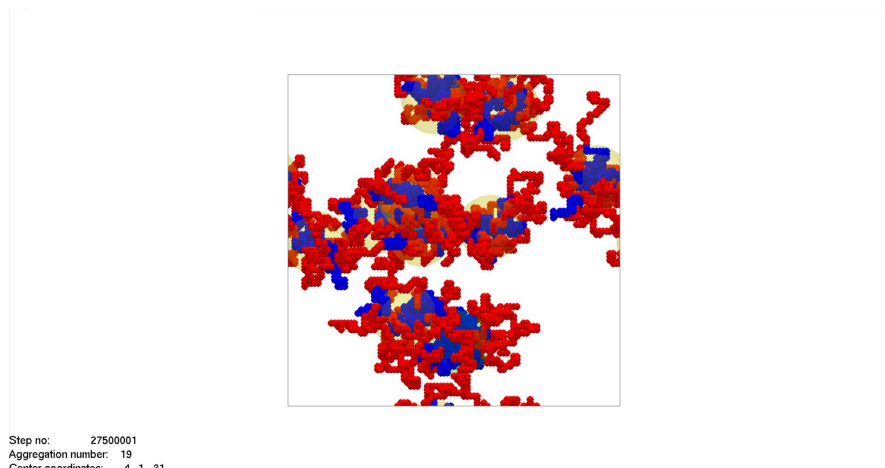


Fig. 6. Parallel projection on the 0xy plane.

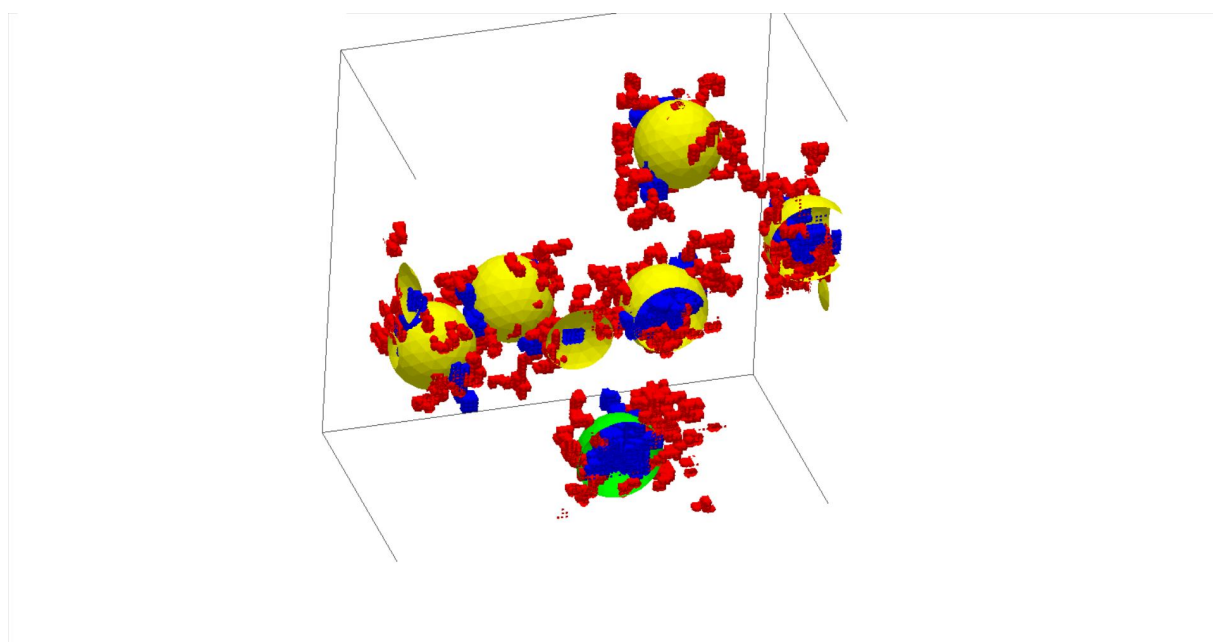


Fig. 7. Cross section of the simulation box.

Text exemplary numerical results:

R_micelle_data.txt

Label coordinates aggregation number

1	3	2	-31	17
2	-21	22	-16	15
3	2	9	5	15
4	10	-9	1	13
5	-35	-22	3	12
6	5	41	20	11
7	1	-12	23	11
8	33	-17	-25	10
9	5	-37	-18	10
10	27	-6	43	9
11	-8	35	-3	9
12	-1	17	-14	9
13	2	-23	-41	9

14	-5	-43	-1	8
15	-9	16	40	8
16	36	5	-2	8
17	13	33	3	8
18	32	-9	12	6
19	-1	-4	44	6
20	32	-37	0	6
21	38	27	-1	4
22	18	13	-44	3

R_micelle_data_E.txt

Label	aggregation_number	Conformational_entropy	Intwrnal_energy
-------	--------------------	------------------------	-----------------

1	17	135.252	-291.029
2	15	136.451	-274.333
3	15	136.476	-277.000
4	13	134.011	-299.615
5	12	135.821	-296.250
6	11	137.154	-286.136
7	11	135.528	-291.591
8	10	135.070	-295.500
9	10	138.941	-279.250
10	9	136.591	-286.111
11	9	138.232	-267.500
12	9	136.187	-295.000
13	9	138.716	-258.333
14	8	130.777	-318.438
15	8	135.859	-295.313
16	8	138.350	-261.875
17	8	139.369	-251.875
18	6	137.559	-283.333
19	6	138.287	-272.500
20	6	135.616	-295.417
21	4	141.247	-256.250
22	3	140.868	-255.000

R_micelle_data_RG_3D.txt

Label	aggregation_number	Rg_of_core	Rg_of_whole_micelle	Density_of_Core	Density_of_shell
-------	--------------------	------------	---------------------	-----------------	------------------

1	17	7.745	11.097	0.206	0.106
2	15	7.299	10.521	0.246	0.123
3	15	7.495	10.370	0.227	0.138
4	13	6.447	9.807	0.356	0.141
5	12	7.411	10.344	0.235	0.136
6	11	7.166	10.954	0.259	0.101
7	11	5.857	9.899	0.475	0.124
8	10	6.947	10.468	0.285	0.118
9	10	6.737	10.008	0.312	0.137
10	9	6.037	8.970	0.434	0.190
11	9	5.723	8.880	0.509	0.186
12	9	5.623	8.787	0.537	0.191
13	9	8.117	9.549	0.179	0.284
14	8	4.649	7.951	0.951	0.237
15	8	6.634	9.870	0.327	0.143
16	8	5.864	9.216	0.474	0.164
17	8	7.146	9.930	0.262	0.155
18	6	4.607	8.888	0.977	0.158
19	6	5.226	7.903	0.669	0.272
20	6	5.478	8.262	0.581	0.239

R_micelle_distribution.txt

Aggregation_number	Ratio_of_appearances
--------------------	----------------------

2	0.00051
---	---------

3	0.03136
4	0.04476
5	0.01922
6	0.01998
7	0.07537
8	0.12393
9	0.10799
10	0.14315
11	0.11507
12	0.09054
13	0.08978
14	0.05033
15	0.02858
16	0.01517
17	0.00683
18	0.00556
19	0.01315
20	0.00658
21	0.00228
22	0.00809
23	0.00177

Radial_GlobDist_S=1_.txt

Distance	Fraction_of_solvent_molecules	Fraction_of_core_monomers	Fraction_of_shell_monomers
----------	-------------------------------	---------------------------	----------------------------

1.000	0.9723053	0.0177291	0.0099656
1.400	0.9804938	0.0143969	0.0051093
1.700	0.9839764	0.0123365	0.0036871
2.000	0.9842380	0.0117255	0.0040365
2.200	0.9866036	0.0103227	0.0030738
2.400	0.9881182	0.0092667	0.0026150
2.800	0.9897026	0.0080199	0.0022775
3.000	0.9903795	0.0074940	0.0021265
3.200	0.9906865	0.0072167	0.0020968
3.300	0.9913431	0.0067226	0.0019343
3.500	0.9920633	0.0061653	0.0017715
3.600	0.9922339	0.0060104	0.0017557
3.700	0.9926586	0.0056724	0.0016690
4.000	0.9928975	0.0054236	0.0016789
4.100	0.9934191	0.0050254	0.0015556
4.200	0.9936294	0.0048496	0.0015210
4.400	0.9939711	0.0045775	0.0014515
4.500	0.9940558	0.0044755	0.0014687
4.600	0.9942706	0.0042945	0.0014349
4.700	0.9945638	0.0040730	0.0013632
4.900	0.9948158	0.0038351	0.0013491
5.000	0.9949068	0.0037459	0.0013473....

Radial_LocDistr_S=1_M=1.txt

Distance	Fraction_of_solvent_molecules	Fraction_of_core_monomers	Fraction_of_shell_monomers
----------	-------------------------------	---------------------------	----------------------------

1.00	0.5000000	0.5000000	0.0000000
1.40	0.4166667	0.5833333	0.0000000
1.70	0.3750000	0.6250000	0.0000000
2.00	0.3333333	0.6666667	0.0000000
2.20	0.4166667	0.5833333	0.0000000
2.40	0.4166667	0.5833333	0.0000000
2.80	0.4166667	0.5833333	0.0000000
3.00	0.4000000	0.6000000	0.0000000
3.20	0.5000000	0.5000000	0.0000000
3.30	0.5416667	0.4583333	0.0000000
3.50	0.3750000	0.6250000	0.0000000

3.60	0.5416667	0.4583333	0.0000000
3.70	0.5000000	0.5000000	0.0000000
4.00	0.6666667	0.3333333	0.0000000
4.10	0.5625000	0.4375000	0.0000000
4.20	0.5000000	0.5000000	0.0000000
4.40	0.5416667	0.4583333	0.0000000
4.50	0.6250000	0.3750000	0.0000000
4.60	0.5208333	0.4791667	0.0000000
4.70	0.5833333	0.3750000	0.0416667
4.90	0.5833333	0.3750000	0.0416667
5.00	0.5000000	0.5000000	0.0000000

Some compiled results

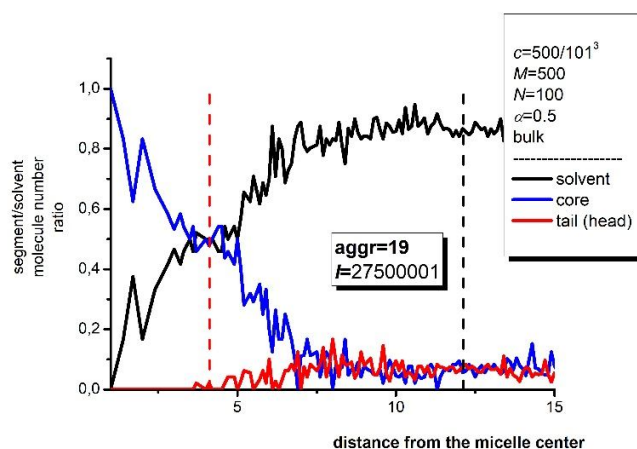


Fig.8. The distribution of “1” - core segemts, “2” - tail segments, and “0” - solvent molecules around the center of the largest micelle.

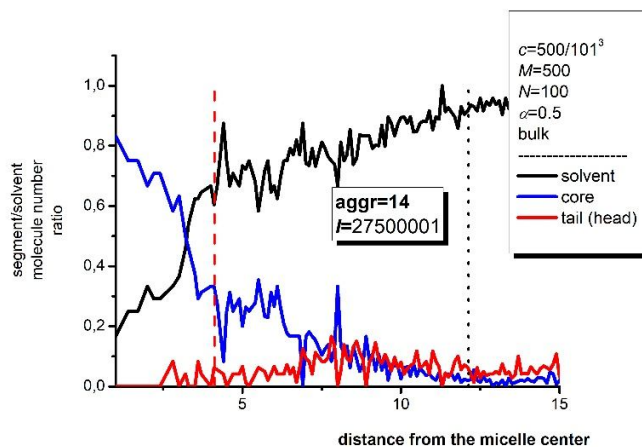


Fig. 9. The distributions of around the center of the 2nd micelle.

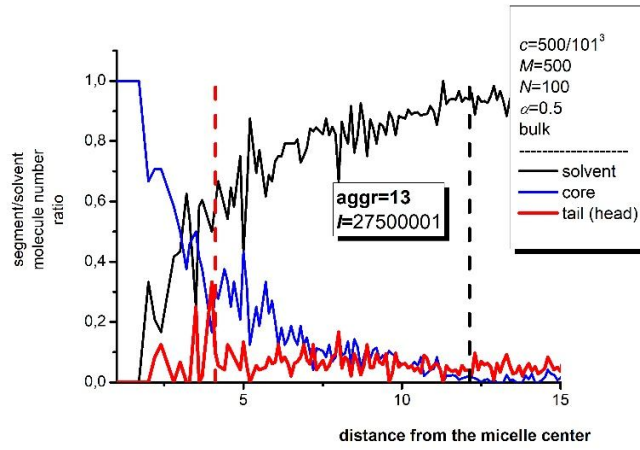


Fig. 10. The distributions of around the center of the 3th micelle.

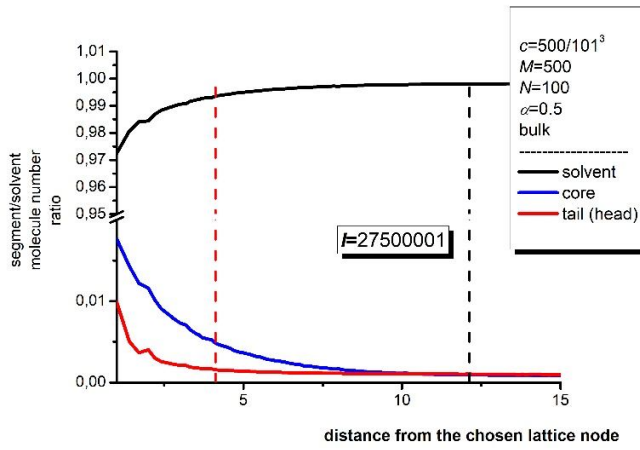


Fig. 11. The radial distribution “0”, “1” and “2” species in the whole system.

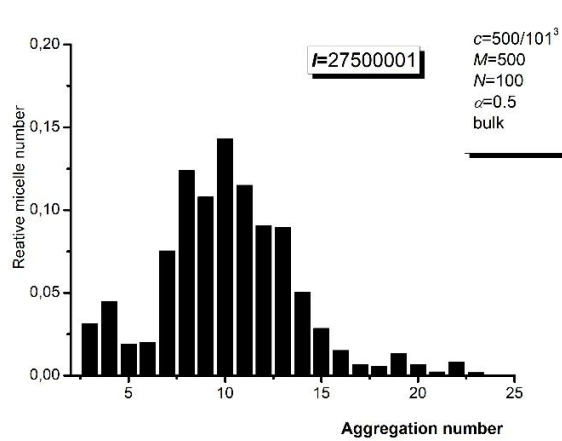


Fig. 12. The distribution of micelles of different aggregation numbers (averaged).