CMST 22(4) 187-196 (2016) DOI:10.12921/cmst.2016.0000055

Simulation of Ionic Copolymers by Molecular Dynamics

M. Dzięcielski[1], **P. Knychała**[2], **M. Banaszak**[3*]

[1]*Faculty of Geographical and Geological Sciences, Adam Mickiewicz University*
ul. Krygowskiego 10, 61-680 Poznań, Poland

[2]*Faculty of Polytechnic, The President Stanislaw Wojciechowski University School of Applied Sciences in Kalisz*
ul. Nowy Świat 4, 62-800 Kalisz, Poland

[3]*Faculty of Physics, A. Mickiewicz University*
ul. Umultowska 85, 61-614 Poznan, Poland
[*]*E-mail: mbanasz@amu.edu.pl*

Received: 19 November 2016; revised: 27 November 2016; accepted: 28 November 2016; published online: 09 December 2016

Abstract: Using GROMACS (a molecular dynamics package) we simulate ionic copolymers and compare the numerical results with those obtained by the lattice Monte Carlo simulations. While the results are qualitatively similar for both methods, the simulation times are significantly longer for the molecular dynamics simulations than those for the corresponding Monte Carlo runs.

Key words: GROMACS, molecular dynamics, ion diblock copolymer, microphase separation

I. INTRODUCTION

Block copolymers exhibit order-disorder transitions which result in formation of different types of nanostructures such as ordered lamellae, cylinders, spheres or double gyroid phase. In case of non-ionic diblock copolymer melts, one can predict stability windows for different types of morphologies from the Flory interaction parameter, χ, and length of the two blocks, N_A and N_B. The lamellar phase is favored for symmetric chains (when both blocks have similar length), whereas cylinders or spheres are recorded in asymmetric case (when block lengths differ significantly). Such copolymers have been investigated by theory and computer simulations for a long time [1-6]. The agreement between predictions of simulations, self-consistent field theory and experiment is surprisingly good [7].

In recent years, continuing interest in ion-containing block copolymers has also been observed due to their use in clean-energy-related applications such as fuel cells and batteries [8-11]. Microphase separation for ion-containing copolymers is different from that for neutral systems. For example, for symmetric diblock copolymers the non-lamellar nanotructures were observed and lamellar microphase was recorded in the asymmetric case. There have been several efforts to understand the effect of ion-containing monomers on the self-assembly of block copolymers [12-15]. Computer simulations of such copolymers are more demanding because of the long-range interactions which have to be included. Some studies suggest that a minimal lattice model without long-range interactions can be used in the case of ion-containing block copolymers [16-19]. In this model the Flory interaction parameter, χ, between blocks is replaced by an effective interaction parameter that accounts for ionic interactions. This simplification allows us to obtain results which are qualitatively consistent with experimental data.

The aim of this study is to demonstrate that GROMACS, which is a Molecular Dynamics (MD) package, can be applied to simulate ion-containing block copolymers. Specifically, we employ GROMACS to investigate ion-containing copolymers using full electrostatic interactions. This kind of simulations is expected to be much more time consuming compared with the lattice Monte Carlo (MC) simulations

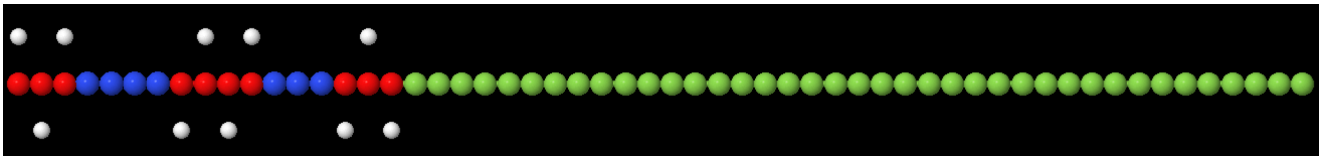


Fig. 1. The coarse-grained model of a chain, used in simulation, consisting of segments A (blue), S_+ (red), B (green), and counterions S_- (white)

which we performed in our previous publications. In this paper, simulations are limited to a single sequence of segments in a copolymer chain. We chose an asymmetric diblock chain where volume fraction of non-ionic block is 0.7. This sequence of segments was previously investigated using Monte Carlo simulations in reference [18] and compared with experimental results [11]. The obtained results were surprising because, quite counterintuitively, lamellar nanophase was observed.

II. MODEL AND METHOD

The coarse-grained model in which a group of atoms is modeled by a single segment, is widely used in computer simulations of copolymers in order to speed up the calculations. For example, each monomer, C_2H_4, consists of 6 atoms. A few such monomers are replaced by a single segment. Such simplification yields results which are consistent with theory and experiment.

The coarse-grained model of poly-(styrenesulfonate)-b-polymethylbutylene (PSS-PMB) is used (Fig. 1). The chain of length $N = 56$ consists of three segment's type: A, S_+, and B, which modeled styrene (S), sulfonated styrene (SS), methylbutylene (MB), respectively. The first two types of segments are part of ion-block whose length is 17. The length of non-ionic block is 39. It means that the chain is asymmetric and volume fraction of non-ionic block is 0.7. Moreover, the volume fraction of segments S_+ in ionic block, which correspond to experimental sulfonation level, is $p = 0.588$. The simulation box contains 59 chains and 590 counterions S_-. The reduced density is $\rho^* = 0.5$. The usual periodic boundary conditions are applied. The box is cubic with size equal to 19.8σ, where σ is the size a polymer segment.

The repulsive interactions are modeled via the WCA potential, defined as follows:

$$U_{WCA} = \begin{cases} 4\varepsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] + \varepsilon & \text{when } r < 2^{1/6}\sigma, \\ 0 & \text{when } r \geq 2^{1/6}\sigma, \end{cases}$$

where ϵ and σ are the usual Lennard-Jones parameters [20].

The harmonic potential [21] or finitely extensible nonlinear elastic (FENE) potential [22] is used to modeled bonds between segments. To simulate the system effectively at high temperatures the LINCS [23] algorithm is used, which is the simplest version of SHAKE [24] algorithm.

Moreover, the Coulomb electric potential is applied:

$$U_C = \frac{1}{4\pi\varepsilon_0\varepsilon_r} \cdot \frac{Q}{r}, \tag{1}$$

where ϵ_0 is the vacuum permittivity, and ϵ_r is the relative permittivity. The system contains neutral segments (A and B) as well as ionic elements S_+ and S_- with value of charges $+1$ and -1, respectively. The number of segments S_+ and S_- is the same so the whole system is neutral. Because of long-range interactions the Particle Mesh Ewald [25] version of Ewald [26-28] summation is applied in simulations.

The reduced dimensionless temperature, T^* is defined as follows:

$$T^* = \frac{k_B T}{E_\sigma}, \tag{2}$$

where T is the absolute temperature, k_B is the Boltzmann constant and

$$E_\sigma = \frac{1}{4\pi\varepsilon_0} \cdot \frac{e^2}{\sigma}. \tag{3}$$

Initially the polymer chains assume statistical conformations, random orientations, and are uniformly distributed within the simulation box. Next, we run thermal simulation, starting with initial velocities of segments, which are proportional to the square root of temperature. The NVT thermostat by Nose and Hoover [29, 30] is used to ensure adequate temperature. Equations of motion are solved using a leapfrog algorithm [31], a variety of the Verlet [32] method. The time step used in simulations is optimized for performance. The simulations are repeated at least 3 times starting from different initial states.

III. RESULTS AND DISCUSSION

The simulations were performed for 15 different value of temperature, T^*, which are expressed in reduced units [33].

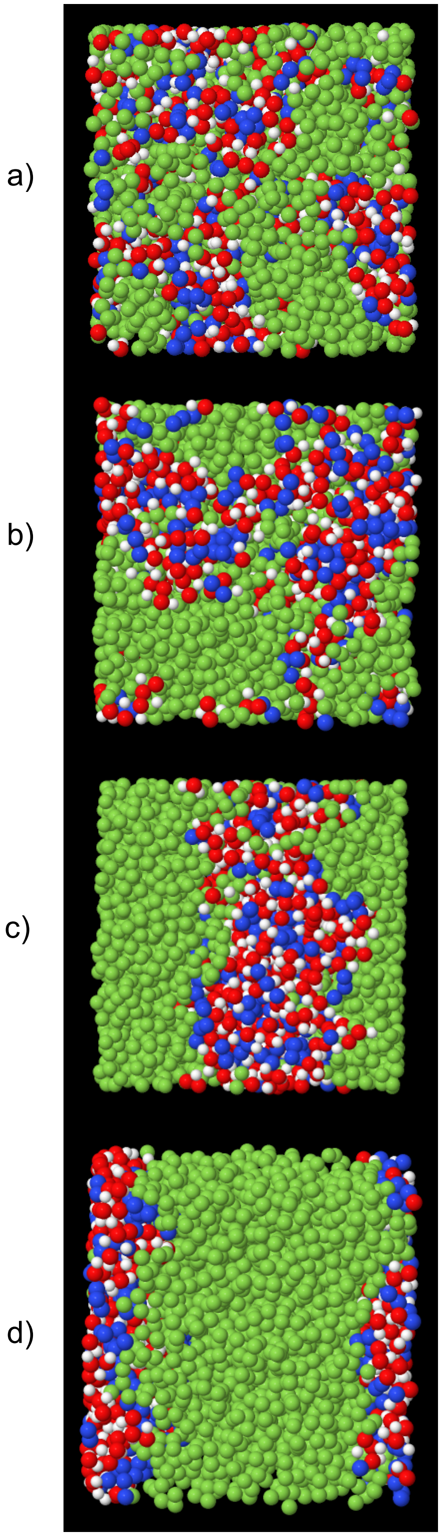


Fig. 2. Selected nanostructures from simulations: a) disordered phase at $T^* = 0.18$ b) disordered phase at $T^* = 0.12$, c) lamellar phase at $T^* = 0.084$, d) lamellar phase at $T^* = 0.042$. The segments A are shown in blue, S_+ in red, B in green, and counterions S_- in white

Typical snapshots from simulations are shown in Fig. 2. In particular, at high temperatures ($T^* = 0.18$ and 0.12) the disordered nanophase is observed (Fig. 2 a, b), and at low temperatures ($T^* = 0.084$ and $T^* = 0.042$) the lamellar nanostructure is obtained (Fig. 2 c and d). The structure factor is calculated using the following equation:

$$S(\vec{k}) = \frac{1}{n_\alpha}\left\langle \left(\sum_{m=1}^{n_\alpha}\cos(\vec{k}\cdot\vec{r}_m)\right)^2 + \left(\sum_{m=1}^{n_\alpha}\sin(\vec{k}\cdot\vec{r}_m)\right)^2 \right\rangle_{\text{thermal average}}, \tag{4}$$

The simulated structure factors are presented in Fig. 3. At $T^* = 0.18$ and 0.12, we can observe a single broad peak which is characteristic for a nanostructure without long range arrangement. The $S(k)$ calculated for $T^* = 0.084$ and 0.042 are also presented in Fig. 3, where one can observe the first-order peak k^* as well as the high-order peaks. The ratio $k/k^* = 1$ and 2 is observed at $T^* = 0.084$. Additionally, at $T^* = 0.042$, the third peak for $k/k^* = 4$ is obtained. The ratios k/k^* indicate the lamellar nanophase.

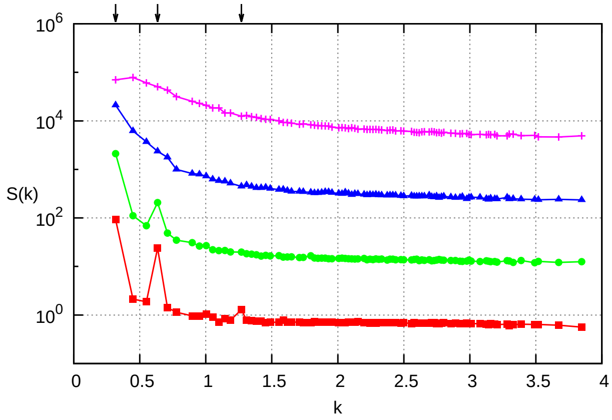


Fig. 3. Structure factor calculated do $\alpha = B$ at: $T^* = 0.18$ (pink crosses), $T^* = 0.12$ (blue triangles), $T^* = 0.084$ (green circles), $T^* = 0.042$ (red squares). The arrows show peaks at $k/k^* = 1$, 2, and 4 indicate L phase. The function for each temperature are shifted for clarity

The average concentration profiles of $A + S$ (ion blocks) and B (non-ion blocks) are plotted as a function of distance along the vector normal to the layer in Fig. 4. At $T^* = 0.18$ and 0.12, the peaks in the profiles are broad, which means that blocks of different types are not fully separated. At lower temperatures ($T^* = 0.084$ and 0.042) the results are completely different because one can observe clear peaks. The interface between ion and non-ion blocks becomes more sharp and clear.

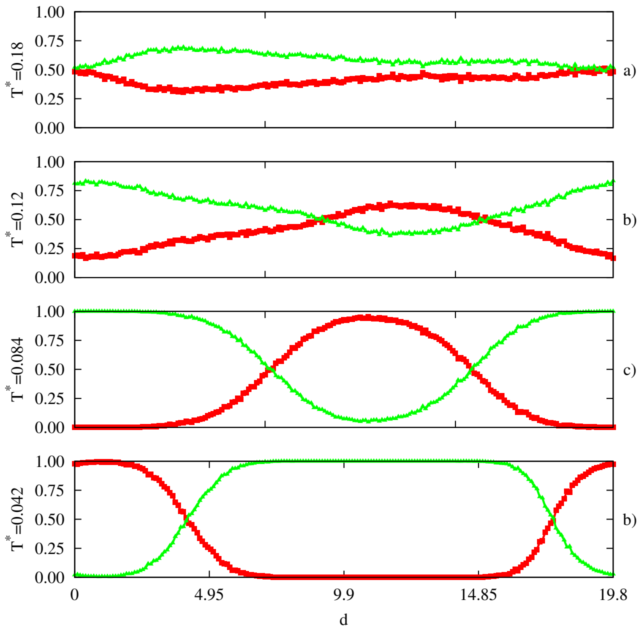


Fig. 4. Monomer concentration profiles at: a) $T^* = 0.18$ (disordered phase), b) $T^* = 0.12$ (disordered phase), c) $T^* = 0.084$ (L phase), a) $T^* = 0.042$ (L phase). The profiles for $A + S$ and B segments are shown in red and green, respectively

To determine order-disorder transition temperature, T^*_{ODT}, more precisely, we calculated the structure factor for four selected T^*'s near microphase separation temperature. The results are presented in Fig.5. At $T^* = 0.108$, the $S(k)$ is smooth with one broad peak which means that disordered nanophase is recorded. At $T^* = 0.102$, an additional peak is observed which, means that the system starts to self-assembly. This second order peak becomes increasingly clear and, at two lower temperatures, the high-order peaks appear. Using these results one can estimate T^*_{ODT} to be about 0.099.

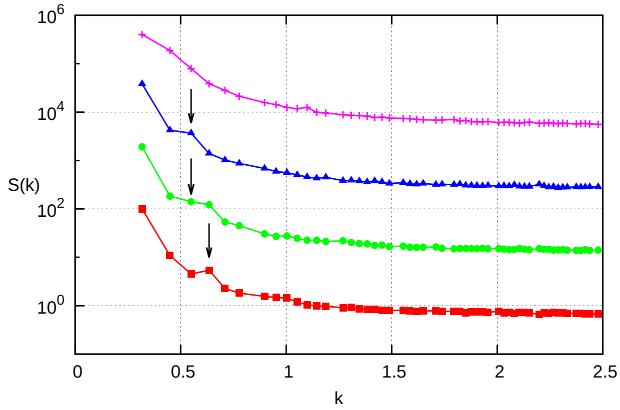


Fig. 5. Structure factor at $T^* = 0.108$ (pink crosses), $T^* = 0.102$ (blue triangles), $T^* = 0.096$ (green circles), $T^* = 0.090$ (red squares). The function for each temperature are shifted for clarity

It is surprising that for an asymmetric diblock copolymer with volume fraction non-ion block 0.7, the lamellar nanostructures are observed. The cylindrical nanophase is recorded for non-ionic diblock for the same volume fraction [3, 34]. The same results were obtained experimentally [11] as well as with Monte Carlo simulations, where coarse-grained block copolymers were studied by a minimal lattice model with short-range interactions [18]. Application of such a simplified model to investigate ion-copolymers is not obvious. Those results suggest that both methods allow us to obtain qualitatively consistent results. The MD is a more accurate method but significantly more time consuming. It would take more time to obtain full phase diagrams (as in references [16-18]) depending on different lengths of chain, symmetry and segment sequences using molecular dynamic simulations with long-range interactions.

To compare the efficiency of both methods, we performed simulations of a test system using molecular dynamics (MD) with GROMACS software and the lattice Monte Carlo (MC) method. We recorded the high-order peaks in $S(k)$ to determine when the microphase separation takes place. In case of molecular simulations, it took $2.1 \cdot 10^6$ simulations steps. It corresponds to $8.2 \cdot 10^5$ real seconds of CPU using. The Monte Carlo simulations took $8 \cdot 10^5$ Monte Carlo steps which corresponds to 120 real seconds. The results are summarized in tab. 1. The MD simulation took nearly 7000 more times of CPU time than calculations using the MC method. It is related to a huge number of floating point calculations in MD, and also with taking into account the long range electrostatic interactions.

Tab. 1. Time of simulation of test system using molecular dynamics (GROMACS software) and lattice Monte Carlo calculation

Method	**time [s]**
Molecular dynamics	820800
Monte Carlo	120

IV. CONCLUSIONS

The molecular simulations of asymmetric ionic diblock copolymers with long range interactions are performed using GROMACS. Simulations have been limited to a single sequence of segments. The lamellar nanostructure is observed for asymmetric diblock copolymers where volume fraction of non-ionic block is 0.7. It is a surprising result since all copolymer theories predict non-lamellar phases for non-ionic diblock copolymers at this volume fraction. Nevertheless, this result is qualitatively consistent with that obtained for a minimal lattice model of Monte Carlo simulations. Moreover, the result is in accordance with the experimental result of PSS-b-PMB. The simulations also show that the MD method is significantly more time consuming that the lattice MC. This result suggests that using a simple lattice model to simulate ion-containing melts, like PSS-b-PMB, is appropriate, which was not obvious previously.

Acknowledgments

PK and MB gratefully acknowledge the computational grant from the Supercomputing and Networking Center (PCSS) in Poznan, Poland and grant N N204 018938 from Polish Ministry of Science and Higher Education.

APPENDIX

IV. 1. GROMACS calculations

GROMACS is one of the most popular software packages designed for simulating, biomolecular systems such as proteins and lipids using a molecular dynamics method [35]. It can also be used to simulate copolymers using coarse-grained model. In case of a huge system, like a copolymers system which contains a lot of chains, calculations are usually made at high performance computing clusters. This software does not have a graphical user interface and the simulations are configured and run using a console text mode. At least three files have to be prepared to start simulation of block copolymers (*grompp.mdp*, *topol.top* i *conf.gro*). Those files are described briefly herein.

The first file, *grompp.mdp*, contains the following simulation parameters:

- *integrator* - algorithm used to solve equations of motion, option *md* means Verlet algorithm at *leap-frog* version,
- *nsteps* - number of simulation steps,
- *nstlist* - the frequency (in steps of simulation) of updating the list of neighborhood,
- *nstxout* - the frequency (in steps of simulation) of saving segments coordinates to *topol.tpr* file, which can be used to calculate structural parameters of investigated system (e.g structure factor),
- *nstlog* - the frequency (in steps of simulation) of saving information about simulation to *md.log* file,
- *dt* - time step (in *ps*) used to solve equations of motion,
- *constraints* - information which molecules have bonds,
- *ns_type* - way of creating the list of neighborhood, option *grid* means that the grid is created in the simulation box,
- *pbc* - in which directions the periodic boundary condition are turned on,
- *periodic-molecules* - turn on/off periodic boundary conditions,
- *vdwtype* - van der Waals interactions defined by user, in our simulation it is WCA,
- *coulombtype* - way of long-range interactions calculation, the *pme* means Particle-Mesh Ewald,
- *rlist* - maximum distance (in *nm*) used in the list of neighborhood,
- *rvdw* - van der Waals potential range (in *nm*),
- *rcoulomb* - electrostatic potential range (in *nm*),
- *epsilon-r* - the relative dielectric constant,
- *tcoupl* - thermostat type,
- *tc_grps* - which segments should be at constant temperature, option *system* means all segments,
- *tau_t* - the frequency (in steps of simulation) of correction by thermostat,
- *nsttcouple* - the frequency (in steps of simulation) of coupling the temperature,
- *ref_t* - the simulation temperature in Kelvin,
- *Pcoupl* - the pressure coupling algorithm, option *no* means no pressure coupling.

GROMACS allows to use interaction potential defined by user [36]. The WCA potential, which is not added by default in this package, is used in our simulations. To use it, the file *table.xvg* which contains values of selected potential should be created. The details of this procedure are described in GROMACS documentation [37].

The *topol.top* file contains information about the topological structure of copolymers. It contains several sections which have the following meaning:

- *defaults* - contains 5 parameters:
  - *nbfunc* - the non-bonded function type (number 1 means Lennard-Jones potential),
  - *comb-rule* - the number of the combination rule (number 2 is used in case of reduced units and Lennard-Jones potential),
  - *gen-pairs* - scaling factor for electrostatic interactions, not used in simulations,
  - *udgeLJ* - the factor by which to multiply Lennard-Jones interactions,
  - *fudgeQQ* - the factor by which to multiply electrostatic interactions,
- *atomtypes* - information about atoms (name, mass, charge, particle type - value A means atom, sigma and epsilon),
- *bondtypes* - information about bonds (names atoms in bond, bond type, bond length, bond energy),
- *moleculetype* - information about name of molecule and number of excluded neighbors for non-bonded interactions.
- *atoms* - information about atoms (number, type, residue number, residue name, name, charge group number and charge value),
- *bonds* - information about bonds (numbers of bonded atoms, bond type),
- *system* - name of our system,
- *molecules* - name of our molecule.

More details about this file are described in chapter 5.7 of GROMACS documentation [33].

The last described file (*conf.gro*) contains information about initial coordinates of all segments. The used *gro* format is close to xyz format and can contains the following information:

- title,
- number of segments,
- information about segments,
- the box size.

Each row with information about segments has the following format:

- chain's number (maximum 5 intiger digits) and chains's name (maximum 5 characters),
- segment's name (maximum 5 characters),
- segment's number (maximum 5 intiger digits),
- segment's coordinates (x, y, z; maximum 8 real digits with an accuracy of 3 decimal places),
- segment's velocities (x, y, z; maximum 8 real digits with an accuracy of 4 decimal places).

Those 3 files should be placed in a simulation directory and compiled using *grompp* command to binary file *topol.tpr*. The simulation is run using the *mdrun* command which has many options. One of them is *-nt* which determines the number of processes to be used in simulation. The *-v* option allow to see the predicted finish time of calculations.

The final results of simulation are 4 files: *confout.gro*, *md.log*, *ener.edr*, and *traj.trr*. The *confout.gro* contains the final coordinates of all segments and can be transformed to xyz format, which can be visualized using V-sim, RasmMol or JMol. The *energy* command can be used to calculate a different type of energy at selected time of simulation. Similarly, the *trajconv* command can be used to obtain coordinates of segments.

The grompp.mdp file

```
integrator         = md
nsteps             = 1000000
nstlist            = 10
nstxout            = 100000
nstlog             = 50000
dt                 = 0.001
constraints        = all-bonds
ns_type            = grid
pbc                = xyz
periodic-molecules = yes
vdwtype            = User
coulombtype        = pme
rlist              = 1.25
rvdw               = 1.13
rcoulomb           = 1.25
epsilon-r          = 1
tcoupl             = nose-hoover
tc_grps            = system
tau_t              = 1
nsttcouple         = 1
ref_t              = 400
Pcoupl             = no
```

The topol.top file

```
[ defaults ]
  1             2               no               1.0      1.0

[ atomtypes ]
 N      1.0         0.000     A               1.0             1.0
 O      1.0        -1.000     A               1.0             1.0
 H      1.0         1.000     A               1.0             1.0
 F      1.0         0.000     A               1.0             1.0

[ bondtypes ]
N  O    1   1           350.
N  N    1   1           350.
O  F    1   1           350.
O  O    1   1           350.
F  F    1   1           350.

[moleculetype]
PE6000        1
```

```
[atoms]
;   nr    type   resnr  residu    atom    cgnr       charge
     1     O       1       O      O1          1   -1.0
     2     O       1       O      O2          2   -1.0
     3     O       1       O      O3          3   -1.0
     4     N       1       N      N1          4    0.0
     5     N       1       N      N2          5    0.0
     6     N       1       N      N3          6    0.0
     7     N       1       N      N4          7    0.0
     8     O       1       O      O4          8   -1.0
     9     O       1       O      O5          9   -1.0
    10     O       1       O      O6         10   -1.0
    11     O       1       O      O7         11   -1.0
    12     N       1       N      N5         12    0.0
    13     N       1       N      N6         13    0.0
    14     N       1       N      N7         14    0.0
    15     O       1       O      O8         15   -1.0
    16     O       1       O      O9         16   -1.0
    17     O       1       O     O10         17   -1.0
    18     F       1       F      F1         18    0.0
    19     F       1       F      F2         19    0.0
    20     F       1       F      F3         20    0.0
                          ...
  3893     H       1       H    H589       3893    1.0
  3894     H       1       H    H590       3894    1.0

[bonds]
;  ai    aj funct
    1     2  1
    2     3  1
    3     4  1
    4     5  1
    5     6  1
    6     7  1
    7     8  1
    8     9  1
    9    10  1
   10    11  1
      ...
 3299  3300  1
 3300  3301  1
 3301  3302  1
 3302  3303  1
 3303  3304  1

[system]
pe

[molecules]
PE6000       1
```

The conf.gro file

```
pol
```

```
 3894
    1AA      O1    1   1.414   9.192  19.092  0.0000  0.0000  0.0000
    1AA      O2    2   0.707   9.192  18.385  0.0000  0.0000  0.0000
    1AA      O3    3   0.707   8.485  19.092  0.0000  0.0000  0.0000
    1AA      N1    4   0.000   7.778  19.092  0.0000  0.0000  0.0000
    1AA      N2    5  19.092   7.778   0.000  0.0000  0.0000  0.0000
    1AA      N3    6  18.385   7.071   0.000  0.0000  0.0000  0.0000
    1AA      N4    7  18.385   7.778  19.092  0.0000  0.0000  0.0000
    1AA      O4    8  18.385   7.071  18.385  0.0000  0.0000  0.0000
    1AA      O5    9  19.092   7.071  17.678  0.0000  0.0000  0.0000
    1AA      O6   10   0.000   7.778  17.678  0.0000  0.0000  0.0000
    1AA      O7   11   0.707   8.485  17.678  0.0000  0.0000  0.0000
    1AA      N5   12   0.707   7.778  16.971  0.0000  0.0000  0.0000
    1AA      N6   13   0.707   7.071  17.678  0.0000  0.0000  0.0000
    1AA      N7   14   0.000   7.071  18.385  0.0000  0.0000  0.0000
    1AA      O8   15   0.707   6.364  18.385  0.0000  0.0000  0.0000
    1AA      O9   16   0.707   5.657  17.678  0.0000  0.0000  0.0000
    1AA     O10   17   0.000   5.657  18.385  0.0000  0.0000  0.0000
    1AA      F1   18  19.092   4.950  18.385  0.0000  0.0000  0.0000
    1AA      F2   19  18.385   5.657  18.385  0.0000  0.0000  0.0000
    1AA      F3   20  17.678   6.364  18.385  0.0000  0.0000  0.0000
    1AA      F4   21  18.385   6.364  19.092  0.0000  0.0000  0.0000
    1AA      F5   22  17.678   5.657  19.092  0.0000  0.0000  0.0000
    1AA      F6   23  16.971   6.364  19.092  0.0000  0.0000  0.0000
    1AA      F7   24  16.263   6.364  18.385  0.0000  0.0000  0.0000
    1AA      F8   25  15.556   6.364  17.678  0.0000  0.0000  0.0000
    1AA      F9   26  15.556   7.071  16.971  0.0000  0.0000  0.0000
    1AA     F10   27  16.263   6.364  16.971  0.0000  0.0000  0.0000
    1AA     F11   28  15.556   6.364  16.263  0.0000  0.0000  0.0000
    1AA     F12   29  16.263   6.364  15.556  0.0000  0.0000  0.0000
    1AA     F13   30  16.971   5.657  15.556  0.0000  0.0000  0.0000
    1AA     F14   31  16.971   6.364  14.849  0.0000  0.0000  0.0000
    1AA     F15   32  16.971   5.657  14.142  0.0000  0.0000  0.0000
    1AA     F16   33  17.678   5.657  13.435  0.0000  0.0000  0.0000
    1AA     F17   34  18.385   6.364  13.435  0.0000  0.0000  0.0000
    1AA     F18   35  19.092   7.071  13.435  0.0000  0.0000  0.0000
    1AA     F19   36  18.385   7.071  14.142  0.0000  0.0000  0.0000
    1AA     F20   37  17.678   7.071  14.849  0.0000  0.0000  0.0000
    1AA     F21   38  18.385   7.778  14.849  0.0000  0.0000  0.0000
    1AA     F22   39  17.678   7.778  14.142  0.0000  0.0000  0.0000
    1AA     F23   40  17.678   8.485  14.849  0.0000  0.0000  0.0000
    1AA     F24   41  16.971   8.485  15.556  0.0000  0.0000  0.0000
    1AA     F25   42  16.263   8.485  14.849  0.0000  0.0000  0.0000
    1AA     F26   43  16.971   9.192  14.849  0.0000  0.0000  0.0000
    1AA     F27   44  16.263   9.192  15.556  0.0000  0.0000  0.0000
    1AA     F28   45  16.971   9.192  16.263  0.0000  0.0000  0.0000
    1AA     F29   46  16.263   8.485  16.263  0.0000  0.0000  0.0000
    1AA     F30   47  15.556   9.192  16.263  0.0000  0.0000  0.0000
    1AA     F31   48  14.849   9.899  16.263  0.0000  0.0000  0.0000
    1AA     F32   49  14.849   9.192  15.556  0.0000  0.0000  0.0000
    1AA     F33   50  14.849   8.485  16.263  0.0000  0.0000  0.0000
    1AA     F34   51  14.142   7.778  16.263  0.0000  0.0000  0.0000
    1AA     F35   52  14.849   7.778  15.556  0.0000  0.0000  0.0000
    1AA     F36   53  14.849   7.071  16.263  0.0000  0.0000  0.0000
```

```
  1AA     F37   54  14.142   6.364  16.263  0.0000  0.0000  0.0000
  1AA     F38   55  14.849   5.657  16.263  0.0000  0.0000  0.0000
  1AA     F39   56  15.556   5.657  15.556  0.0000  0.0000  0.0000
  2AB     O11   57   4.950  12.728  19.092  0.0000  0.0000  0.0000
  2AB     O12   58   5.657  12.728  18.385  0.0000  0.0000  0.0000
                               ...
647ZV    H588 3892  16.263  19.092  16.971  0.0000  0.0000  0.0000
648ZW    H589 3893  16.263   1.414  17.678  0.0000  0.0000  0.0000
649ZX    H590 3894  19.092   1.414  17.678  0.0000  0.0000  0.0000
19.79899  19.79899  19.79899
```

References

[1] L. Leibler, *Theory of Microphase Separation in Block Copolymers*, Macromolecules **13**(6), 1602 (1980).

[2] A.E. Likhtman and A.N. Semenov, *Stability of the OBDD structure for diblock copolymer melts in the strong segregation limit*, Macromolecules **27**(11), 3103 (1994).

[3] M.W. Matsen and M. Schick, *Stable and unstable phases of a diblock copolymer melt*, Phys. Rev. Lett. **72**(16), 2660 (1994).

[4] M.W. Matsen and F.S. Bates, *Block copolymer microstructures in the intermediate-segregation regime*, J. Chem. Phys. **106**, 2436 (1997).

[5] M. Takenaka, T. Wakada, S. Akasaka, S. Nishitsuji, K. Saijo, H. Shimizu, M.I. Kim, and H. Hasegawa, *Orthorhombic Fddd Network in Diblock Copolymer Melts*, Macromolecules **40**(13), 4399 (2007).

[6] B. Miao and R.A. Wickham, *Fluctuation effects and the stability of the Fddd network phase in diblock copolymer melts*, J. Chem. Phys. **128**, 054902 (2008).

[7] M.W. Matsen, *The standard Gaussian model for block copolymer melts*, J. Phys.: Condens. Matter **14**(2), R21 (2002).

[8] K.A. Mauritz and R.B. Moore, *State of Understanding of Nafion*, Chem. Rev. **104**, 4535 (2004).

[9] Michael Hickner, Hossein Ghassemi, Yu Seung Kim, Brian R Einsla, and James E McGrath, *Alternative polymer systems for proton exchange membranes (PEMs)*, Chemical Reviews **104**(10), 4587 (2004).

[10] M.J. Park and N.P. Balsara, *Phase Behavior of Symmetric Sulfonated Block Copolymers*, Macromolecules **41**, 3678 (2008).

[11] Xin Wang, Sergey Yakovlev, Keith M. Beers, Moon J. Park, Scott a. Mullin, Kenneth H. Downing, and Nitash P. Balsara, *On the Origin of Slow Changes in Ionic Conductivity of Model Block Copolymer Electrolyte Membranes in Contact with Humid Air*, Macromolecules **43**(12), 5306 (2010).

[12] Y. Rabin and J.F. Marko, *Microphase separation in charged diblock copolymers: the weak segregation limit*, Macromolecules **24**(8), 2134 (1991).

[13] R. Kumar and M. Muthukumar, *Microphase separation in polyelectrolytic diblock copolymer melt: weak segregation limit*, J. Chem. Phys. **126**, 214902 (2007).

[14] I. Nakamura, I., N.P. Balsara, and Z.-G. Wang, *Thermodynamics of Ion-Containing Polymer Blends and Block Copolymers*, Phys. Rev. Lett. **107**, 198301 (2011).

[15] I. Nakamura, I. and Z.-G. Wang, *Salt-doped block copolymers: ion distribution, domain spacing and effective [small chi] parameter*, Soft Matter **8**, 9356 (2012).

[16] P. Knychała, M. Banaszak, Park M. J., and N.P. Balsara, *Microphase Separation in Sulfonated Block Copolymers Studied by Monte Carlo Simulations*, Macromolecules **42**(22), 8925 (2009).

[17] P. Knychała, M. Dzięcielski, M. Banaszak, and N.P. Balsara, *Phase Behavior of Ionic Block Copolymers Studied by a Minimal Lattice Model with Short-Range Interactions*, Macromolecules **46**(14), 5724 (2013).

[18] P. Knychała, M. Banaszak, and N.P. Balsara, *Effects of Composition on the Phase Behavior of Ion-Containing Block Copolymers Studied by a Minimal Lattice Model*, Macromolecules **47**(7), 2529 (2014).

[19] P. Knychała and M. Banaszak, *Simulations on a swollen gyroid nanostructure in thin films relevant to systems of ionic block copolymers*, European Physical Journal E **37**, 67 (2014).

[20] J.E. Lennard-Jones, In *On the Determination of Molecular Fields*, volume 106, page 463 The Royal Society, 1924.

[21] S. Blonski, W. Brostow, and J. Kubat, *Molecular-Dynamics Simulations of Stress-Relaxation in Metals and Polymers*, Physical Review B **49**, 6494 (1994).

[22] K. Kremer and G.S. Grest, *ynamics of Entangled Linear Polymer Melts: A Molecular-Dynamics Simulation*, J. Chem. Phys. **92**, 5057 (1990).

[23] B. Hess, H. Bekker, H.J.C. Berendsen, and J.G.E.M Fraaije, *LINCS: A Linear Constraint Solver for Molecular Simulations*, Journal of Computational Chemistry **18**, 1463 (1997).

[24] J.P. Ryckaert, G. Ciccotti, and H.J.C. Berendsen, *Numerical Integration of the Cartesian Equations of Motion of a System with Constraints: Molecular Dynamics of n-Alkanes*, Journal of Computational Physics **23**, 327 (1977).

[25] T. Darden, D. York, and L. Pedersen, *Particle Mesh Ewald: An N-log(N) Method for Ewald Sums in Large Systems*, J. Chem. Phys. **98**, 10089 (1993).

[26] P. Ewald, *Die Berechnung optischer und elektrostatischer Gitterpotentiale*, Ann. Phys. **369**, 253 (1921).

[27] M.P. Allen and D. Tildesley, *Computer Simulation of Liquids*, Clarendon Press Oxford, 1989.

[28] J. Kolafa and J.W. Perram, *Cutoff Errors in the Ewald Summation Formulae for Point Charge Systems*, Molecular Simulation **9**, 351 (1992).

[29] S. Nose, *A Unified Formulation of the Constant Temperature Molecular-Dynamics Methods*, J. Chem. Phys. **81**, 511 (1984).

[30] W.G. Hoover, *Canonical Dynamics: Equilibrium Phase-Space Distribution*, Phys. Rev. A **31**, 1695 (1984).

[31] W.F. Van Gunsteren and H.J.C. Berendsen, *A Leap-frog Algorithm for Stochastic Dynamics*, Molecular Simulation **1**, 173 (1988).

[32] L. Verlet, *Computer Experiments on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules*, Physical Review **159**, 98 (1967).

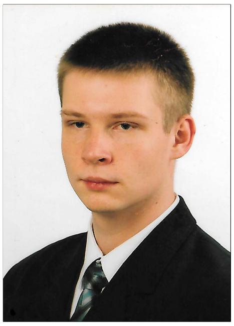
[33] Gromacs User Manual 4.0, ftp://ftp.gromacs.org/pub/manual/manual-4.0.pdf, Available: 30.08.2015.

[34] T.M. Beardsley and M.W. Matsen, *Monte Carlo phase diagram for diblock copolymer melts*, Eur. Phys. J. E **32**(3), 155 (2010).

[35] D. Van Der Spoel, E. Lindahl, B. Hess, G. Groenhof, A.E. Mark, and H.J.C. Berendsen, *GROMACS: Fast, Flexible, and Free*, J. Comput. Chem. **26**, 1701 (2005).

[36] Tabulated Potentials, http://www.gromacs.org/Documentation/How-tos/Tabulated_Potentials, Available: 30.08.2015.

[37] User Specified non-bonded potentials in gromacs, http://www.gromacs.org/@api/deki/files/94/=gromacs_nb.pdf, Available: 29.10.2016.



Michał Dzięcielski
received his PhD degree in Physics in 2016 from the Adam Mickiewicz University in Poznań. Since 2016 he has been working at the Institute of Socio-Economic Geography and Spatial Management of the Adam Mickiewicz University in Poznań. His main areas of interest are: computer simulations, complex systems, self-organization and polymer physics.

Piotr Knychała graduated in Computer Science in 2007 from the Adam Mickiewicz University in Poznań, received his PhD degree in Physics in 2012 from the Adam Mickiewicz University in Poznań. Since 2011 he has been working at the Faculty of Polytechnic of the President Stanislaw Wojciechowski University School of Applied Sciences in Kalisz. His main research interests include polymer melts and solutions, block copolymer nanostructures, ionic copolymers, and coarse-grained models of biopolymers investigated by the computer simulations.

Michał Banaszak graduated in Theoretical Physics in 1985 from the Adam Mickiewicz University in Poznań. He received his PhD degree in Physics in 1991 from Memorial University in St. John's, Canada, specializing in Polymer Physics. From 1992 to 1995 he worked as postdoctoral fellow in Exxon Research & Engineering Co. in Annandale, New Jersey, USA, working on theory and modeling of polymer solutions. From 1995 to 1997 he worked in the Chemistry Department of UMIST in Manchester, UK, as a research associate, specializing in large-scale computer simulations of ionic copolymers. In 1997 he joined the Adam Mickiewicz University, obtaining DSc degree (habilitation) in 2004 in Physics (specialty: Soft Matter Physics and Computer Simulations), and later professorship in Physics. His main interest is in developing new models and theories for nanoscale self-assembly of various polymer systems. He also collaborates with experimentalists, using large-scale computing.