Advanced Computer Simulations of Nafion / Water Systems

Dissertation

zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften

Dr. rer. nat.

vorgelegt von

Gabriel Marchand, MSc

geboren in Bordeaux

Insitut für Chemie

der

Universität Duisburg-Essen

2012

Die vorliegende Arbeit wurde im Zeitraum von September 2008 bis Juni 2012 im Arbeitskreis von Prof. Dr. Eckhard Spohr am Lehrstuhl für theoretische Chemie der Fakultät für Chemie der Universität Duisburg-Essen und unter Anleitung von Prof. Dr. Philippe A. Bopp am Fachbereich (UFR) Chemie der Universität Bordeaux 1 durchgeführt.

Tag der Disputation: 16 Juli 2012

Gutachter:	Prof. Dr. Eckhard Spohr
	Prof. Dr. Stefan Kast
	Prof. Dr. Michael Probst
Prüfer:	Prof. Dr. Jean-Christophe Soetens
	Prof. Dr. Philippe A. Bopp
	Prof. Dr. Georg Jansen
Vorsitzender:	Prof. Dr. Georg Jansen

N° d'ordre: 4366

THÈSE

PRÉSENTÉE À

L'UNIVERSITÉ BORDEAUX 1

ÉCOLE DOCTORALE DES SCIENCES CHIMIQUES

Par Gabriel MARCHAND

POUR OBTENIR LE GRADE DE

DOCTEUR

SPÉCIALITÉ: chimie physique

ADVANCED COMPUTER SIMULATIONS OF NAFION / WATER SYSTEMS

Directeurs de thèse: Professeur Eckhard Spohr (Universität Duisburg-Essen),
Professeur Philippe A. Bopp (Université Bordeaux 1)

Soutenance le: 16 juillet 2012

Devant la commission d'examen formée de:

Jansen, Georg	Professeur à l'université Duisburg-Essen	*Président du Jury*
Kast, Stefan	Professeur à l'université de Dortmund	*Rapporteur*
Probst, Michael M.	Professeur à l'université d'Innsbruck	*Rapporteur*
Spohr, Eckhard	Professeur à l'université Duisburg-Essen	*Directeur de thèse*
Bopp,Philippe A.	Professeur à l'université Bordeaux 1	*Directeur de thèse*
Soetens, Jean-Christophe	Professeur à l'université Bordeaux 1	*Examinateur*
Jansen, Georg	Professeur à l'université Duisburg-Essen	*Examinateur*

"Toute la science du monde ne vaut pas les larmes des enfants"

Fiodor Dostoïevski

Abstract

Perfluorinated membranes are used in particular in polymer electrolyte fuel cells (PEFC). The well-known ionomer Nafion$^{®}$ (Dupont) is, due to its high proton mobility, a reference material for fuel cell applications. In water or other hydrophilic solvents the membrane segregates into a hydrophobic backbone matrix and a hydrophilic sub-phase containing clusters of both water and ions, where the cluster sizes and connectivity increase with increasing water content [1].

What is the Nafion morphology and the structure of the solvent in such systems? It has been shown recently [2] on large simulated systems that several morphological models fit the experimental scattering data, suggesting the inability of scattering experiments alone to elucidate the true structure of Nafion. However, a 'random' model described in [2], i.e. the only explored model that did not assume a particular initial structure, could not reproduce the experimental data.

It remains a real computational challenge to generate in molecular simulations system configurations which are really decorrelated from the initial one. The time scales that can be achieved simply do not allow to obtain significant motions of the polymer (e.g. conformational changes, folding, etc.). We thus propose in this work a new random model of Nafion. A newly developped algorithm is used to generate Nafion chains with random growth paths and random starting points. A significant difference with the random model in [2] is that we do not build our systems at a density close to the final one. In order not to start with too much entangled chains, the systems are initially built at a density below the experimental one. The density after equilibration is again close to the experimental one.

Even though further improvements of the new algorithms can easily be envisaged, we demonstrate here that with the present version several sets of configurations that are compatible with the available scattering data can be generated and equilibrated. Twelve large random Nafion systems are built with different initial positions of the atoms as well as different water contents and side chain lengths (Nafion/Hyflon). They are equilibrated and then simulated for several ten nanoseconds. After equilibration, the structures are, as mentioned, compatible with the experimental scattering data. In addition we study a model similar to the one by Schmidt-Rohr and Chen [3], i.e. the newest morphological model of Nafion. The experimental scattering data are also satisfactorily reproduced with this model, hence, the prolonged debate over the structure of Nafion.

This agreement gives confidence that a more detailed analysis of the so-obtained configurations is scientifically warranted. We characterize and analyze the local, intermediate and large-scale structures by various structural parameters and domain size distributions. We therefore compute, for example, radial distribution functions (rdf), total and

partial structure factors ($S(q)$) as well as numbers and sizes of hydrophilic clusters (depending on the definition of a cluster). The dynamics of various species in the system is also investigated, e.g. via the computation of the mean square displacements (msd) and the self-diffusion coefficients. These simulations are probably at the limit of what can today be achieved with all-atom molecular simulations of the MD type. We hope that this work will advance the ongoing debate on the structure and dynamics of these important materials.

Zusammenfassung

Perfluorierte Membranen werden insbesondere in Polymerelectrolyt-Brennstoffzellen (PEFC) eingesetzt. Das wohlbekannte Ionomer Nafion® (Dupont) ist wegen seiner hohen Protonenbeweglichkeit ein Referenzmaterial für solche Anwendungen in Brennstoffzellen. Die Membran separiert in Wasser oder anderen hydrophilen Lösungsmittel in eine hydrophobe Polymermatrix und eine hydrophile Subphase, die Cluster mit Wasser und Ionen enthält. Dabei vergrößern sich die Ausdehnung der Cluster und ihre Konnektivität mit zunehmendem Wassergehalt [1].

Welche ist die Morphologie des Nafions und die Struktur des Lösungsmittels in diesen Systemen? Es ist jüngst anhand großer simulierter Systeme gezeigt worden [2], dass mehrere morphologische Modelle die experimentellen Streudaten wiedergeben können, was nahelegt, dass solche Streudaten alleine nicht geeignet sind, die wahre Struktur des Nafion aufzudecken. Ein in [2] beschriebenes 'Zufallsmodell', d.h. das einzige der untersuchten Modelle, das keine besondere Anfangsstruktur annahm, konnte die experimentellen Daten allerdings nicht wiedergeben.

In molekularen Computersimulationen Konfigurationen zu erzeugen, die wirklich nicht mehr mit der angenommenen Anfangskonfiguration korreliert sind, bleibt eine echte Herausforderung. Die erreichbaren Zeitskalen sind zu kurz, um eine signifikante Bewegung des Polymers (z.B Konformationsänderungen, Faltungen, usw.) zuzulassen. In dieser Arbeit wird daher ein neues Zufallsmodell für Nafion vorgestellt. Ein neuentwickelter Algorithmus erzeugt Nafionketten mit zufälligem Wachstumspfad ausgehend von zufälligen Anfangspunkten. Ein signifikanter Unterschied zu dem Zufallsmodell von [2] ist, dass hier nicht versucht wird, die Systeme bei einer Dichte vergleichbar der experimentellen Dichte aufzubauen. Anstattdessen werden die Systeme, um alzu starkes Verknäuelung zu vermeiden, anfangs bei einer deutlich kleineren Dichte erzeugt. Nach Äquilibrierung ist die Systemdichte wieder in etwa gleich der experimentellen.

Wiewohl weitere Verbesserungen des neu Algorithmuses leicht ins Auge gefaßt werden können, so kann hier doch gezeigt werden, dass mit der gegenwärtigen Version Konfigurationen erzeugt und äquilibriert werden können, die mit den verfügbaren Streudaten kompatibel sind. Zwölf große Nafion Zufallssysteme, mit verschiedenen Anfangspositionen der Atome, verschiedenem Wassergehalt und Längen der Seitenketten (Nafion/Hyflon) werden aufgebaut. Diese werden äquilibriert und mehrere zehn Nanosekunden lang simuliert. Nach der Äquilibrierung sind die Strukturen, wie erwähnt, kompatibel mit den experimentellen Streudaten. Weiterhin wird ein Modell ähnlich dem von Schmidt-Rohr und Chen [3], d.h. dem neuesten morphologischen Modell für Nafion, studiert. Auch hier werden die experimentellen Streudaten zufriedenstellend wiedergegeben, daher die weiterhin bestehende Debatte über die Struktur des Nafion.

Die gefundenen Übereinstimmungen lassen darauf vertrauen, dass eine detaillierte Analyse der simulierten Konfigurationen wissenschaftlich sinnvoll ist. So wird die Struktur der Systeme auf verschiedenen Längenskalen charakterisiert, zum Beispiel durch radiale Paarverteilungsfunktionen (rdf), totale und partielle Strukturfaktoren ($S(q)$) sowie Anzahl- und Größenverteilungen hydrophiler Cluster (abhängig von der Definition eines Clusters). Die Dynamik einzelner Spezies im System wird ebenfalls untersucht, zum Beispiel durch die Berechnung der mittleren quadratischen Verschiebungen (msd) und der Selbstdiffusionskoeffizienten. Diese Simulationen sind wahrscheinlich an der Grenze dessen, was heute mit 'all-atom' molekularen MD-Simulationen möglich ist. Ich vertraue darauf, dass diese Arbeit dennoch einen Fortschritt in der aktuellen Debatte über die Struktur und Dynamik dieser wichtigen Materiale darstellt.

Résumé

Les membranes fluorées sont utilisées en particulier dans les dénommées piles à combustible à membrane électrolyte polymère. Gràce à sa grande mobilité en protons, le célebre ionomer Nafion® (Dupont) est un matériau de référence pour les applications liées aux piles à combustible. En présence d'eau ou d'autres solvants hydrophiles la membrane se sépare en une matrice polymérique hydrophobe et une sous-phase aqueuse contenant des clusters d'eau et ions, dont les tailles et la connectivité augmente quand la quantité d'eau augmente [1].

Quelle est la morphologie du Nafion et la structure du solvant, dans de tels systèmes? Il a été récemment montré [2] sur des simulations de large systèmes que plusieurs modèles morphologiques reproduisent les données expérimentales de diffusion, évoquant l'incapacité des mesures de diffusion seules à élucider la véritable structure du Nafion. Néanmoins, un modèle 'aléatoire' décrit dans [2], c'est à dire l'unique modèle étudié sans présumer d'une structure initiale particulière, n'a pas pu reproduire les données expérimentales.

Générer en simulations moléculaires des configurations du système qui soient vraiment décorrélées de la configuration initiale reste un vrai défi statistique. Les échelles de temps réalisables ne permettent simplement pas d'obtenir des mouvements significatifs du polymère (comme des transitions de conformations, repliements de chaînes, etc.). Nous proposons ainsi dans cette étude un nouveau modèle de Nafion à morphologie aléatoire. Un algorithme récemment développé est utilisée pour générer des chaînes de Nafion avec des chemins et des points de départ aléatoires. Une différence majeure avec le modèle aléatoire dans [2] est que nous ne construisons pas nos systèmes à une densité proche de la densité finale. Pour ne pas démarrer avec des chaînes trop enchevêtrées, les systèmes sont initialement préparés à une densité en dessous de la référence expérimentale. La densité aprés équilibration est de nouveau proche de l'éxperience.

Bien qu'il soit facilement envisageable d'améliorer les nouveaux algorithmes, nous démontrons ici qu'avec la présente version plusieurs séries de configurations compatibles avec les donées expérimentales de diffusion disponibles peuvent être générées et équilibrées. Douze large systèmes de Nafion à morphologie aléatoire sont construits avec des positions initiales des atomes ainsi que des quantités d'eau et des longueurs de chaînes (Nafion/Hyflon) différentes. Ils sont équilibrés puis simulés sur plusieurs dizaines de nanosecondes. Après équilibration, les structures sont, comme indiqué ci-dessus, compatibles avec les données éxperimentales de diffusion. En plus nous étudions un modèle ressemblant à celui de Schmidt-Rohr and Chen [3], c'est à dire le plus récent modèle morphologique. Avec ce modèle, les données expérimentales sont également reproduites de manière satisfaisante, d'où la prolongation du débat sur la structure du Nafion.

La cohésion entre les valeurs calculées et celles mesurées experimentalement incite à des analyses plus en détails de ces configurations obtenues. Nous caractérisons et analysons les structures locales, intermédiaires et à grande échelle avec divers paramètres structuraux et distributions des tailles de domaines. Nous calculons donc, par exemple, des fonctions de distribution radiale (rdf), des facteurs de structure ($S(q)$) totaux et partiels tout comme des nombres et des tailles de clusters hydrophiles (selon la définition d'un cluster). La dynamique de diverses espèces dans le système est également examinée, par exemple au travers des déplacements carrés moyens (msd) et des coefficients de diffusion. Ces simulations sont probablement à la limite de ce qui est réalisable aujourd'hui avec des simulations 'full-atom' du type MD. Nous espérons que ce travail fera avancer le débat sur la structure et la dynamique de ces matériaux importants.

Contents

1 Introduction **20**

2 State of the Art **26**

2.1 General Aspects 26

2.2 Experimental Studies 27

2.2.1 Nafion 28

2.2.2 Short side chain membranes (e.g. Hyflon) 34

2.3 Computational Studies 36

2.3.1 Small Nafion Systems 36

2.3.2 Large Nafion Systems 39

2.3.3 A Hyflon Simulation 40

3 Simulation method **43**

3.1 General principles 43

3.1.1 Models 44

3.1.2 The MD method 47

3.1.3 Applying statistical mechanics 49

3.2 Practical aspects 50

3.2.1 Simulating bulk phases 50

3.2.2 Short-range and long-range interactions 51

3.2.3 Initialization 52

3.2.4 Thermodynamical ensembles 53

3.2.5 Time step 53

4 Data Analysis **55**

4.1 Thermodynamics . . . 55

4.2 Estimating Errors . . . 55

4.3 Vibrational Analysis . . . 56

4.4 Structure . . . 56

4.5 Dynamics . . . 59

5 Models **60**

5.1 Morphology . . . 60

5.2 Systems, Force Field, and Simulation Details . . . 62

5.2.1 Systems . . . 62

5.2.2 Force Field . . . 65

5.2.3 Simulation Details . . . 65

5.3 Initial Geometries . . . 66

5.3.1 Random Model . . . 66

5.3.2 Cylindrical Model . . . 71

5.4 Equilibration and Data Collection . . . 73

5.A Appendix: Force Field Parameters . . . 75

5.B Appendix: Algorithm and Codes for the Initial Geometries . . . 76

5.C Appendix: Densities . . . 93

6 Results and Discussions **94**

6.1 Structure . . . 94

6.1.1 VMD Snapshots . . . 94

6.1.2 Radial Distribution Functions and Coordination Numbers . . . 95

6.1.3 Structure Factors 102

6.1.4 End-to-End Distances 108

6.1.5 Cluster Analysis 111

6.2 Dynamics 127

6.2.1 VMD Snapshots 127

6.2.2 Structure Factors 129

6.2.3 Mean Square Displacements 132

6.2.4 Cluster Analysis 138

6.3 Vibrational Analysis 139

6.3.1 Fourier Transformation of $\mathbf{c_{vv}}$ 139

6.3.2 Extraction of the Normal Modes 145

6.A Appendix: VMD Snapshots 148

6.B Appendix: Radial Distribution Functions 154

6.C Appendix: Structure Factors 156

6.D Appendix: End-to-End Distances 157

6.E Appendix: Cluster Analysis 158

6.F Appendix: Mean Square Displacements 161

6.G Appendix: Vibrational Analysis 162

7 Conclusions and Outlook **163**

8 Bibliography **166**

9 Curriculum Vitae **182**

10 'Erklärung' **183**

11 Acknowledgments **184**

List of Symbols

N	[-]	number of particles
M	[-]	number of time steps (can also be the number of time origins)
t, τ	[ps]	times
δt	[ps]	integration time step
$\mathbf{F}$	$[\mathrm{kJ} \cdot \mathrm{mol}^{-1} \cdot \mathrm{nm}^{-1}]$	vector force
m	[kg]	mass
$\mathbf{r}$	[nm]	vector position
$\dot{\mathbf{r}}, \mathbf{v}$	$[\mathrm{nm} \cdot \mathrm{s}^{-1}]$	vector velocity
$\ddot{\mathbf{r}}, \mathbf{a}$	$[\mathrm{nm} \cdot \mathrm{s}^{-2}]$	vector acceleration
∂	[-]	partial derivative
∇	[-]	gradient
p	$[\mathrm{kJ} \cdot \mathrm{mol}^{-1} \cdot \mathrm{nm}^{-3}]$	pressure
T	[K]	temperature
V	$[\mathrm{nm}^3]$	volume
E	$[\mathrm{kJ} \cdot \mathrm{mol}^{-1}]$	total energy
λ	$[\mathrm{H_2O/SO_3^-}]$	water content (can also be the wavelength [nm])
$\mathcal{H}$	$[\mathrm{kJ} \cdot \mathrm{mol}^{-1}]$	Hamiltonian
$\mathcal{K}$	$[\mathrm{kJ} \cdot \mathrm{mol}^{-1}]$	Kinetic energy
$\mathcal{V}$	$[\mathrm{kJ} \cdot \mathrm{mol}^{-1}]$	Potential energy
$\mathbf{q}$	[nm]	set of coordinates
$\mathbf{p}$	$[\mathrm{kg} \cdot \mathrm{nm} \cdot \mathrm{s}^{-1}]$	set of momenta
ε	$[\mathrm{kJ} \cdot \mathrm{mol}^{-1}]$	depth in the potential well
σ	[nm]	distance at which the interparticle potential is zero (can also be the root mean square deviation)
r, d	[nm]	position, distance
v	$[\mathrm{nm} \cdot \mathrm{s}^{-1}]$	velocity
q	$[e]$	charge (can also be the scattering vector $[\mathrm{nm}^{-1}]$)
ϵ_0	$[\mathrm{kJ}^{-1} \cdot \mathrm{mol} \cdot \mathrm{nm}^{-1} \cdot \mathrm{e}^2]$	dielectric permittivity
α, ϕ	[degree, radian]	angles
k_b	$[\mathrm{kJ} \cdot \mathrm{mol}^{-1} \cdot \mathrm{nm}^{-2}]$	bond force constant
k_α	$[\mathrm{kJ} \cdot \mathrm{mol}^{-1} \cdot \mathrm{rad}^{-2}]$	angle force constant
k_ϕ	$[\mathrm{kJ} \cdot \mathrm{mol}^{-1} \cdot \mathrm{rad}^{-2}]$	dihedral angle force constant
$< >$	[-]	average
$c_{\mathrm{vv}}(t)$	[-]	velocity autocorrelation function (normalized to 1)
D	$[\mathrm{nm}^2 \cdot \mathrm{s}^{-1}]$	self-diffusion coefficient
ω	$[ps^{-1}]$	frequency
$F(\omega)$	[arb. units]	spectral density of motions
r_c	[nm]	cut-off radius
r_{list}	[nm]	Verlet neighbor list radius
r_{cyl}	[nm]	cylinder radius
$g(r), n(r)$	[-]	radial distribution function (rdf), and its integral

ρ	$[\mathrm{kg} \cdot \mathrm{nm}^{-3}]$	density
$S(q)$	[arb. units]	structure factor
si	[-]	statistical inefficiency
msd(t)	$[\mathrm{nm}^2]$	mean square displacement function
$d_{\mathrm{n-n}}$	[nm]	end-to-end distance

List of Tables

1 Simulation systems. 63

2 Lennard-Jones parameters and partial charges. 75

3 Stretch parameters. 75

4 Bend parameters. 75

5 Self-diffusion coefficients, for all systems. 134

List of Figures

1 The Electrovan from General Motors. 21

2 Different kinds of fuel cell vehicles. 21

3 A PEM fuel cell scheme. 22

4 Nafion ionomer unit and Nafion membrane schemes. 23

5 Schematic representation of the Nafion model proposed by Gierke et al. in 1981. The image has been taken from [4]. 29

6 Periodic boundary condition scheme. 51

7 'Basic' clustering algorithm scheme. 57

8 'Advanced' clustering algorithm scheme. 58

9 Random and cylindrical morphological models of Nafion. 60

10 Simulation performance test. 64

11 Random growth process of the CF_2 backbones and side chains. 68

12 Structures of a Nafion chain before and after energy minimization. . . . 69

13 200 Nafion chains in a $20 \times 20 \times 20$ nm^3 cube. 70

14 Random distribution of 40000 water molecules and 4000 K^+ in a system containing 200 polymer chains. 71

15 Small cylindrical system (8 Nafion chains in a $6 \times 6 \times 6$ nm^3 box). . . . 72

16 Initial positions of the polymer in the cylindrical system (216 Nafion chains in a $18 \times 18 \times 18$ nm^3 box). 72

17 Total density vs. the simulation time, for various systems. 74

18 algorithm for the growth process of a Nafion/Hyflon chain 76

19 Total density vs. the simulation time, for all systems. 93

20 NR-10-1 system at t=40 ns. Only the K$^+$-ions are shown. 94

21 Average number of S-atoms closer than 0.8 nm to an S-atom vs. the simulation time, for various systems. 96

22 $g_{\mathrm{S-S}}(r)$, for various systems. 96

23 Number of S-clusters vs. the simulation time, for various systems. . . . 98

24 Distribution of the number of S-atoms per cluster, for various systems. 98

25 $g_{\mathrm{S-Ow}}(r)$ and $n_{\mathrm{S-Ow}}(r)$, for various systems. 99

26 $g_{\mathrm{Ow-Ow}}(r)$ and $n_{\mathrm{Ow-Ow}}(r)$, for various systems. 101

27 S(q) (neutron) of the last configuration of various systems. 103

28 S(q) (X-ray) of the last configuration of various systems. 103

29 S(q) (neutron), for the last configuration and the last 10 ns of the NC-10-1 simulation. 104

30 S(q) (X-ray), for the last configuration and the last 10 ns of the NC-10-1 simulation. 104

31 Small-angle X-ray scattering curve obtained from the random morphological model after equilibration and from experiment [5]. 106

32 Small-angle X-ray scattering curve obtained from the random morphological model at t=0 (i.e. before equilibration). 106

33 Partial $S_{\alpha\beta}(q)$s for different pairs, for the NR-10-1 system. 107

34 Partial $S_{\alpha\beta}(q)$s for different pairs, for the NC-10-1 system. 107

35 Average polymer chains end-to-end distance vs. the simulation time, for various systems. 109

36 Average side chain end-to-end distance vs. the simulation time, for the Nafion and Hyflon systems. 110

37 Number of aqueous clusters vs. the simulation time, for various systems. 112

38 Number of atoms (in % of total) in the largest aqueous cluster vs. the simulation time, for various systems. 112

39 Histogram of the number of atoms per cluster in the NR-10-1 system, averaged over 10 ns. 113

40 Number of atoms (in % of total) in the largest aqueous cluster vs. the simulation time, for various selections of atom types. 114

41 Effect of changing the value of the distance criterion on the number of aqueous clusters. 115

42 Effect of changing the value of the distance criterion on the number of atoms in the largest aqueous cluster. 115

43 Histogram of the number of atoms per cluster in the NR-10-1 and HR-10-1 systems, averaged over 10 ns. 117

44 Histogram of the number of atoms per cluster in the NC-10-1 and NR-10-1 systems, averaged over 10 ns. 117

45 Histogram of the number of atoms per cluster in the NC-10-1 and NR-10-1 systems, at t=0. 118

46 Population of the largest cluster vs. λ and the distance criterion, d. . . 119

47 Number of subclusters, after iterative removal of atoms with up to two neighbors, vs. the simulation time, for various systems. 120

48 Number of atoms in the largest subcluster, after iterative removal of atoms with up to two neighbors, vs. the simulation time, for various systems. 121

49 Number of subclusters, after iterative removal of atoms in the largest cluster with up to three neighbors, vs. the simulation time, for all systems.122

50 Number of atoms in the largest subcluster, after iterative removal of atoms in the largest cluster with up to three neighbors, vs. the simulation time, for all systems. 122

51 Spatial distribution of the Ow-atoms and K^+-ions in the largest cluster, before and after removing atoms with up to two and three neighbors. . 123

52 Number of subclusters and number of atoms in the largest subcluster, after iterative removal of atoms with up to two neighbors, vs. λ. 124

53 Probabilities of the number of atoms per subcluster in the NR-10-1 and NR-5-1 systems. 124

54 Probabilities of the number of atoms per subcluster in the NR-10-1 and NC-10-1 systems. 125

55 Pictures of the simulation of the NR-10-1 system, taken every 5 ns. . . 127

56 S(q) (neutron), computed every 5 ns of the NR-10-1 system. 129

57 S(q) (X-ray), computed every 5 ns of the NR-10-1 system. 130

58 S(q) (neutron), computed every 5 ns of the NR-10-2 system. 130

59 S(q) (X-ray), computed every 5 ns of the NR-10-2 system. 131

60 Mean square displacement of the water molecule oxygens (Ow), for various systems. 133

61 Mean square displacement of the K^+-ions, for various systems. 133

62 Self-diffusion coefficient of the Ow-atoms and K^+-ions vs. λ. 134

63 Mean square displacement of the S-atoms, for all systems. 136

64 Mean square displacement of the C_9-atoms, for all systems. 136

65 Histogram of the number of Ow-atoms per cluster in the NR-10-1 system at t=19.5 ns and 19.8 ns. 138

66 Histogram of the number of Ow-atoms per cluster in the NR-10-1 system at t=20.0 ns and 20.5 ns. 138

67 Velocity autocorrelation function c_{vv} for the flexible and rigid Nafion molecule in a small Nafion/Water system. 140

68 Power spectrum for the flexible and rigid Nafion molecule in a small Nafion/Water system. 140

69 Power spectrum for the Ow- and Hw-atoms in a small Nafion/Water system. 141

70 Time evolution of the total energy of a small Nafion/Water system, using different δt-values. 143

71 Time evolution of the x-coordinate of a given Ow-atom in a small Nafion/Water system, for different δt-values. 144

72 Deviation from the 'correct' trajectory of a small Nafion/Water system. 144

73 Drift coefficients of the total energy vs. δt. 145

74 Eigenfrequencies obtained from the extraction of the normal modes of a Nafion molecule. 146

75 Eigenfrequencies obtained from the extraction of the normal modes of a Nafion molecule, using constant forces for the bonds 10 times larger. . . 147

76 NR-10-1 system at t=20 ns. Only K^+-ions are shown. 148

77 NR-10-1 system at t=30 ns. Only K^+-ions are shown. 148

78 NR-7.5-1 system at t=15 ns. Only K^+-ions are shown. 149

79 NR-5-1 system at t=15 ns. Only K^+-ions are shown. 149

80 NC-10-1 system at t=15 ns. Only K^+-ions are shown. 150

81 HR-10-1 system at t=20 ns. Only K^+-ions are shown. 150

82 Pictures of the simulation of the NR-10-3 system, taken every 5 ns. . . 151

83 Pictures of the simulation of the NR-10-6 system, taken every 5 ns. . . 151

84 Pictures of the simulation of the NR-7.5-1 system, taken every 5 ns. . . 152

85 Pictures of the simulation of the NC-10-1 system, taken every 5 ns. . . 152

86 Pictures of the simulation of the HR-10-1 system, taken every 5 ns . . . 153

87 Average number of S-atoms closer than 8 Å of each S-atom vs. the simulation time, for all systems. 154

88 g_{S-S}, for all systems. 154

89 $g_{\mathrm{S-Ow}}(r)$, for all systems. 155

90 $g_{\mathrm{Ow-Ow}}(r)$, for all systems. 155

91 S(q) (neutron), computed for the last configuration, for all systems. . . 156

92 S(q) (X-ray), computed for the last configuration, for all systems. . . . 156

93 End-to-end distance vs. the simulation time, for all systems. 157

94 Number of aqueous clusters vs. the simulation time, for all systems. . . 158

95 Number of atoms in the largest aqueous cluster vs. the simulation time, for all systems. 158

96 Number of aqueous clusters vs. the simulation time, for all systems. . . 159

97 Number of atoms in the largest aqueous cluster vs. the simulation time, for all systems. 159

98 Number of subclusters, after iterative removal of atoms with up to two neighbors, vs. the simulation time, for all systems. 160

99 Number of atoms in the largest subcluster, after iterative removal of atoms with up to two neighbors, vs. the simulation time, for all systems. 160

100 Mean square displacement of the water molecule oxygens (Ow), for all systems. 161

101 Mean square displacement of the K^+-ions, for all systems. 161

102 Power spectrum of the K^+-ions in a small Nafion/Water system. 162

1 Introduction

Terms like 'alternative energy' and 'environmental protection' nowadays belong to the standard vocabulary in science and technology, and also in politics and business. A vast variety of applications, such as generating power for vehicles and houses, is involved. Over the last decades, great efforts have been deployed by scientists and industrialists on the common quest for an alternative to the depleting natural source of fossil fuel. For the design of an innovative technology, the quantity of energy that can be delivered (or more precisely: transformed), the cost, and the impact on the environment (especially the quantity of emission of CO_2) are of major concern. Legitimately, one expects such a technology to be competitive over the whole range of applications of the energy conversion device market: transportation (in other terms the automotive market), stationary power generation, and portable applications (like cellular cell phones, portable computers, auxiliary power unit in cars, etc.).

Fuel cells are considered to be one of the most promising electrical energy conversion devices. As early as 1839, the basic operating principle of fuel cells was discovered by C. F. Schoenbein and W. R. Grove by reversing water electrolysis to generate electricity from hydrogen and oxygen [6–8]. At that time the device was called 'gaseous voltaic battery'. "A fuel cell is an electrochemical 'device' that continuously converts chemical energy into electric energy (and some heat) for as long as fuel and oxidant are supplied" [9]. Their main advantage over other devices is the high energy-to-mass ratio that can be achieved. Also they are relatively environmentally safe, easy to use and to transport, and they are not affected by the economy of fossil fuels. What remains, however, is their high cost. Since 1950, fuel cells (especially low- and intermediate-temperature fuel cells) have nevertheless been involved in a wide range of applications; presently their major source of applications is the automotive industry.

In 1966, the first fuel cell vehicle on the road by the General Motors Company was ready: the Electrovan (see fig. 1). It had the capacity to transport two passengers at a top speed of $110\ \mathrm{km{\cdot}h^{-1}}$ and over a distance of 240 km. However, the start-up procedure for the Electrovan took about three hours [10]. In 2001, five new fuel cell cars from the manufacturer Toyota were able to transport five passengers each at a maximum speed of $150\ \mathrm{km \cdot h^{-1}}$ over a maximum distance of 250 km and without a long start-up procedure. During the road tests in Japan, these vehicles have accumulated more than 3000 miles in total [10]. Until now, fuel cell vehicles have not been commercialized. There is still some progress needed for their usage in 'every-day life', notably concerning the size, the weight, the distance that can be traveled before refueling and, of course, the price. However, there are prototypes submitted to road tests for almost all modes of transport such as motorcycles, cars, buses, boats, planes, etc. (see fig. 2). Altogether, more than 3 million miles have been already traveled by fuel cell vehicles [10]. Some manufacturers announced the introduction of a production model with a reasonable price in the coming years [11].

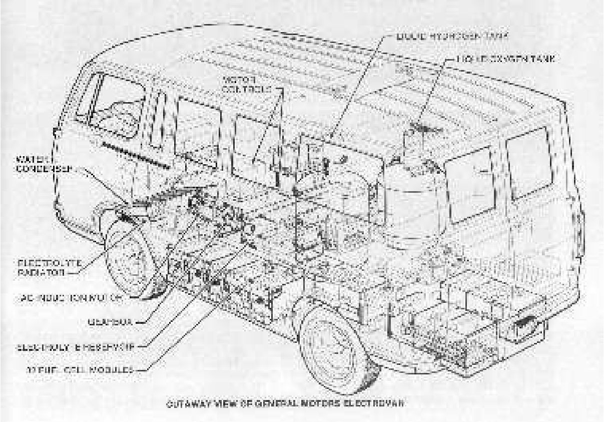

LIQUID HYDROGEN TANK
LIQUID OXYGEN TANK
MOTOR CONTROLS
WATER CONDENSER
ELECTROLYTE RADIATOR
AC INDUCTION MOTOR
GEARBOX
ELECTROLYTE RESERVOIR
32 FUEL CELL MODULES
CUTAWAY VIEW OF GENERAL MOTORS ELECTROVAN

Figure 1: The Electrovan from General Motors. Image taken from [12].

Figure 2: Different kinds of fuel cell vehicles. Images taken from [11].

There are many different types of fuel cells, classified by the type of electrolyte employed. Proton-conducting materials are used as the electrolyte for low- and intermediate-temperature fuel cells. Examples are, among others, phosphoric acid and polymer electrolyte membranes (**PEM**), also called proton exchange membranes. The way these fuel cells work remains the same since Schoenbein and Grove's discovery in 1839: energy from chemical reactions is converted into electrical energy, producing direct current electricity, which is fed to an external circuit. Typically, a fuel cell consists of two metallic segments, called electrodes, separated by the electrolyte and connected by a conducting wire. Oxidation and reduction reactions occur at the electrodes and ions diffuse through the electrolyte medium only, whereas the freed electrons flow through the wire, producing the electric current. Ideally, ion transport through such electrolytes must be fast and highly selective; these two properties are, however, frequently at odds with each others [13]. For some reasons (see below), much on the current research is on PEM fuel cell (see a scheme of a typical PEM fuel cell in fig. 3). More general information about fuel cells, the fundamentals, technologies and applications can be found e.g. in [9–11, 13–16].

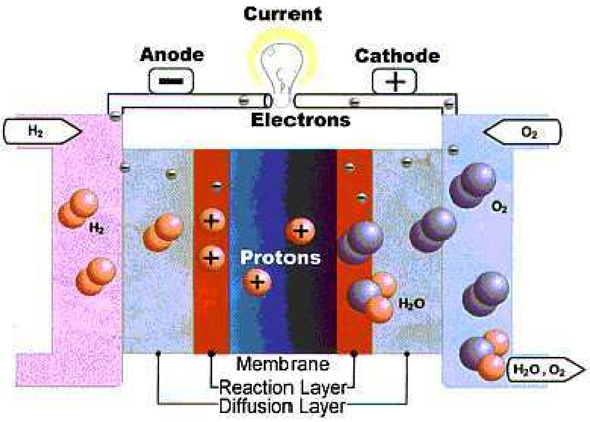


Figure 3: A PEM fuel cell scheme, taken from [17]. Overall reaction : $H_2 + 1/2\ O_2 \rightarrow H_2O$

Perfluorinated ionomers are prominently involved in fuel cell technology as the electrolyte medium in PEM fuel cells. They are the result of the copolymerisation of non-polar tetrafluoroethylene monomers $CF_2 = CF_2$ (the monomeric unit of Teflon), and polar perfluorosulfonic vinyl ether monomers, $CF_2 = {}^*CF(O - CF_2 - {}^*CF(CF_3) - O - CF_2 - CF_2 - SO_3H)$, where $^*C$ denotes a chiral center. Nafion$^{\circledR}$ (a registered trademark by E.I. DuPont de Nemours) is such an ionomer. It is composed of a CF_2 backbone with pendant sulfonated side chains, which are able to induce, in presence of water, an hydrophobic / hydrophilic phase separation at the nanometer scale [1, 4] (see fig. 4). The sulfonic acid functional groups aggregate to form a hydrophilic domain that is hydrated upon absorption of water. It is within this continuous domain that ionic conductivity occurs. When increasing the water content, the membrane swells, the phase separation becomes more pronounced, and the connectivity between the hydrophilic domains increases until only one but large domain can be distinguished, i.e. until the 'water percolation threshold' is reached [1, 18]. With enough surrounding water molecules, the protons can dissociate from their anionic SO_3^--groups and can diffuse or shuttle through the hydrogen bond network via a 'hop-turn' mechanism of bond formation and cleavage, as it was observed e.g. in [19–22]. This shuttling mechanism was firstly suggested by Grotthuss in 1806 [23], validated in 1995 [24] and refined in 2004 [25].

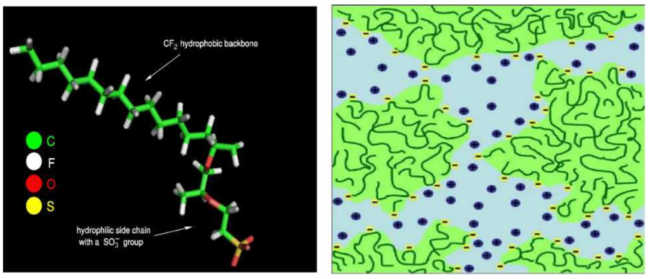

CF_2 hydrophobic backbone
C
F
O
S
hydrophilic side chain with a SO_3^- group

Figure 4: Left: Nafion ionomer unit (2 nm). Hydrophilic side chains are terminated by SO_3^- groups. Right: Schematic view of a Nafion membrane (12 nm), image taken from [26].

Because high proton conductivity is only obtained at high levels of hydration, the maximum operation temperature is limited to, approximately, the condensation point of water . Another drawback of PEM fuel cells is that any protonic current also leads to transport of water through the membrane (as a result of electroosmotic drag) and, if methanol dissolves in the membrane, this is transported virtually at the same rate [13]. The limited operating temperature and the acidity of the electrolyte makes it necessary to use platinum or platinum alloys (the most active but also the most expensive electrocatalyst) to promote the electrochemical reactions in the anode and cathode structures. However, even with platinum, only rather pure hydrogen can be oxidized at sufficient rates. The humidification requirements, along with the high electro-osmotic drag of water and methanol in conventional membranes, complicate the water and heat management of the fuel cell and lead to a significant chemical short-circuiting, i.e., parasitic chemical oxidation of methanol at the cathode. Therefore, tremendous engineering efforts has been expended to at least control the fluxes of water and methanol in such a way that the resulting transient and steady-state concentration profiles of these species across the membrane still permit acceptable function of the membrane and the electrode structures [13].

Because of the high proton mobility, and because of their stability, Nafion and similar ionomers (e.g. the short-side chain ionomer, also called Hyflon) have been up to now considered the reference material with which most experimental and theoretical results have been compared. Although the nanoscale phase separation and the organization into hydrophilic domains or clusters within the hydrophobic matrix is generally accepted, a debate concerning their nanoscale morphology, where unambiguous observations by experiment are missing, persists. Unfortunately, the numerous scattering data collected over the last years using a variety of scattering methods cannot provide a direct answer. Rather simple morphological models involving specific assumptions are needed for their interpretation. Numerous morphological models have thus been proposed by experimentalists, from the historical hard-sphere model of Gierke et al. [4] to, for instance, a more recent parallel cylindrical water nanochannel picture [3]. For many reasons, none of them has been able to offer a satisfactory agreement up to now.

The investigation of such a morphological model via computer simulations remains particularly computationally challenging as it requires very large systems and very long simulations. The systems should be large enough in order to observe several of these hydrophilic domains. Many molecular simulations of Nafion have been performed at small and intermediate length scale (typically up to a few nanometers), as it can be found in e.g. [19, 20, 27–34], but as far as we know very rarely at sufficient length scale and resolution required to investigate the nanoscale morphology. Recently, Voth and Knox [2] have supplied not less than half a dozen different domain shapes, ranging from spheres to cylinders through rods and slabs, to very large-scale computer simulations: a few nanoseconds of 2 million atoms in $30 \times 30 \times 30$ nm^3 boxes. As a result, almost all of the models were found to exhibit scattering profiles in agreement with the experiment. We will come back in more details to the morphological characterization and the models that have been proposed over the years by experimentalists and simulators in the chapter 'State of the Art' (chapter 2) of this thesis. A better knowledge and understanding of the microstructure and nanoscale morphology thus seems desirable for the design of the properties of fuel cells in general. Furthermore, the Nafion polymer is a solid at room temperature and pressure. Many dynamical processes are thus very slow and very little information on its dynamics has been available up to now.

Computer simulations are evidently a tool capable of making the link between the microscopic and the macroscopic scale. It is well known that at the level of atoms and molecules, quantum mechanics is the adequate description. However, for very large and complex systems like the present ones, these approaches become computationally very expensive, if not unfeasible. Therefore approximations based from the results of the calculations performed at a high level of accuracy are done incrementally to yield a consistent, but simplified description of the system, compatible with statistical mechanics methods which are based on classical mechanics and which are necessary to reproduce the thermodynamic properties of such large and complex systems.

Since the late 60's and the emergence of powerful computers, a vast range of information related to complex systems, for example liquids and their chaotic Brownian diffusion, have been obtained by simulation methods. The most prominent of these methods are the **MONTE CARLO (MC)** method, the principles of which have been developed by Metropolis et al. in 1953 [35], and the **MOLECULAR DYNAMICS (MD)** method proposed by Alder and Wainwright in 1959 [36]. The big advantage of MD over MC is the availability of the time-dependency. Whether or not the time scale of the mechanics of interest in the system is achievable with the available computational resources is a different question.

A few references in the history of computer simulations show the progress in the last decades. First MC and MD simulations of liquid argon were performed by Wood and Parker in 1957 [37] and by Rahman in 1964 [38]. In 1974 Rahman and Stillinger proposed the first MD simulations of a realistic water model [39]. This is the most abundant and certainly one of the more complex molecular liquid on earth. The first protein simulations appeared in 1977 [40] and nowadays it is common to find in the literature MD simulations on even more complex systems such as solvated polymers, solvated proteins, or phase transition systems.

”The field has been expanding at a tremendous speed and applications of these and other simulation methods have emerged in almost every field of physics, chemistry, the biosciences and even in the engineering sciences. Because of the underlying approach, which is in philosophy often close to the experimental approach, simulations are also sometimes called ’computer experiments’ [41].”

In the present work, we perform new MD experiments on two different morphological models of a hydrated Nafion membrane to investigate and compare their nanoscale morphology and, cursorily, the dynamics: a random model, similar to, but different from, the one proposed by Voth and Knox [2], and a ’cylindrical’ model, similar to the more recent picture by Schmidt-Rohr and Chen [3]. Building the initial configurations for such large and complex systems was the first challenge to overcome. We have tried to generate the polymer chains with as few constraints as possible except the ones inherent in the molecular structures and the size imposed by the required periodicity of our systems (see section 2.2.1). For a better understanding and comparison of these models, we investigate the influence of the water content and of the side chain length. We will describe the systems and the simulations of our computer experiments after a brief ’State of the Art’ of the debate about the Nafion morphology and a brief section to describe the simulation method and the analysis tools that we employed. In total 13 systems have been equilibrated and then simulated over simulation times in the range 15 to 40 nanoseconds. The analyses of our simulation data will be shown and compared with both experimental and simulation results, and the conclusions are at the end.

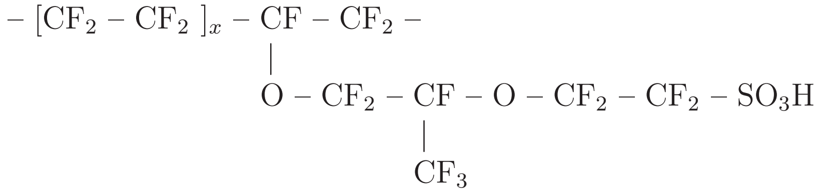
Structural data of the obtained configurations after equilibration together with the solvent are in agreement with the literature: scattering profiles were computed and compared with available experimental data. This agreement encouraged us to perform more detailed analyses of the typical configurations. We characterize and analyze e.g. the different local, intermediate and large-scale structures by various structural parameters and domain size distributions. As it was expected, one of the major difficulties was that there are widely differing relaxation regimes in these systems, with different relaxation times and energies associated: from the picosecond range which governs the motions of individual molecules such as water and ions to the slightly slower mostly reorientational motion of individual groups in the polymer (e.g. in the side chains) up to the range of nanoseconds or even microseconds or seconds that governs the dynamics of the polymeric chains and of the whole system.

2 State of the Art

In this chapter we discuss a selection of both experimental and theoretical studies that have addressed the most fundamental questions concerning the structure and properties of Nafion and, cursorily, similar membranes. This brief review is, for the time up to 2004, in parts based on the work by K. A. Mauritz and R. B. Moore [1], published in that year. We also restrict ourselves to the morphological characterization of the 'conventional' membranes, hence omitting e.g. solutions, recast films, and oriented membranes [1]. Also, other important issues like the thermal and electrochemical stability, the chemical degradation, the electro-osmotic drag, the gas permeation, the methanol crossover, the catalyst incorporation, and so on, are omitted. For a discussion of these issues, see the literature, e.g. [13, 14, 16, 42, 43]. Even restricting ourselves to the characterization of the morphology, the number of publications in the literature is very large and still growing today.

2.1 General Aspects

Nafion® ionomers are the DuPont de Nemours product of the copolymerization of non-polar tetrafluoroethylene (Teflon) $[-CF_2-CF_2-]$ and polar perfluorosulfonic vinyl ether $[-CF_2-CF(O-CF_2-CF(CF_3)-O-CF_2-CF_2-SO_3H)-]$ chain fragments. They have the following chemical structure:



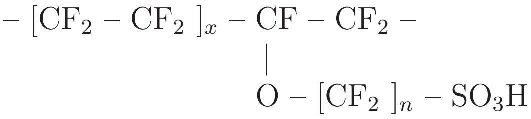
$$\begin{array}{llllllllll}
-[CF_2-CF_2]_x- & CF & -\ CF_2- & & & & & & & \\
 & | & & & & & & & & \\
 & O & -\ CF_2 & - & CF & -\ O & -\ CF_2 & -\ CF_2 & -\ SO_3H & \\
 & & & & | & & & & & \\
 & & & & CF_3 & & & & &
\end{array}$$

where x is the average value resulting from the unknown comonomer sequence distribution. Usually, it is found in the literature that $6 < x < 7$, so that the side chains are said to be separated in average by about 14-15 CF_2 groups. This is in agreement with an average equivalent weight (**EW**) of around 1100, in agreement with the relation [1]:

$$\mathrm{EW} = 100x + 446 \tag{1}$$

EW corresponds to the number of grams of dry Nafion per mole of sulfonic acid groups when the material is in the acid form, it can be measured by acid-base titration, by analysis of atomic sulfur and by FT-IR spectroscopy, as mentioned in [1]. The low EW Nafion polymers (EW < 900) are soluble in many polar solvents whereas the high EW polymers (EW > 1000) are swollen but not soluble in solvents [43]. A 1100 EW holds for the standard commercialized so-called Nafion 117, for which the designation '117' refers to a film having 1100 EW and a thickness of 0.007 in., i.e. 0.18 mm. Similarly, the designation '122' would refer to a film with 1200 EW and a thickness of 0.002 in., i.e. 0.05 mm.

The DuPont Nafion materials are not unique. Their limited range of thermal stability and the requirement of a significant level of hydration have driven the development of similar perfluorinated ionomers [44]: 'Aciplex', 'Flemion', 'Hyflon', and the '3M membrane', different in EW and with different side chains. Nafion, Aciplex and Flemion are so-called 'long-side-chain' (LSC) polymers. The comonomer distribution of and further information on Aciplex and Flemion can be found in [45]. They are not strictly defined in their composition and can be very similar to Nafion. Hyflon and the 3M membrane, in contrast, are more narrowly defined. Compared to Nafion, they are characterized by a shorter side chain, hence a lower EW:



$$\begin{array}{rl} -[\mathrm{CF_2}-\mathrm{CF_2}]_x- & \mathrm{CF}-\mathrm{CF_2}- \\ & | \\ & \mathrm{O}-[\mathrm{CF_2}]_n-\mathrm{SO_3H} \end{array}$$

where n=2 and 4 for Hyflon and the 3M membrane, respectively.

Hyflon ionomers (also called short-side-chain (SSC) membranes) were first synthesized in 1982 by the Dow Chemical Company. Despite the demonstration, by the manufacturer, of significant improvements in fuel cell performance [46–51] using these ionomers, their industrial development was stopped, possibly because the Dow synthesis process [52] was too expensive and too difficult to achieve [53]. Since 2003, they are produced again at a lower cost by Solvay-Solexis, and they have been reinvestigated and reconsidered as a serious competitor to Nafion [54–58]. 3M membranes are more recent [44, 59–61]. We will consider later in this chapter the SSC membranes only, considering the 3M membranes as of intermediate side chain length between the extreme side chain lengths of the LSC and SSC membranes.

2.2 Experimental Studies

X-ray and neutron scattering techniques are particularly suited to study the structure of disordered, or only partly ordered, systems at different length scales [62]. With respect to the morphological characterization of Nafion, a wealth of information has been gathered over the years using a variety of scattering methods. However, the interpretation of these data is inherently indirect and often limited by the necessity of employing rather simple models that involve specific assumptions concerning the structure [1]. Parameters inherent to these assumed model structures are then adjusted to fit the measured scattering functions. Therefore these models cannot be considered to be predictive, but rather as (hopefully reasonable) assumptions that can be calibrated to experimental data. The resolution is usually mesoscopic (domains) rather than molecular. Consequently, a universally accepted morphological model has still to be defined.

2.2.1 Nafion

The first model on which the concept of Nafion morphology is commonly based is the one proposed in 1981 by T. D. Gierke, G. E. Munn and F. C. Wilson [4], which is widely referred as the so-called 'cluster-network model'. Although it is based only on x-ray scattering (wide-angle (WAXD) and small-angle (SAXS)) experiments, it has been considered for many years as the conceptual basis for rationalizing the structure and properties of Nafion membranes. In their experiments, these authors had films of 0.1-0.3 nm thickness and of known equivalent weight EW. They investigated the effects of the EW, the nature of the cation, the temperature, and the water content, on the WAXD and SAXS scans of both unhydrolyzed and hydrolyzed Nafion polymers. From these different experiments, the authors made the following observations and deductions, verbatim from [4]:
– "WAXD scans from unhydrolyzed polymer, $-SO_2F$, reveal some cristallinity...As the EW decreases, the amount of cristallinity detected also decreases."
– "...the degree of cristallinity in the polymer is only moderately altered, compared to the effect of EW, when the polymer is hydrolyzed. Although the cristallinity does decrease somewhat upon hydrolysis, it is clear the hydrolyzed polymer is still partly crystalline."
By increasing the temperature, the degree of cristallinity was found to decrease until almost zero above 270 °C, whereas the melting point of Nafion, determined by differential scanning calorimetry, is about 275 °C. The cristallinity feature was also observed in the SAXS scan, with a signature similar to other polymers that have a CF_2 backbone. Therefore, its origin was attributed to the fluorocarbon matrix CF_2. Applying tensile drawing, the Bragg spacing was found, compared with undrawn samples, larger in the machine direction and smaller in the transverse direction, indicating that the crystallites are oriented and elongated along one direction. Nevertheless, cristallinity was not considered within their model.

Additional findings were [4]:
– "When the polymer is hydrolyzed, a new reflection which corresponds to a Bragg spacing of from 3 to 5 nm is observed in the SAXS scan."
– "The reflection at ca. (*i.e. corresponding to a Bragg spacing of*) 4 nm persists above the melting point of the polymer..."
– "... the Bragg spacing increases with increasing water content."
This demonstrates the capacity of Nafion to swell and to form a polymer / water separation phase with aqueous domains increasing in size with increasing water content.

Accordingly, and unlike the crystalline peak, the intensity of the reflection 'at 4 nm' was found to increase and its position was shifted to larger Bragg spacings with decreasing EW. The intensity of this peak was also found to decrease with increasing the size of the cation, hence the presence of cations within these aqueous domains. However, no significant change in the position was seen when changing the cation, indicating that the morphology is not affected by the nature of the cation.

This reflection is still today the source of much debate on the Nafion morphology. It has been called the 'ionomer peak' and assigned to so-called 'ionic clusters'. The term 'ionic clusters' here refers to nano-phase separated, hydrated ionic aggregates and not to several ions in direct contact with each other.

From their observations and results, Gierke et al. [4] proposed a morphological model "of an approximately spherical, inverted micellar structure." They stated furthermore: "In this model the absorbed water phase separates into approximately spherical domains (*interconnected by narrow rods*) and the ion exchange sites are found near the interface". This is illustrated in fig. 5 (image taken from [4]).

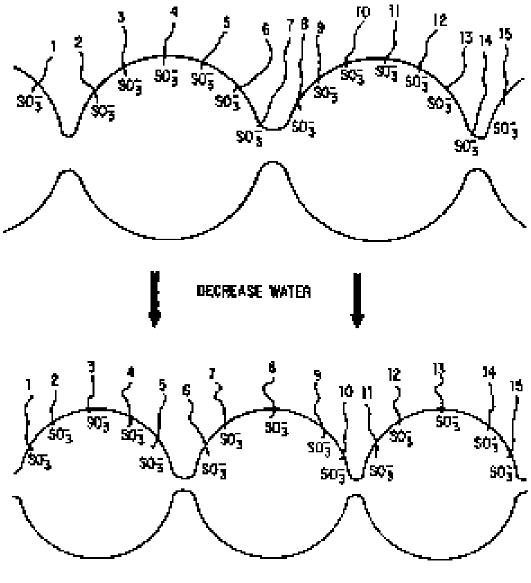


Figure 5: Schematic representation of the Nafion model proposed by Gierke et al. in 1981. The image has been taken from [4].

Within the same year, Roche et al. [63] used both SAXS and small-angle neutron scattering (SANS) to investigate the structure of 1200 EW Nafion membranes in their acid, $-SO_3H$, and salt, $-SO_3Na$, forms in a range of water contents. Similarly to Gierke et al. [4], Roche et al. observed on the SANS curves a first scattering maximum at a corresponding Bragg spacing of about 18 nm for unhydrolyzed and hydrolyzed (with different water contents) samples in both their acid and salt forms. This maximum did not persist in quenched (from 330°C) samples in the salt form and was therefore assigned to interferences between crystalline structures. From WAXD measurements and based on the analysis of the crystalline diffraction ring at 5 Å and the amorphous halo at 4.9 Å, the degree of cristallinity was found to be of the order of 15-20 %. A second maximum on the SANS scan appeared after hydration, at an angle corresponding to a Bragg spacing of the order of 3 to 5 nm. Again in agreement with [4], the intensity of this reflection increased and its position shifted to larger distances with increasing the water content. This was observed on samples in both their acid and salt forms.

Unlike the first reflection attributed to cristallinity, this second reflection remained after quenching, and could thus be analyzed independently of the first one at room temperature. It was thus concluded that the 'upswing' for unquenched samples at low q (q the modulus of the scattering vector) that was not seen for quenched samples is the tail of the first scattering peak arising from cristallinity. Further SAXS curves were obtained for the quenched sample in the salt form in order to highlight this second scattering maximum in a range of water contents. For all hydrolyzed samples, the reflection was observed at Bragg spacings of about 3 to 5 nm while no reflection was observed anymore "within experimental error above a constant background" [63] after the sample was dried at 200°C.

From their observations and results, these authors postulated the coexistence of three distinct phases in hydrated Nafion, namely a CF_2 partly crystalline phase, an amorphous CF_2 matrix phase, and ionic clusters. Such a description presents strong similarities with the one suggested by Yeager and Steck [64]. Furthermore, no significant changes were observed when changing the acid $-SO_3H$ form to the salt $-SO_3Na$ form, supporting the result that the choice of the cation is not of relevance here.

In spite of the fact that their results seemed rather in agreement, Roche et al. did not support the cluster-network model of Gierke et al., arguing that there is no particular evidence for a driving force leading to spherical hydrophilic domains within a stretched and oriented lamellar crystalline phase. At very low water content, one would not expect such a typical arrangement but rather an inhomogeneous distribution of water and ions within the polymer matrix. As an alternative, Roche et al. postulated an 'intracluster' origin of the ionomer peak, for which the maximum Bragg spacing would correspond to characteristic distances between structural elements inside the ionic clusters. The shift observed when changing the water content would thus correspond to changes in the structural dimensions. This is in contrast with the intercluster view of Gierke et al., in which the maximum Bragg spacing corresponds to distances between the centers of the (spherical) ionic clusters, and the shift would thus correspond to a volume change of these clusters. One year later, Roche et al. performed isotopic replacement SANS experiments [65] with different ratios of H_2O/D_2O, using amorphous quenched samples that have no fluorocarbon cristallinity. They found that at high water content, a two-phase model, with the majority of the water molecules ($\geq 60\%$) clustered in one phase and most of the CF_2 units ($\geq 90\%$) in the other phase, separated by a sharp interface, fits scattering data. At low water content, however, the phase separation was not so well defined, and the model did not fit anymore. Later on, Fujimura et al. [66, 67], for example, further discussed the intra or intercluster origin of the ionomer peak.

Since these and other early studies, numerous investigators (e.g., in [3, 45, 68–79]) have attempted to find a model that can reconcile the structural information, obtained with a variety of techniques, with other observed properties, especially transport properties. Examples of models are the 'local order' model by Dreyfus, Gebel et al. [68, 69], a 'lamellar' model by Litt [73], and more recently, a cylindrical model by Schmidt-Rohr et al. [3]; for a detailed description of these 'popular' models, see [1]. Nevertheless, all quantitative estimations that have been reported should not be taken too literally. For example, the degree of cristallinity, the number of crystallites and of 'ionic clusters' (see above), and their relative sizes or geometrical factors, vary from one study to another. In some cases, values determined in different ways may even show an opposite trend. In addition, the properties of a given sample of membrane can change with time during the experiment: the sample may be submitted to modifications that could lead to misinterpretations. It may also be difficult to characterize the sample sufficiently well, e.g. the exact comonomer sequence distribution (x in eq. 1), the EW, or the quantity of water absorbed. Therefore only results and interpretations that have been done on really 'well-defined' materials should be trusted.

Among the very large number of data, results, and models that have been proposed over the years, some are more or less consistent with each other and can thus be thought to give real hints as to the morphology. Others may depend on rather strong, and not always evident, assumptions and should not be taken too literally. In 2008, Schmidt-Rohr and Chen [3], using a different approach, tested the compatibility of several popular morphological models of Nafion found in the literature together with a new model assuming elongated parallel water channels surrounded by side branches and forming a lattice of inverted-micelle cylinders. At 20 volume % of water, the water channels have diameters between 1.8 and 3.5 nm, with an average of 2.4 nm. In this so-called 'parallel water-channel model', Nafion crystallites were included (about 10 volume %) and considered to be elongated and parallel to the water channels with cross-sections of ≈ 5 nm^2.

The predicted scattering profile resulting from the parallel water-channel model was found to 'best' fit the ionomer peak, the small-angle upturn, and also the experimentally observed q^{-1} and q^{-4} power laws in the SAXS data. It can therefore be considered as a universal model of the structure for both unoriented and oriented samples. This model could even explain why the self-diffusion coefficient of water in Nafion is only one order of magnitude smaller than in bulk water and almost one order of magnitude larger than in other sulfonated polymers or ion-exchange resins, at an equivalent water volume fraction: this may be due to the relatively large channel diameters postulated in this view.

These questions will be discussed later from the viewpoint of the present simulations. Thus, according to Schmidt-Rohr and Chen [3] not less than a dozen of other models fail to match the experimental data. However, looking at the curves in [3], other models, e.g. the slab model, the polymer-bundle or a periodic array of water channels, also fit the experimental scattering spectra reasonably well, see also section 6.1.3 below. Furthermore, adjusting parameters like the size of the clusters or the water content should lead to a better compatibility with the experimental spectra. The postulated agreement with the q^{-1} and q^{-4} asymptotic behaviors seems also not really evident.

We note that a comparison between computed and experimentally obtained scattering profiles can be performed both by experimentalists (inverse Fourier transform of the partial structure factors, see eq. 27 in section 4.4) and simulators (Fourier transform of the radial distribution functions (rdf), see section 4.4). The rdfs for the H-H and O-H pairs were thus measured from neutron scattering spectra by Lee et al. [76]. These authors found that the H-H and O-H rdfs in Nafion, hence the local structure of water, are similar to that of bulk water. For complex systems like the ones discussed here, only comparison at the level of the scattering functions, $S(q)$, is possible, see section 6.1.3.

The sizes, geometries and distributions of the hydrophilic domains thus continue to be debated: the effect of water content, equivalent weight, and other features of the scattering profile (especially of the ionomer peak, see above) continue to be examined. It is nonetheless accepted that a peak, called the ionomer peak, is found between 0.1-0.2 Å^{-1}, depending on the water content. In some cases, the scattering intensities at large q scale approximately as q^{-4}, which can be interpreted as a sharp polymer-water interface [1, 69]. In many cases [1], changing the nature of the cation does not change significantly the scattering profiles. One can however discuss the self-diffusion coefficients for the various cations. Even though they differ significantly as to the geometry and spatial distribution of the ionic clusters, all models agree that the ionic groups aggregate in the perfluorinated polymer matrix to form a network of clusters at the nanometer scale that allow for significant swelling by polar solvents and efficient ionic transport [1].

Even though it is clear that the side chains, and particularly the sulfonate groups, play an important role in the links between the hydrophilic domains, the specifics are not well understood. It is known, nonetheless, that sulfonate groups are able to aggregate within the hydrophilic clusters [1]. Furthermore, it is known from theoretical work [21, 22, 30, 80] that the sulfonate groups play a competitive role: on one hand, they enable the conductivity of protons across the polymer membrane, on the other hand they act as proton traps: the mobility of a given proton is higher when it is far from any SO_3^- group.

The bridging connections have thus never been directly observed at the molecular level, either in experiments or in simulations, but have been hypothesized and discussed in the literature [1, 4, 32]. Some of the proposed morphological models have assumed and incorporated such channels or bridges in their design (e.g. the sphere rod-models, the rod models, and many others, see [1]). We shall see in the simulations whether assumed structures without such connecting structural elements (e.g. the cylinder, slab, sphere, random models, see [1]) are stable without the formation of such bridges, which would show how the domains connect and percolate. Although the model proposed for this by Gierke et al. [4] may be oversimplified, it allows the water molecules and ions to be redistributed in the hydrophilic domains (or ionic clusters) as the water network can percolate due to the postulated presence of linking rods between the clusters (see above and fig. 5).

There should furthermore be a particular water content (or range of water contents) at which the transition from several to one cluster occurs. Since a percolated network (i.e. a network of continuously connected clusters) would facilitate the protons to diffuse via the Grotthuss mechanism [19, 20], diffusion coefficients vs. water content curves should display a transition regime. Measuring (self-)diffusion coefficients should thus be a way to find the percolation threshold. For example, based on their diffusion coefficients estimated from nuclear magnetic resonance (NMR) and electrical conductivity, Ochi et al. [81] found values of 0.042 and 0.045 for the water volume fraction (which is a quantity difficult to estimate [1, 81]), respectively, for the percolation threshold. These values are said to be much smaller than those estimated from simple theoretical calculations (see [82] and below).

Also, many sizes and structures are observed for the crystallites in Nafion, see e.g. [3, 70–73, 75, 83]. They depend on the hydration level, the equivalent weight, the unknown comonomer distribution, and also on the various treatments performed on the 'as-received' membrane. It seems undeniable, nonetheless, that the oriented fibrillar or lamellar structures of crystallites found in Teflon and similar polymers are also observed in Nafion. However, this does not entail such strongly defined structures as the cylinders postulated by Schmidt-Rohr and Chen [3]. This remark also applies for the ionic (or hydrophilic) clusters, see above: there is no certainty concerning the different reported shapes (rods, spheres, rod-spheres, cylinders, channels, etc). Although the presence of crystallites in Nafion is clearly demonstrated, some models (e.g., [68, 74]) like the one by Gierke et al. (see above), do not consider the cristallinity feature.

In summary, none of the models derived from scattering experiment closes the debate, suggesting that such data alone is not sufficient. Even though computer simulations hinge on assumptions, and are subject to other limitations, they can provide a description of such large systems at the molecular level [2], if enough computational resources are available. MD simulations are furthermore useful because of the availability of the time-dependence of the system. One can thus examine whether some model fits several sets of experimental data better than another. This has been done on large-scale (30 nm, 2 millions atoms) systems by Voth and Knox [2] and will be discussed below. More coarse-grained approaches have also beeen proposed, see e.g. [84]. Despite the fact that the absolute values for any computed quantity may be questionable unless agreement with experimental values has been found, the trends and the qualitative results from simulations can be trusted and provide a consistency test for several observables.

2.2.2 Short side chain membranes (e.g. Hyflon)

In the mid 1980s, significant improvements in fuel cell performance using short side chains (SSC) ionomers from the Dow Chemical Company were suggested by the manufacturer [46–51] through the 'demonstration' of higher power-generating capability in fuel cells [54]. The fuel cell performance of SSC versus long side chains (LSC, i.e., Nafion, Flemion, and Aciplex) membranes was also studied by Prater et al. [85] from the Ballard Power Systems company, who claimed that, using six-cell stacks, four times the power obtained with a standard Nafion membrane was delivered when using the SSC membrane from Dow [55]. However, several unclarities remain concerning the EW values of the Dow membranes, as mentioned in [54, 55]. This prompted us to include a few simulations of Hyflon in this study and to see whether any significant structural difference appears.

There are more differences between these two classes of material. While all fluorinated polymers exhibit a good thermal and chemical resistance [55], fuel cell systems made of SSC ionomer membranes can operate up to higher temperatures without being damaged, compared with LSC membranes. This is because of their higher glass-transition temperature [54, 55, 86, 87], T_g. (Values for T_g (from dynamic mechanical spectroscopy (DMS)) include: 127 °C for the SSC ionomer and 67 °C for the LSC ionomer [54]. With the same DMS method, Eisman [86] reported $T_g = 165$ and 110 °C for Nafion and Hyflon, respectively). The difference with the values reported by Ghielmi et al. [54] was explained by the large difference in hydration state, as the membranes were highly hydrated in Eisman's work, while dried in nitrogen in Ghielmi's work. On the other hand, Moore et al. [75] suggested two distinct glass transitions temperatures, one for the fluorocarbon matrix and one for the ionic cluster phase. We note also that a higher thermal resistance gives further advantages in terms of cooling and fuel preprocessing (CO content reduction), and consequently in terms of cost [55].

Also, for membranes equivalent in EW, the SSC membranes show a higher degree of cristallinity [54, 55, 75, 87], as measured by WAXD. This is due to the lower molecular weight (MW) side chains and the thus relatively longer CF_2 backbone. Accordingly, the water absorption at a given EW is found significantly lower for the SSC compared to the LSC membranes [54, 55, 75]. Better mechanical properties (due to higher cristallinity) favors the preparation of membranes with lower thickness, which would in turn, according to Ghielmi et al. [55], mean higher membrane conductance and peak power. However, the mechanical properties of such SSC membranes were qualified by these authors as "insufficient" [55]. In that study, it was further stated, verbatim: "However, for the more-crystalline higher-EW polymers, the limited gain in mechanical stability ... is outweighed by a loss in conductivity that is due to the presence of a fraction of volume that is occupied by the non-conducting material".

Alternately, SSC ionomer membranes with low EW (say, 800-900 EW) seem to present certain advantages over other SSC and LSC membranes with a wide range of EW, especially over the Nafion 117 product. In the conclusion of another of Ghielmi's article [54], one can read, verbatim: "Hyflon Ion extruded membranes of EW comprised between 850 and 900 give the best balance of properties for fuel cell applications in terms of dimensional stability, mechanical strength and beginning of life fuel cell performance.". This is in keeping with the higher ionic content and the higher water uptake content (comparing membranes at an equivalent thickness, or when comparing a 850 EW SSC membrane with a 1100 EW Nafion membrane). In contrast with these results of Ghielmi et al., Eisman [86] found that the same conductivity values are found for a 1100 EW Nafion membrane and a 900 EW SSC membrane and that, at equivalent EW, the conductivity is even lower in the SSC membrane. On the other hand, four years later, Eisman [88] suggested an increased fuel cell performance of an 800 EW Dow membrane of 130 μm thickness versus a Nafion 117 (1100 EW, 175 μm thickness).

Despite the difference in the degree of cristallinity, the scattering profiles obtained from SAXS and/or SANS are found almost equivalent for the two types of membranes [72, 75, 89], for sufficiently large q. More specifically, roughly the same shape and peak position, the same swelling behavior (shift of the ionomer peak position to lower q values when increasing water content), and the same asymptotic behavior at large q values (when comparing a 800 EW SSC membrane with a 1100 EW LSC membrane (the standard Nafion 117)). This suggests that the particular structures in SSC membranes are similar to that found in LSC membranes, the structure of both materials being characterized by a phase separation between the water and the polymer, as a consequence of the dominating polymer-interfacial energy. In a study by Halim et al. [89], the same cluster size, yet a different cluster density, was suggested. The comparison from simulation work between SSC and LSC ionomer membranes is discussed below.

2.3 Computational Studies

Since the characteristic length scale of the hydrophilic domains in Nafion membranes is ≈ 2-5 nm, very large systems are required in order to observe several of these domains in molecular simulations, i.e. to investigate the morphology at a nanometer scale. Such large systems can be simulated with MD, depending on the computational resources available. However, most MD simulations of Nafion membrane found in the literature have been performed at length scales too small to investigate the nanoscale morphology [18–22, 27–34, 90–96]. Also, to obtain the dynamics of the governing polymer motions and its influence on the hydrophilic domains remains a challenge that has never been taken up: it would require very long simulation times indeed.

2.3.1 Small Nafion Systems

A typical study is the one by Jang et al. [90]. These authors performed MD simulations to investigate the effect of the monomeric sequence on the nanophase separation and transport properties in Nafion membranes. Therefore, two small systems extremely different in their monomeric sequence (even though these particular sequences can not be achieved with the synthetic techniques available at that time), were compared: a "dispersed" case with the side chains evenly spaced by 14 CF_2 groups and a "blocky" sequence with all the side chains at the end of the backbone (140 CF_2 groups). Both simulated systems were initially composed of four molecules of Nafion 117 at 1150 EW ($x = 7$ in the nomenclature of eq. 1). Each of the four molecules had 10 side chains and there were 560 water molecules (λ= 15), and 40 hydronium ions in the system (in total 4568 atoms). In addition, 8 times larger simulation cells were constructed to try to determine how the finite size influences the nanostructure. No particular geometry was imposed on the distribution of the water and the hydronium molecules around the polymer chains.

After equilibration, the experimental density [97] was achieved for both the blocky and dispersed systems, and data for analysis were collected in the NpT ensemble ($p = 1$ bar, $T = 300$ and 353 K) over 3 ns and 200 ps for the small and the large systems, respectively. As a result, a water/polymer phase segregation was observed in both monomer sequences. All the sulfonate groups were located in the water phase and the network formed by the hydrophilic domains was found to be percolated, which is expected for such a high water content ($\lambda = 15$). However, the degree of segregation was found to be larger in the blocky geometry. This was observed in the S-S and S-Ow (S and Ow are sulfur and water oxygen atoms, respectively) radial distribution functions (rdfs), which indicated that the sulfonate groups in the dispersed case are slightly more hydrated than in the blocky case. This was also supported by the significantly different characteristic dimensions of the hydrophilic clusters that were determined from the computation of the structure factor: $\approx$ 5 nm and $\approx$ 2-3 nm for the blocky and dispersed geometries, respectively. Also, the self-diffusion coefficient of water was found larger in the blocky sequence. This is due to the larger sizes for the hydrophilic domains that were observed in this case.

As another example: Devanathan and Dupuis [18] performed a detailed analysis of water clustering and percolation in configurations of hydrated Nafion 117 membranes generated by MD for $\lambda = 1, 3, 5, 7, 9, 11, 13.5$ and 20. Experimental results [97] show that the maximum water sorption capacity of Nafion corresponds to a λ- value of about 16. Thus, the λ values examined by the authors represent the complete range from the dry to the most hydrated membranes. In this study, the authors used force field parameters similar to the ones used in [90]. The systems were composed of four chains of Nafion (each containing 10 SO_3^- groups) with the side chains evenly spaced (x=7 in eq. 1). No particular geometrical distribution for the water molecules was imposed, and as a function of the water content λ, the number of the water molecules ranges from 0 to 760. 40 H_3O^+ cations were added as counterions of the 40 SO_3^- anions. In addition, similar systems with 8 and 48 Nafion chains were prepared. However, the structure and dynamical properties for these larger systems were found to be similar to those of the smaller ones. After an energy minimization and an annealing procedure, they found densities comparable with those obtained from experiment [97].

The quantitative analysis by Devanathan and Dupuis led to a water content λ between 5 and 6 as the percolation threshold. This is larger than the λ values of about 3-4 reported by Gottesfeld and Zawodzinski [98]. This might be explained by the fact that Devanathan and Dupuis considered the percolation threshold as the λ value below which the hydrogen-bonded network is not continuous but contains isolated water molecules, while Gottesfeld and Zawodzinski defined their percolation threshold as the λ value below which the proton conductivity drops to zero. Another result in [18] is that the mean residence time of the proton on a water molecule decreases by 2 orders of magnitude when λ is increased from 5 to 15. Also, the reported value for the H^+ self-diffusion coefficient for $\lambda = 15$ at 300 K is about $1.1 \times 10^{-5} \mathrm{cm}^2 \cdot \mathrm{s}^{-1}$, which is of the order of the experimental value (about $9.3 \times 10^{-5} \mathrm{cm}^2 \cdot \mathrm{s}^{-1}$). A more detailed discussion is found in [91].

Also, Seeliger et al. [19] performed MD simulations of both a slab-like and a random-like model of Nafion at two water contents ($\lambda = 5$ and 10) with both a flexible and a rigid-bond representation of the Nafion chains. The rigid SPC/E water model [99] and a rigid classical hydronium model [100] were used. A so-called two-state empirical valence bond (EVB) model [101,102] was used to approximate the proton hopping mechanism predicted by Grotthuss. Once again, the systems were too small ($\approx$ 4 nm cube edge length, or $\approx$ 5000 atoms) to examine multiple clusters and hence the structure at the nanometer scale.

Nonetheless, these authors reported the formation of large water clusters connected by narrow bridges at high water content ($\lambda = 10$) and of filamentous clusters with irregularly shaped narrow cylindrical pores at $\lambda = 5$. As expected, the proton transport increased significantly with increasing λ, and the total proton diffusion was shown to be dominated by the Grotthuss mechanism rather than by the classical vehicular diffusion. Even though the systems were simulated over relatively large times (i.e. up to 30 ns), no unambiguous conclusion on the polymer dynamics and its influence on the proton transport properties could be drawn. On a 2 ns time scale (the one achievable with the EVB model and the flexible representation), the aqueous pore structure was found to be essentially static while after 20-30 ns (achieved with the rigidified model), some structural changes were observed. However, no significant difference between the initial and the final configurations occurred after 30 ns in the slab system. This indicates that very long simulation times are required to equilibrate such a glassy system.

The solvation and transport properties of the hydrated proton in the single hydrophilic domains of small Nafion systems ($\approx$ 4 nm cube edge length, or $\approx$ 5000 atoms with 40 excess protons) have been analyzed at a water loading of 15 H_2O/SO_3^- by Petersen et al. [20]. In this study, the Grotthuss proton hopping mechanism was treated by employing a sophisticated model, the so-called self-consistent iterative multistate empirical valence bond model [30, 103–106]. It is found that the excess proton is most of the time solvated between two water molecules of a Zundel ion in contact with a SO_3^- group. The passage of the excess proton from the contact to the solvent-separated ion pair (i.e. from $SO_3^- - H^+$ to $SO_3^- - H_2O - H^+$), and vice versa, occurs through the Grotthuss mechanism rather than by classical vehicular diffusion. Overall, the contributions of the vehicular and Grotthuss components to the total proton diffusion were found to be of the same magnitude. However, the Grotthuss diffusion is in general anticorrelated with the vehicular diffusion, resulting in a relatively small net diffusion.

In other words, the SO_3^- ions are said to act as proton "traps". According to Petersen et al., this could in part explain why side chains different in size exhibit different transport rates: a shorter side chain may restrain the SO_3^- groups from trapping the protons in the bulk water region, where transport could be fastest. On the other hand, a shorter side chain may possibly allow the protons to interact more closely with the hydrophobic backbone of the polymer, enabling transport along the hydrophilic/hydrophobic boundary.

Although these and numerous other simulation studies have significantly advanced the understanding of the intracluster structural and dynamical properties in Nafion membranes, none of them has been able to address the questions of larger-scale intercluster structure and dynamics.

2.3.2 Large Nafion Systems

A fully atomistic MD simulation study at a length scale sufficiently large to investigate the morphology of Nafion membranes was performed in 2010 by C. K. Knox and G. A. Voth [2], who examined six of the most popular morphological models of hydrated Nafion found in the literature, going from a so-called cluster-channel model similar to the one of Gierke et al. [4] to a so-called parallel cylinder model comparable with the more recent picture proposed by Schmidt-Rohr et al. [3]. They also looked at the so-called lamellar and random models. In their simulation experiment, each system was initially built to closely reproduce the proposed hydrophilic cluster structure of the given models. Each system was large enough ($\approx$ 2 million atoms in a cubic box with a $\approx$ 30 nm edge length!) to obtain several hydrophilic domains coexisting in the simulation box. Up to now, these Nafion systems are the biggest ones described in the literature.

Each system contained 200 distinct polymer chains with EW = 1100 $\mathrm{g} \cdot \mathrm{mol}^{-1}$ (corresponding to $x = 6.5$ in eq. 1) and 21000 counterion pairs. The systems were built at a water content $\lambda = 7.4$. In addition, the random model was also constructed at $\lambda = 15$. The authors used the all-atom modified DREIDING force field of Jang et al. [90, 107, 108], the F3C water model [109] and a classical model for the hydronium ions [90, 109], in keeping with many other studies of Nafion. Each initial configuration was equilibrated in the NVT ensemble at 600 K for at least 0.5 ns. The data for analysis were then collected in the NpT ensemble at p=1 bar and T = 300 K over at least 5 ns.

These simulations revealed fast intercluster bridge formation and network percolation in all systems. Furthermore, SO_3^- groups were located inside these bridges and played an important role in the percolation phenomenon. Sulfonate groups also strongly aggregated around and inside the so-called ionic clusters. Based on the observation that after equilibration, the ionomer peak (discussed above) was "fairly well" reproduced in the Ow-Ow structure factor by all the non-random models, in spite of their very different geometries, the authors concluded that each of these models is a plausible representation of real Nafion. This result supports a low order of structure and the inability of scattering data to elucidate the true structure of Nafion, hence the need for more detailed measurements. The authors further stated that if given sufficient time, the random model would reproduce the ionomer peak as well, "although the time scale for such peak formation is not computationally feasible". This is a point where this work is expected to contribute to the debate.

Alternately to MD simulations, mesoscale simulations have been used to study the hydrated morphologies of Nafion membranes (in e.g. [84, 110–113]). Such simulations lead beyond the bounds of what can be considered purely atomistic modeling. They allow, however under more restrictive assumptions, to determine a sample of the hydrophilic domains representative of a macroscopic amount of Nafion material.

In 2006, Wescott et al. [113] performed such novel mesoscale simulations of hydrated Nafion using a self-consistent mean field theory approach and a coarse-grained model by dividing the system into three components: backbone, side chain, and water. The interaction parameters for this model were determined from classical MD. In this study, four systems ($\lambda = 2, 4, 8, 16$) with an initially "homogeneous distribution of the components" were simulated at room temperature for 5000 steps (i.e. ≈ 150 μs).

The simulations at lower water contents led to isolated, nearly spherical domains. The domains were more elliptical or barbell shaped due to domain merging at higher water contents. For $\lambda = 16$, percolation was observed, as expected. However the authors concluded, verbatim: "The mesoscale simulations suggest that percolation could well occur in subnanometer sized hydrophilic domains, which cannot be resolved by mesoscale simulations." The phase separation was characterized by an order parameter for each bead type (a low order parameter means a low degree of order, or an homogeneous distribution, while a high order parameter means a strong phase separation). The time evolution of these order parameters showed that:
i) at $\lambda = 2$, the values are close to zero, indicating that no microphase segregation has occurred during the 150 μs simulated,
ii) increasing water content results not only in larger order parameter values (stronger segregation) but also in a quicker onset and increase in the rate of the phase separation,
iii) the water and fluorocarbon backbone beads display a higher degree of ordering than the side chain beads, indicating that segregation is driven by the strong hydrophobicity of the backbone chains.
Note, however, that the time evolution of the order parameters for the backbone and the water beads plotted in the paper does not converge to a constant value at the higher water contents ($\lambda = 8$, 16), suggesting that the phase separation was still evolving after 150 μs.

2.3.3 A Hyflon Simulation

With respect to the Hyflon materials, J. Karo, A. Aabloo and coworkers [58] have recently (2010) performed MD simulations of both this material and Nafion. For both systems, 32 polymer chains (each containing 10 side chains separated by 14 CF_2 groups) were placed together with 4800 water molecules and 320 H_3O^+-ions in a cubic box of edge length 8 nm, i.e. at a density below the expected experimental one. After equilibration, the density of both systems was found to be reasonable compared to the experimental density of Nafion at T=300 K and λ=15, .

In this study, the typical side-chain end-to-end distance was found to be 5.4 Å in Hyflon, whereas two characteristic values were found in Nafion: 5.9 and 7.1 Å, indicating a higher rotational degree of freedom for the longer side chain. Despite the difference in the side chain length, the authors concluded, from closely overlapping radial distribution and coordination numbers functions, that the local structures close to the SO_3^- groups and in the aqueous environment were very similar in both systems. Even though the typical S-S distance between adjacent side chains was found to be higher in Nafion (16.1 Å) than in Hyflon (15.1 Å), the S-S rdfs (i.e. the distribution of the distances between any given S-S pair) did not show any significant differences between the short and the long side-chain materials. In passing, we note that the S-S rdf is of particular interest because it reveals the mean sulfonate-sulfonate distance (based on the location of the first peak), which is believed to strongly influence how rapidly protons can hop between sulfonates according to the Grotthuss mechanism [2]. In both cases, broad peaks were observed, indicating that the local distances between side-chain ends are not well determined. However, two maxima were observed, at 7.1 Å and 9.0 Å, irrespective of the side-chain length. This is in agreement with Voth and Knox [2] and others [90,91,114], who found the first maximum in the 5-8 Å region. On the other hand, by counting side chains that are closer to each other than 8 Å (the distance at which the first minimum occurs in their S-S rdf plot, i.e. within the first hydration shell), the average size of SO_3^- clusters was found to be slightly higher in Hyflon (4.1) than in Nafion (3.6). This difference is rather small and may be fortuitous.

Even though the systems were not large enough to investigate the nanoscale morphology, a hydrophilic/hydrophobic phase separation was observed. In both ionomers, almost all the water oxygens (Ow) in the system were found in a unique and very large cluster (with an $\mathrm{Ow}-\mathrm{Ow}$ distance criterion of 4.5 Å for the cluster analysis, see section 4.4, i.e. the border between the first and the second coordination shell according to their $\mathrm{Ow}-\mathrm{Ow}$ rdf plot), the rest of the waters being distributed in very small clusters which regularly rejoin and break away from the dominant one. Also, in both materials, almost all SO_3^- groups were located either close to the backbone/water interface or within the aqueous phase, and most of the hydronium ions were observed around the sulfonate groups. As the distribution of the number of water molecules in water clusters displayed a larger width in Nafion, the authors deduced a better stability of the water network structure in Hyflon. Also, from spatial distributions of the main components in the system (backbone, side chain and H_2O/H_3O^+), the phase separation at the interface was said to be more dispersed in Hyflon.

Although in both systems most of the H_3O^+ ions are bonded to SO_3^- groups, the number of 'free' H_3O^+ was found to be larger in Nafion. Mean residence times τ_{MRT} were computed (following [115]) in order to measure how long, in average, an H_2O or an H_3O^+ remains within the first coordination shell of the SO_3^- groups. For both Nafion and Hyflon, an H_3O^+ ion was found to remain roughly four times longer than a H_2O molecule around a given SO_3^- group, because of the stronger $H^+ - O^-$ interaction. Because of the larger number of 'free' H_3O^+ observed in Nafion, one expects, assuming a simple kinetic model, shorter τ_{MRT} in Nafion. The τ_{MRT} values were indeed found to be larger in Hyflon, by a factor of ≈ 1.32 and 1.24, for H_2O and H_3O^+, respectively.

The authors thus concluded on a more stable coordination shell around sulfonates in Hyflon. On the other hand, the local mobility of the different side-chain atoms, investigated using mean square displacement (MSD) functions, showed that:
i) the mobility increases outward along the side-chain, with a maximum mobility for the SO_3^- groups,
ii) the absolute MSD values of the side chain atoms are significantly higher in Hyflon than in Nafion.
Finally, self-diffusion coefficients D were calculated from the MSD curves via the Nernst-Einstein equation for H_2O and H_3O^+. The values for $D_{H_3O^+}$ were found to be roughly four times lower than those for D_{H_2O}. Furthermore, the D values were found slightly higher in Hyflon, the difference being 3.9×10^{-6} and 0.3×10^{-6} $\mathrm{cm}^2 \cdot \mathrm{s}^{-1}$ for H_2O and H_3O^+, respectively. We will compare results from both Nafion and Hyflon as well, using large-scale systems.

3 Simulation method

3.1 General principles

Both the Molecular dynamics (**MD**) and the Monte Carlo (**MC**) computer simulations are based on statistical mechanics [116, 117]. They allow to make the link from an assumed microscopic description of a chemical system to its measurable macroscopic properties or observables. Examples are the density of a system under given thermodynamic conditions, its transport coefficients, structural order parameters, etc. Such calculated quantities can be directly compared with the experimentally observed ones. If eventually the experimental observables are well reproduced, the model, the simulation method and the assumptions involved can be accepted. Any new quantity that can be obtained from such a simulation is at least consistent with the underlying model. One may thus assume that a so-computed quantity is a realistic prediction for a possible corresponding experiment. It should be noted that, in contrast to the experiment, there is in principle no restriction to the simulation regarding the thermodynamic conditions. Simulation results for well tested systems are thus often used to complement experiments, or to replace experiments, which cannot be performed for technical reasons. Similarly, if agreement has been found between a computed and a measured quantity (observable), the simulations often allow a more refined (and via the model consistent) interpretation of the experiment, e.g. the determination of partial structure factors that are difficult or impossible to measure.

Even though these methods neglect explicit quantum effects (e.g. compared to others with an admittedly more accurate microscopic description like in ab initio or semi-empirical approaches), it is their capacity to reproduce the average behavior (the macroscopic properties) of dense many-particle systems at finite temperatures which makes them the tool of choice for our purposes. It is particularly important that the computational times remain within reasonable limits.

It is clear that at the level of atoms and molecules, the adequate description is quantum mechanics. In chemistry, it usually deals exclusively with the electronic degrees of freedom. Modern quantum chemistry codes allow to solve numerically the Schrödinger equation for 'small systems', i.e. molecules or aggregates with up to a few hundred electrons. One can study, for instance, reactions between atoms and/or molecules in their accessible and probable quantum states. For much bigger systems, say several hundred atoms, the Schrödinger equation cannot be solved with reasonable accuracy and effort.

Statistical mechanics, as developed by Ludwig Boltzmann [116] and Gibbs [118], on the other hand, assumes that classical mechanics can be used to reproduce the motions of atoms and molecules. This approximation

i) is adequate for any atomic nucleus heavy enough so that its de Broglie-wavelength does not exceed its own size, i.e. all but the hydrogen atom,
ii) leads to a great simplification in almost all calculations [119].
As electrons are ignored, methods based on classical mechanics cannot provide properties that depend upon the electronic distribution in the system. Problems in statistical mechanics are hence subject to the Born-Oppenheimer approximation, in which the Hamiltonian $\mathcal{H}$ of a system of N particles can be expressed as a function of the nuclear variables only (the electronic motions having been 'averaged out'). Consequently, the microscopic state of a statistical mechanics system can be specified in terms of the positions and momenta of a constituent set of classical particles that represent the atoms and molecules in the system. The Hamiltonian can then be written as:

$$\mathcal{H}(\mathbf{q},\mathbf{p}) = \mathcal{K}(\mathbf{p}) + \mathcal{V}(\mathbf{q}) \tag{2}$$

where $\mathcal{K}$ and $\mathcal{V}$ are the kinetic and potential energy functions, respectively. $\mathbf{q}$ and $\mathbf{p}$ are the set of coordinates $\mathbf{q}_i$ and momenta $\mathbf{p}_i$ of each particle i, respectively.

$$\mathbf{q} = (\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_\mathrm{N}) \tag{3}$$

$$\mathbf{p} = (\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_\mathrm{N}) \tag{4}$$

The kinetic energy is usually written as:

$$\mathcal{K} = \sum_{i=1}^{N} \sum_{\alpha} \mathbf{p}_{i\alpha}^2 / 2m_i \tag{5}$$

where m_i is the mass of particle i and the index α runs over the (x, y, z) components of the momentum $\mathbf{p}_i$.

The basic input to a computer simulation is the potential energy $\mathcal{V}$; its aim is to reproduce the real intermolecular interactions with an acceptable accuracy, using unsophisticated mathematical functions.

3.1.1 Models

The atoms and molecules constituting the chemical system are represented by classical particles, i.e. point masses or rigid bodies subject to the laws of classical mechanics and interacting through known potentials. The parameters for these interactions are often derived from fitting procedures based on quantum mechanics calculations. Only a limited number of such particles can be treated even in the most favorable case. Thus, extended systems must be represented by a finite number of such particles, which are usually located in a cell (most frequently a cube). The interactions between these particles, i.e. atoms or molecules, are often expressed as pairwise additive potentials between so-called interaction sites. A site may be identified with an atom in a molecule, or with a group of atoms (united atom model [120]). Sites may also be located at various positions on molecular frames, assuming they remain unchanged (rigid) throughout the simulation.

Since the computation of the interactions (forces) is the most computer-time consuming step in a simulation, the computational effort can be reduced by thus defining only one interaction site for a group of atoms. In this case one talks about united atoms or beads and the interaction sites are usually located at the centers of masses of the groups of atoms to be considered. We note in passing that in order to speed up the calculations even more, the number of degrees of freedom (particles) can be much reduced by using e.g. a continuum-dielectric model to replace solvent molecules. These methods are, however, not subject of this thesis.

The main assumption of the present methods is that the interatomic and/or the intermolecular interactions are assumed to result from a pairwise summation involving all possible pairs of the interaction sites. The assumption of a pairwise potential is not a fundamental one, as the interaction between two molecules can be affected by the presence of a third, fourth or more molecules. However using three-body-interactions would lead to computer time requirements larger by about one order of magnitude [41]. Some work has nevertheless been performed in this direction for simple systems [121], for water [122–124] and also for salts such as $BeCl_2$ [125] and $AlCl_3$ [126] dissolved in water. Instead, the average many-body effects can be included by defining an 'effective' pair potential. Even with a simple pairwise interaction model, the computation of the intermolecular interactions require by far the largest amount of computational time.

The construction of an intermolecular potential can be decomposed into two distinct tasks:
(i) first an arbitrary but reasonable guess for the functional form and the set of parameters defining the intermolecular potential has to be done.
(ii) From the preliminary simulation results, these parameters (and if necessary the functional form) can be refined iteratively until it reproduces correctly the expected behavior of the system at both the microscopic and macroscopic scales.
One would want to avoid to have an entirely different model for each chemical system and thermodynamic condition one wants to study. The adjusted functional and set of parameters should therefore be appropriate for a range of systems as extended as possible. This is the so-called concept of 'transferability' or 'portability'. For instance, the same set of parameters should give satisfactory results for all n-alkanes [120]. Similarly, if a model reproduces the properties of, say, polyethylene in water, correctly, the properties of, say, 'linear' polymers in other polar solvents, should be well reproduced as well.

Typically an intermolecular pair potential is written as a sum of a Lennard-Jones (**LJ**) and a Coulomb potential:

$$V = \sum_{i>j}^{N} \sum_{j=0}^{N-1} \Big\{ 4\, \varepsilon_{ij} \Big[\big(\frac{\sigma_{ij}}{r_{ij}}\big)^{12} - \big(\frac{\sigma_{ij}}{r_{ij}}\big)^{6} \Big] + \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} \Big\} \tag{6}$$

where N is the total number of particles in the system, ε is the depth of the LJ potential well, σ is the finite distance at which the interparticle potential is zero, r is the distance between the sites, i (on a molecule α) and j (on another molecule β), q is the charge and ϵ_0 is the dielectric permittivity.

The parameters ε_{ij} and σ_{ij} with $i \neq j$ in the LJ potential (left term in eq. 6) are deduced from ε_{ii}, σ_{ii}, ε_{jj} and σ_{jj} using the so-called Lorentz-Berthelot mixing rules:

$$\sigma_{ij} = \frac{1}{2}(\sigma_{ii} + \sigma_{jj}) \tag{7}$$

$$\varepsilon_{ij} = (\varepsilon_{ii}\varepsilon_{jj})^{1/2} \tag{8}$$

The term in r^{-12} in eq. 6 mimics the repulsion due to the overlap between the electronic clouds that surround the nuclei whereas the term in r^{-6} aims to reproduce the attractive correlation between the electronic clouds (Van der Waals forces or London dispersion). In spite of its simplicity, and although more sophisticated potentials exist [127], the Lennard-Jones potential is considered as a reference and is very often used. It can reproduce in many cases the experimental data and/or the quantum mechanics calculations for a wide range of monoatomic and molecular systems [128–130]. Presently, a collection of appropriate values of ε_{ii} and σ_{ii} is available for almost every kind of atom i and for different 'classes' of molecules. They can be found in force field databases like [131–135].

All electrostatic interactions are modeled by sums of Coulomb potentials (rightmost term in eq. 6) between pairs of point charges on different molecules α and β or within the same molecule, if they are sufficiently distant [129]. These terms are very often the dominant ones, especially in systems with polar molecules or ions. Multipoles of order higher than one, i.e. dipoles, quadrupoles, and so on, do not appear explicitly in eq. 6. However, for ionic and polar systems, it may be convenient to describe the interactions (at least in the far field) in terms of, e.g., charge-dipole, dipole-dipole, dipole-quadrupole interactions, and so on. The charge distributions for most water models [136] leads to a molecular dipole of 1.9-2.2 Debye. However, higher multipole moments may not be equally well represented.

It is thus possible to add to such simple point charge models additional point multipoles at selected sites. Making these additional moments field dependent is a way to add the molecular polarizability to a model [137, 138]. Polarization effects are not pairwise additive and thus difficult (and computationally expensive) to handle: including them into the simulations requires to readjust the induced moments at each time step in a self-consistent way with the field. A lot of work has been performed in this direction, especially for water [136]. Since none of the so-called polarizable models has been shown to be really superior to the conventional ones [137,138], and due to the computational expenses, polarization effects are usually omitted for large systems, so also here.

In addition to the intermolecular pair potential, bonds, bendings and torsions inside the molecules can be modeled with e.g. simple harmonic potentials or with more sophisticated potentials like the ones described in [139]. Typical simple potentials for bonds, bendings, and torsions are:

$$V_{\text{bond}} = \frac{1}{2}\, k_b\, (r - r_0)^2 \tag{9}$$

where k_{b} and r_0 are the force constant and the equilibrium length of a given bond type, respectively.

$$V_{\text{angle}} = \frac{1}{2}\, k_\alpha\, (\cos\alpha - \cos\alpha_0)^2 \tag{10}$$

where k_α and α_0 are the force constant and the equilibrium angle of a given bend type, respectively.

$$V_{\text{dihedral}} = \frac{1}{2}\, k_\phi\, \{1 - \cos[3(\phi - \pi)]\} \tag{11}$$

where k_ϕ and ϕ are the force constant and the dihedral angle between the three first and the three last bonded atoms, respectively. This is the simplest possible case, higher order terms ($\cos(n\phi)$, $n > 3$) often need to be added. Coulombic and LJ interactions are usually not computed between the atoms participating in the bonds, angles, and dihedrals. As an alternative to intramolecular potentials, a molecule can be made rigid using constraint algorithms as described in [140,141]. We will come back to the inter- and intramolecular potentials more explicitly in a chapter dedicated to the model and force field.

3.1.2 The MD method

A major difference between the MD and MC methods is that only the positions of the particles (or sites) are generated in MC, hence there is no explicit kinetic energy in the system. MD, on the other hand, also yields the velocities of the particles and one can compute time-dependent properties. More specifically, MD generates a sample of a thermodynamic ensemble of system configurations that are connected in time. This is achieved by integrating Newton's law of motion for each particle in the system. This yields the trajectory, i.e. the positions and velocities of all particles as a function of time, reproducing the time-evolution of the system over a period of time Δt.

In principle, an MD simulation is very similar to a real experiment. In both cases, a sample of the material of interest has to be prepared. The sample is then submitted to measurements over the period required for the measurement. The longer the simulation, i.e. the Δt defined above, compared to the intrinsic time scale of the measurement, the more accurate the results of the simulation .

The MD procedure can be summarized by the following two equations:

$$\frac{\mathbf{F}_i}{m_i} = \ddot{\mathbf{r}}_i = \frac{\partial^2 \mathbf{r}_i}{\partial t^2} \tag{12}$$

$$\mathbf{F}_i = -\nabla_i V \tag{13}$$

in which $\mathbf{F}_i$ is the force acting on a particle i with a mass m_i and a position $\mathbf{r}_i$ at time t. ∇_i is the gradient vector with respect to the coordinates of particle i.

The new positions and velocities of a given particle are determined from the force acting on it, itself derived from the potential energy of the system. The potential energy V depends on the positions of the particles only. The differential equations of motion of the particles cannot be solved analytically, instead finite difference methods based on Taylor series expansions are used. As an example, the Verlet algorithm determines the positions at time $t + \delta t$ from the positions at time $t - \delta t$, δt being a finite time step:

$$r(t + \delta t) = 2r(t) - r(t - \delta t) + \delta t^2 a(t) \tag{14}$$

Important criteria for the choice of an integration algorithm are:
i) it should reproduce the 'correct' trajectory,
ii) the conservation law of energy and linear momentum,
iii) to allow for a time step δt as large as possible.

The trajectory is also a sample of the phase space, i.e. of possible combinations of positions and velocities that satisfy the given thermodynamics conditions. The sample of phase space must be as representative as possible in order to lead to satisfactory results i.e. with a statistical uncertainty small enough to draw the desired conclusions. Technically it would be desirable to obtain configurations that are statistically uncorrelated, i.e. that do not depend on each others and that do not depend on the initial configuration. This allows an estimation of the uncertainties with standard statistical tools. In principle, if the simulation is performed over a sufficient long time Δt, an average computed property does not depend on the initial positions and velocities. The ergodicity theorem says that the time average of a given quantity equals its ensemble average, in other words, averaging over all initial phase space coordinates is equivalent to averaging over the time-evolved phase space coordinates [119].

3.1.3 Applying statistical mechanics

Statistical averages over the simulated sample trajectory can be computed as

$$\langle \mathcal{A} \rangle = \langle \mathcal{A} \rangle_t = \frac{1}{M} \sum_{i=1}^{M} \mathcal{A}(\mathbf{r}_i, \mathbf{v}_i) \tag{15}$$

for any quantity $\mathcal{A}$ that can be obtained from the positions $\mathbf{r}$ and velocities $\mathbf{v}$ of one or several equivalent particles. M is the number of time steps and $\langle \ \rangle$ means the ensemble average. $\langle \ \rangle_t$ means the time average. Taken over a sufficient number of points (as mentioned, preferably uncorrelated in time and space), the so-obtained quantities are the proper thermodynamic averages of the structural properties of the system. In other terms, one assumes that

$$\mathcal{A}_{\text{obs}} = \langle A \rangle_t \tag{16}$$

$\mathcal{A}_{\text{obs}}$ being the experimentally measured quantity.

An important concept in statistical mechanics, that we shall introduce here, is the one of correlation. Its objective is to quantify the relation between two measurements (or two sets of data) taken at two different points in space or time, e.g. between the value of a quantity B at time t and the value of a quantity A at time 0. For this case, a time-correlation function can be defined as:

$$c_{\text{AB}}(t) = \langle A(0) \odot B(t) \rangle \tag{17}$$

where $\odot$ is, for instance, the scalar product operator $\cdot$. It can be easily shown that in this case:

$$\lim_{t \to 0} c_{\text{AB}}(t) = \langle A \cdot B \rangle \tag{18}$$

and

$$\lim_{t \to +\infty} c_{\text{AB}}(t) = \langle A \rangle \langle B \rangle \tag{19}$$

$B(t)$ is said to become uncorrelated with $A(0)$ when the correlation function approaches its long-time limit. For the case of exponential correlation functions, usually written $c_{\text{AB}}(t) = \exp(-t/\tau)$, the correlation (or rather the de-correlation) time τ is usually reported. For the general case of non-exponential correlation functions, an effective τ can be determined by integrating the time-correlation function from $t = 0$ to ∞.

One talks about cross-correlation when A is different from B and about auto-correlation when A is the same as B. Time auto-correlation functions are of special interest, because:
i) their time integrals are often related to macroscopic transport coefficients,
ii) their Fourier transforms are often related to experimental spectra.

One typical example is the normalized auto-correlation function of the particle velocities:

$$c_{\mathrm{vv}}(t) = \frac{1}{M}\frac{1}{N}\sum_{j=1}^{M}\sum_{i=1}^{N}(v_i(0)\cdot v_i(t)) \tag{20}$$

where N is the number of particles and M is the number of time origins. The time integral of c_{vv} is directly related to the self-diffusion coefficient (D) of the particles in the system:

$$D = \frac{1}{3}\int_0^{+\infty} dt\ c_{\mathrm{vv}}(t) \tag{21}$$

and its Fourier transform, called the spectral density of motions, is related to the vibrational spectrum (infrared, Raman, neutron scattering) of the system:

$$F(\omega) = \frac{1}{\sqrt{2\pi}}\int_0^{+\infty} dt\ e^{i\omega t}\ c_{\mathrm{vv}}(t) \tag{22}$$

Two chapters are dedicated to time-correlation functions in [116] and the reader is referred to this or similar standard textbooks.

3.2 Practical aspects

3.2.1 Simulating bulk phases

MD and MC aim to reproduce the properties of a macroscopic sample, however, the number of atoms that can be handled by MD or MC is usually small compared with any macroscopic sample. However, Lustig [142] has shown that for simple thermodynamic quantities the 'thermodynamic limit' can be reached for small (less than 100, say) numbers of particles. Furthermore, a non-negligible fraction of all particles present in the cell would be affected by the presence of hard cell boundaries. These particles would not behave as if they were in the bulk phase. Actually, for a three-dimensional system containing N particles, the fraction of all particles that are at the surface of the cell is proportional to $N^{-1/3}$ [119]. According to [119] and as an example, in a simple cubic crystal of 1000 atoms, some 49% of all atoms are at the surface, and for 10^6 atoms this fraction has decreased to only 6%. Therefore, some 'tricks' are needed in order to obtain a pseudo- or quasi-infinite system to simulate a bulk phase. We follow here [41].

- Surface effects are avoided by the so-called 'periodic boundary conditions' construction. The cell and all the particles are replicated infinitely in all directions. The boundaries of the cell are virtual (or transparent), so that the particles are free to move between the cell and its periodic replica. As a replica particle moves into the cell whenever a particle moves into a replica cell (and thus becomes a replica particle), the number of particles is kept constant. This is illustrated in fig. 6: a particle from the cell moves into the replica cell C while its replica image from the replica cell G moves into the cell. Thus, with such a construction, a given particle i interacts with an infinity of particles in the infinite periodic lattice and so far, the periodic boundary condition algorithm alone is not helpful, as the potential would be rewritten as an infinite sum rather than a finite one.

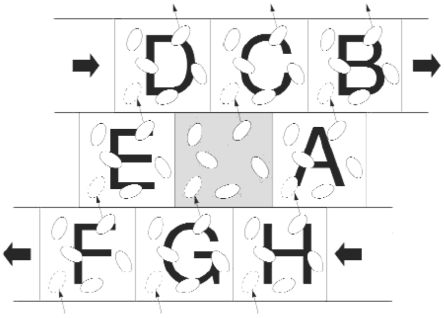


Figure 6: Periodic boundary condition scheme, taken from [143].

- The 'minimum distance convention' is applied. This is a prescription for how the interaction potentials, and thus the interparticle forces, exerted by all particles j on a given particle i, are to be computed. The smallest of all distances between particle i and either particle j, or any replica particle of j in any of the surrounding replica cells, is used to determine the energy and forces. Periodic boundary conditions together with minimum distance convention allow to study an infinite system with a finite number of independent particles.

One consequence of the periodicity of the model is that only those collective motions that have a wavelength λ such that $\lambda \leq L$, L being the edge of the box (or the smallest edge if the box is not a cube), are allowed. Therefore L has to be chosen larger than the longest wavelength of interest in the system. Another consequence of this construction is that interactions between particles separated by a distance further than $L/2$ are ignored (or truncated). For homogeneous systems, it is more appropriate to truncate the interactions beyond a sphere defined by a cut-off radius r_c (with $r_c \leq L/2$) rather than beyond a cube.

3.2.2 Short-range and long-range interactions

Truncating the interactions beyond r_c is feasible for so-called 'short-range' interactions, i.e. interactions $V(r)$ where the contributions to the total energy beyond r_c tend towards zero:

$$\int_{r_c}^{\infty} V(r)\ r^2\ g(r)\ dr \to 0 \tag{23}$$

where $g(r)$ is the so-called pairwise radial distribution function. As $g(r) \to 1$ when $r \to \infty$, the condition for this integration to tend to zero is that $V(r)$ tends toward zero like r^{-n} with $n > 3$.

There is a discontinuity in the intermolecular potential at $r = r_c$ when $V(r)$ is truncated. This leads to (in principle) infinite forces and thus numerical errors. Therefore an additional term is needed in order for the intermolecular potential to converge smoothly to zero in the vicinity of the truncation distance r_c. Also, the error introduced in the evaluation of the intermolecular potential by truncating can be corrected with another additional term, resulting from the integration of the pair potentials from r_c to ∞. This is discussed in textbooks like [119,120,143], under the terms of 'potential truncation', 'shifted force', and 'long-range correction'.

Long-range interactions, such as charge-charge, charge-dipole, dipole-dipole interactions typically decay not faster than $r^{(-n-m+1)}$, n and m being the orders of the two multipoles. A solution to handle such long-range interactions, since they cannot be truncated, is e.g. the Ewald summation method [144], in which a given particle i interact not only with the other particles in the simulation box but also with all other replica particles in the infinite periodic lattice. This is akin to the determination of the Madelung [145] constant in solid state physics, it is explained in books like [119,120,143].

3.2.3 Initialization

Before starting a simulation, initial positions and velocities have to be assigned to all particles in the system. From this initial configuration, the system evolves until equilibrium is reached (which may not happen spontaneously in times accessible to the simulation). It is very difficult to demonstrate strictly that equilibrium has been reached. Necessary conditions for equilibrium are, however, that thermodynamic quantities like e.g. (depending on the ensemble) the total energy, the temperature or the density fluctuate around constant values. The thermodynamic quantities that are expected to be strictly constant in a simulation are the ones stipulated in the conditions $((NVE), (NpT), ...)$ that have been chosen (see next section). It is an indication of numerical or other computational problems (e.g. truncation errors [146], time step too big, etc.) if they are not constant.

As the equilibrium properties of the system should not depend on the initial conditions, all reasonable initial conditions are acceptable. It should, however, be mentioned that any high energy interaction between two overlapping particles may cause instabilities in the system (or would require unreasonably small time steps), which would certainly lead to a violation of energy and linear momentum conservation and could result in a program failure due to numerical overflow. It is recommended to remove such 'bad contacts' by running (prior to the simulation itself) an energy minimization algorithm. A natural choice for the initial positions is to prepare the system in a structure close to the experimental one (if known), a previous simulation of a similar system, or a combination of the two. The velocities of the particles are distributed randomly so that the total linear momentum is zero and so that each particle contributes, on the average, equally to the total kinetic energy. The distribution of choice is the one of Maxwell-Boltzmann, which is strictly valid in the canonical ensemble and closely approximates the velocity distributions in the other ensembles.

If it is found that certain computed quantities are not independent of the initial conditions, it either means that the system behaves non-ergodically or that our sample of the phase space is simply inadequate and that the equilibrium has not yet been reached. In the latter case, the last configuration obtained from the last simulation run should be taken as a new start for a new run, until equilibrium is reached. Similarly, if one wants to simulate a system under different thermodynamic conditions (e.g. at a different temperature), one should take a well pre-equilibrated configuration from a simulation performed under thermodynamic conditions close to the new ones.

3.2.4 Thermodynamical ensembles

MD simulations are usually performed in the so-called 'MD' or (NVE)-ensemble, in which the number of particles N, the volume V, and the total energy E defining the system are constant. This is a sub-ensemble of the microcanonical ensemble of statistical mechanics in which the total linear momentum is also kept constant, usually $\vec{P} = 0$. Other statistical ensembles like the canonical (NVT) or the isothermal-isobaric (NpT) ensembles can be sampled by MD using a thermostat and/or a barostat as external degrees of freedom [147–152]. Briefly, a thermostat constrains the system to small fluctuations around a constant temperature by scaling the velocities of the particles whereas a barostat fixes the pressure by scaling the positions of the particles.

3.2.5 Time step

As stated e.g. in [120], the value of the finite integration time-step δt has to be chosen carefully; too small and the trajectory will cover only a limited proportion of the phase space; too large and instabilities may arise in the integration algorithm due to high energy overlaps between atoms. The aim is to find the correct balance between simulating the 'correct' trajectory and covering the phase space. A useful guide is that the time step should be smaller than approximately one tenth of the time of the shortest period of motion (as seen e.g. in the spectral density of motions). To speed up the calculations, this highest frequency in the system, which is often due to bond stretches, can be removed by fixing the distances between bonded atoms. A larger δt can thus be chosen. δt can be increased even further by ignoring the second fastest motions in the system: usually the angle bendings. Making in this way a flexible molecule rigid is achieved by using constraint algorithms as described in [140, 141]. Other special procedures and computational tricks to speed up the calculations (like the Verlet neighbor list or the multiple time step algorithm) are found in standard textbooks like [119, 120, 143].

Quantum corrections to quantities computed from classical MD simulations, exist, but we shall not discuss them here. More information about computer simulations and MD can be found in standard textbooks (e.g. [119, 120, 143]).

4 Data Analysis

In this chapter, we give a list of the analyses we applied to our trajectory data. Only the 'non-standard' ones are described. For the other ones, the reader is referred to a standard textbook (e.g. [120, 143]).

4.1 Thermodynamics

- A thermodynamical quantity that can be directly compared with the experiment is the density ρ. Note that besides the macroscopic mass density (kg·m^{-3}) the term density will mostly be used here as a short form of 'number density' (number of particles per unit volume (e.g. nm^{-3})). We will also make use of the concept of local density as defined below.

4.2 Estimating Errors

This section is based on [143]. We refer back to section 3.1.3 for the fundamental equations.

- RMSD (root mean square deviation) σ:
  For the thermodynamic average of a quantity $\mathcal{A}$:

  $$\langle \mathcal{A} \rangle_{\text{run}} = \frac{1}{M} \sum_{\tau=1}^{M} \mathcal{A}(\tau) \tag{24}$$

  where M is the number of time steps in the simulation run and τ is the time step index. Because simulation averages are taken over runs of finite length, statistical imprecisions in the mean values so obtained arise. It is often possible to analyze such statistical errors in averages like $\langle \mathcal{A} \rangle_{\text{run}}$ by assuming that $\mathcal{A}(\tau)$ is a Gaussian process. If it is furthermore assumed that each quantity $\mathcal{A}(\tau)$ is statistically independent of the others, the error can be estimated by $\sigma(\langle \mathcal{A} \rangle_{\text{run}})$:

  $$\sigma(\langle \mathcal{A} \rangle_{\text{run}}) = \frac{\sigma(\mathcal{A})}{M^{1/2}} \tag{25}$$

  where

  $$\sigma^2(\mathcal{A}) = \frac{1}{M} \sum_{\tau=1}^{M} (A(\tau) - \langle \mathcal{A} \rangle_{\text{run}})^2 \tag{26}$$

4.3 Vibrational Analysis

The computed frequencies of the vibrational modes in a molecule, and also of the hindered translational and rotational (librational) motions, can be compared with the experimental ones obtained e.g. from infra-red spectroscopy. This is a way, first, to test the force field and the force field parameters, second, to determine a reasonable value for the integration time-step of the simulation, δt (see section 3.2.5). There are essentially two ways to compute these vibrational frequencies:

- Normal mode analysis:
  Based on the assumption of harmonicity of the dynamics, the extraction of the normal modes is done by diagonalization of the mass-weighted Hessian matrix for the total potential energy of the system. For a usual normal mode calculation, it is necessary to completely minimize the energy of the given configuration prior to the computation of the Hessian matrix. The calculation of the normal modes also requires that no constraints regardings the stretching and the bending modes are applied on the molecules present in the system. We used the simulation package Gromacs [129] to perform the normal mode analysis. An overview of normal mode analysis can be found in [153], the classical textbook being the one by Wilson and al. [154].
- Fourier transformation of the velocity auto correlation function c_{vv}:
  The correlation function $c_{\mathrm{vv}}(t)$ (see section 3.1.3) in the time domain is transformed to a spectrum in the frequency domain. More specifically, we used the method "LADO" (from its author [155]) described in the microfiche 37 of the standard textbook *Computer Simulation of Liquids* [143]. This and others microfiches are found on the Internet [156]. The Fourier transform of $c_{\mathrm{vv}}(t)$ is called the spectral density of motions or the power spectrum. It is directly related to the spectra obtained from inelastic neutron scattering. It reflects all motions in the system, inter- and intramolecular, and is not restricted to the harmonic approximation. Projection techniques [157] allow to disentangle these spectra and to make assignments to approximate normal modes. We note that the infrared spectra can be obtained directly with a very similar technique involving the autocorrelation function of the dipole moment(s) or their derivatives.

4.4 Structure

- Total structure factor $S(q)$:
  A way to compare the simulated structure with experimental data is to calculate the so-called total structure factor $S(q)$ by Fourier transformation of the radial distribution functions (rdf or $g(r)$) involving all pairs in the system. Partial

structure factors involving the pairs $\alpha - \beta$ in the system, $S_{\alpha\beta}(q)$s, describe the intensity of the scattered beam (mostly X-rays or neutrons) as a function of the scattering angle (or more generally the scattering vector q):

$$S_{\alpha\beta}(q) = 1 + 4\pi\rho \int_0^\infty dr\ r^2\ [g_{\alpha\beta}(r) - 1]\frac{\sin(qr)}{qr} \tag{27}$$

where q is the scattering vector, ρ is the macroscopic density, r is the radial distance in the rdf $g_{\alpha\beta}(r)$.

The total $S(q)$s can be seen (see below) as weighted sums of partial $S_{\alpha\beta}(q)$s obtained from the various $g_{\alpha\beta}(r)$s involving the pairs $\alpha - \beta$ in the system. The weights depend on the concentrations and on the nature of the particles involved, they are different for different types of scattering (e.g. neutron / X-ray / electron scattering). We used the software R.I.N.G.S. [158] (Rigorous Investigation of Networks Generated using Simulations) to compute the total and the partial structure factors.

- Partial structure factor $S_{\alpha\beta}(q)$:
  An advantage of the simulation over the experiment is that $S(q)$ can be decomposed as a sum of partial contributions $S_{\alpha\beta}(q)$ involving the atom types α and β.

- Cluster analysis:
  A standard cluster analysis discretizes the spatial distribution of given particles into clusters, according e.g. to a distance criterion. At the start of the clustering algorithm, there are as many clusters as particles. Two clusters merge into one when a distance between any two particles (that do not already belong to the same cluster) is shorter than the distance criterion. A cluster is 'closed' when there is no cluster closer than the distance criterion. This is illustrated in fig. 7.

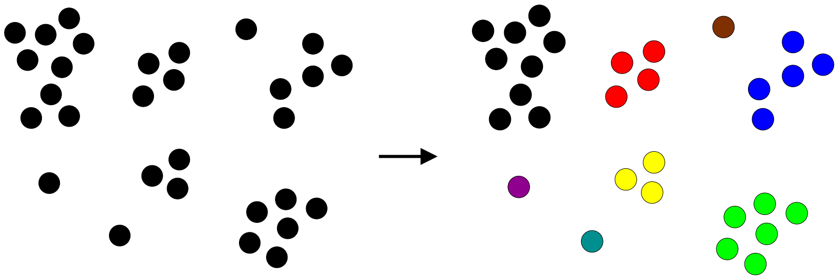


Figure 7: 'Basic' clustering algorithm scheme. Left: starting situation: N particles, N clusters, average cluster size = 1 particle; right: after analysis: N particles, 8 clusters, average cluster size=3.75 particles.

The simulation package Gromacs [129] has a subroutine called ”g_clustsize” to perform such an analysis. It extracts (among other quantities), for a given configuration or for a given set of configurations, the number of clusters, the size (in number of atoms or particles) of the largest cluster, and the distribution of the sizes of the clusters. The algorithm is found e.g. in the source code library of Gromacs, or in the microfiche 34 [156] of *Computer Simulation of Liquids* [143].

Since in most cases where we applied such an analysis, we found a very large and dominant cluster (i.e. most of the selected particles belonged to the same unique cluster), we extended this clustering routine to separate the largest cluster into smaller 'dense' subclusters that would be connected by bridges. Therefore we extracted for each configuration the largest cluster from which we then removed the particles that have a low coordination number (up to three) until we found multiple subclusters. The algorithm is described with a simple scheme in fig. 8. In this scheme we remove the particles that have less and up to only two neighbors. At the end of the algorithm, there are n subclusters (or n-1 bridges). We study the size distribution of these subclusters.

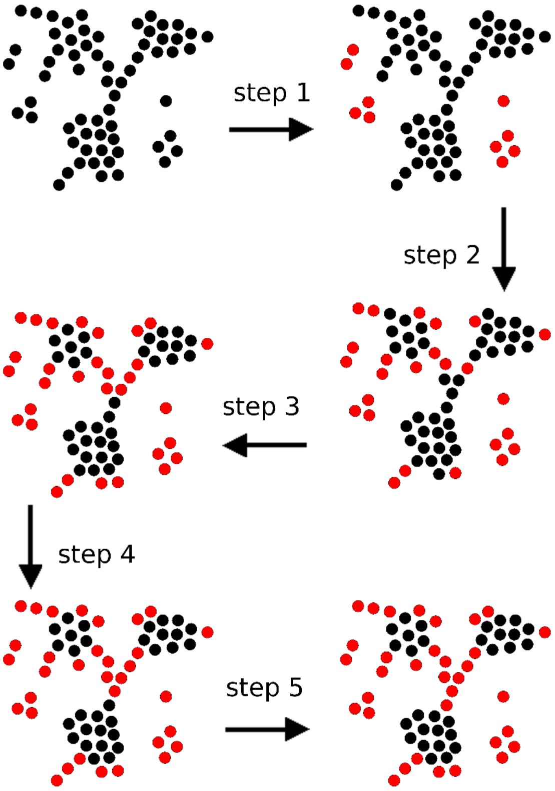


Figure 8: 'Advanced' clustering algorithm scheme. 1: extract the largest cluster; 2, 3, 4, 5: remove atoms with less and up to two neighbors iteratively.

- Other evaluations:
  The radial distribution function (rdf) $g(r)$, i.e the distribution of the distances of selected pairs of selected atom types, and its integral, $n(r)$, i.e. the coordination number, are routinely computed.

  'VMD snapshots': Taking 'pictures' of the system is an intuitive way to learn about the morphology. The VMD [159] (visual molecular dynamics) software is a versatile public domain 3D tool particularly useful to look at the system graphically.

  The end-to-end distance $d_{\mathrm{n-n}}$, i.e. the distance from the head to the tail of a given polymer chain, is a rough measure of the 'compactness' of a structure. It is also routinely computed.

4.5 Dynamics

- The self-diffusion coefficient, D, is computed from the mean square displacement (msd) function, outside of the initial ballistic regime, via the Nernst-Einstein equation:

$$D = \frac{\lim \mathrm{msd(t)}}{6\ t} \tag{28}$$

$$\mathrm{msd(t)} = \langle(\vec{r}(t) - \vec{r}(0))^2\rangle \tag{29}$$

  where $\langle\ \ \rangle$ means the average over the number of particles and over the number of time origins. D can also be obtained from the velocity autocorrelation function mentioned above.

- Other evaluations:
  The time evolution of the density (i.e. ρ versus the simulation time) is particularly useful to determine whether the thermodynamical equilibrium has been reached and if so, after how many time steps. Studying the time evolution of the structure factor $S(q)$ in the simulation also gives information about the structural changes during the simulation. In the same vein, the time evolution of the number of clusters, of the size of the largest cluster, of the distribution of the sizes may yield useful information.

  Other possible investigations involve the time evolution of the number of subclusters and of their size distribution. It may also be useful to plot the rdfs at different times or over different periods of the simulation as well as the time evolution of specific distances such as the end-to-end distance $d_{\mathrm{n-n}}$. The VMD software allows to construct movies to give a more visual impression of the evolution in the system.

5 Models

5.1 Morphology

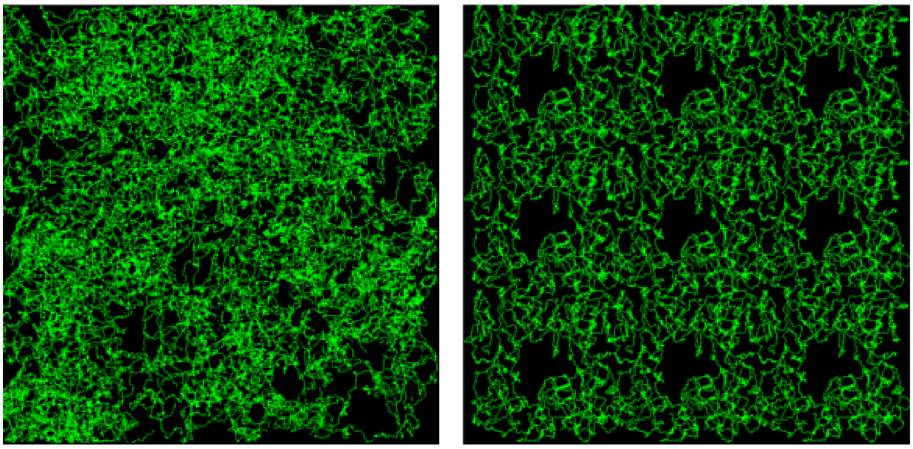


Figure 9: Random (left) and cylindrical (right) morphological models of Nafion. Only the polymer backbone is shown.

Fig. 9 shows two configurations of Nafion, obtained as described in this work, to illustrate the two typical morphological models of Nafion/Hyflon that we study in this thesis. The major difference between the two morphologies is the presence of contiguous free volume (black spots in fig. 9, right). These free volumes basically have a cylindrical structure along the z-axis.

We have tried to generate the polymer chains with as few constraints as possible except the ones inherent in the molecular structures and the size imposed by the required periodicity of our systems (see section 3.2.1). The generation of such structures with as little bias as possible is not trivial; the algorithm devised for this purpose will be described below (see also fig. 18). We call the model on the left 'random' because we initially distributed the water molecules and counterions homogeneously in the volume of the box not occupied by the polymer. We call the model on the right 'cylindrical' because we initially distributed the water molecules and counterions with a higher density in the cylindrical canals.

Six morphological models of hydrated Nafion: the cluster-channel model by Gierke et al. [4], the parallel cylinder model by Schmidt-Rohr and Chen [3], the local order (or hard sphere) model by Gebel et al. [68], the lamellar model by Litt [73], the rod network model by Kreuer [160], and a 'random' model, have been examined by Voth and Knox [2], as seen in section 2. The parallel cylinder model by Schmidt-Rohr and Chen [3] is the newest morphological model of Nafion to-date (see section 2).

It consists of an array of non-overlapping parallel water channels surrounded by the Nafion polymer. We also note that no connecting bridge structures have been proposed in this model to explain the observed percolation threshold in Nafion.

In the simulation study by Voth and Knox [2], each system was built to approximate one of these proposed models with respect to the structure of the hydrophilic domains. In other words, the purpose of these simulations was to study the behavior of the assumed model structures rather than to predict the true structure. Consequently, each system was build at a density close to the experimental one. This must, however, result in a high degree of entanglement for the polymer chains. Considering the very large time-scale and/or the very large amount of energy necessary for the backbone chains to disentangle, such generated structures are likely to be never-changing.

The characteristic experimental scattering peak (the so-called ionomer peak) was observed for all initial configurations [2] after some annealing (to relax and rearrange the polymer chains only, the positions of the water molecules and of the hydronium ions were fixed in order to preserve the assumed initial structures). This is true except for their so-called random model (not identical to the one discussed here), i.e. the only model that does not assume a particular initial structure. With respect to the random model, these authors further stated (verbatim): "Much longer time scales, in theory, would change this model into an accurate depiction of true Nafion, but such time scales are not computationally feasible."

Going beyond the conclusion of Voth and Knox, we have chosen to study such a random model. We expect our 'random' model, i.e. the model with randomly placed water molecules in a box of Nafion polymer, to be more likely to evolve into the true structure of Nafion without applying additional constraints. The main difference with the approach by Voth and Knox is that we did not build our systems at a density close to the final one. Instead, not to start with entangled chains, our systems were built at a density below the experimental reference.

In addition to the random model, we studied a model similar to the one by Schmidt-Rohr and Chen [3]. In contrast to the simple picture put forward by these authors, i.e. the existence of domains where the water density is strictly zero, we distributed the water molecules and counterions inside and outside predefined cylindrical canals, with a low density outside and a high density inside the canals. Such a distribution seems to us to be more realistic than the highly ordered one proposed by Schmidt-Rohr and Chen.

5.2 Systems, Force Field, and Simulation Details

5.2.1 Systems

The different systems simulated in this work are described in table 1. 'box' in table 1 means the average (final) edge length of the cubic simulation box. ρ is the average density (averaged over the last 10 ns of the simulation; the statistical error has been estimated using the RMSD function, see section 4.2). The 6 first systems have similar initial geometries (random) and water content (λ=10), they differ only in the initial positions of the atoms. Also, the behavior of the first system was examined over a longer simulation period (see table 1). We then changed the water content, (systems 7/8 and 9/10), from the last configurations of the simulations of systems 1 and 2. Systems 7 and 9 are generated from system 1, while system 8 and 10 are generated from system 2. System 11 has a cylindrical starting morphology. Finally systems 12 and 13 are simulations of Hyflon. In total 13 systems of about 350000-425000 atoms each have been simulated over about 15-40 ns each. To our knowledge, these simulation systems are, after the ones by Voth and Knox [2], the only fully atomistic ones containing many domains.

We define a nomenclature, given in the last column of table 1, to identify our systems with a meaningful abbreviation: the first letter, N or H, means Nafion or Hyflon. The second letter refers to the initial geometry: R and C stand for random and cylindrical, respectively. The first number is the water content and the second number is a version index indicating different initial positions of the atoms. Hence, the name 'NR-10-1' and 'NR-10-6' are the first and the sixth Nafion systems with the random initial geometry and a water content $\lambda = 10$. The initial structures of systems 7/8 and 9/10 are classified as 'random' even though they were prepared differently and result from the simulations of systems 1 and 2.

Except for the system NC-10-1, each system is composed of 200 polymer chains (each containing 20 side chains). There are thus 4000 SO_3^- groups and as many K^+-ions. Depending on the water content, there are between 20000 ($\lambda = 5$) and 40000 water molecules ($\lambda = 10$). The system NC-10-1, on the other hand, contains 216 polymers, hence 4320 (216*20) SO_3^- groups, 4320 K^+, and 43200 H_2O.

The choice of K^+ as counterion is arbitrary, since it has been shown that the nature of the cation should not be, at the expected level of accuracy, of particular importance for the study of the nanoscale morphology [1, 4, 63]. In the literature on simulation studies of Nafion, the cations H_3O^+, Na^+, and K^+ are often used. The choice of the hydronium ion would be interesting, however, it might slow down the computations compared to the Na^+- and K^+-ions (see also [161]).

System	Ionomer	Initial geometry	λ	# atoms	box (nm)	ρ ($\mathrm{kg \cdot m^{-3}}$)	t_{sim} (ns)	t_{equi} (ns)	t_{col} (ns)	name
1	Nafion	random	10	396000	17.25	1762 ± 0.5	41.5	2.0	39.5	NR-10-1
2	Nafion	random	10	396000	17.26	1759 ± 0.4	25.2	2.0	23.2	NR-10-2
3	Nafion	random	10	396000	17.27	1758 ± 0.6	23.5	2.0	21.5	NR-10-3
4	Nafion	random	10	396000	17.26	1759 ± 0.5	25.2	2.0	23.2	NR-10-4
5	Nafion	random	10	396000	17.27	1758 ± 0.4	23.3	2.0	21.3	NR-10-5
6	Nafion	random	10	396000	17.26	1759 ± 0.5	25.1	2.0	23.1	NR-10-6
7	Nafion	random	7.5	366000	16.9	1804 ± 0.4	15.0	1.0	14.0	NR-7.5-1
8	Nafion	random	7.5	366000	16.9	1803 ± 0.3	15.0	1.0	14.0	NR-7.5-2
9	Nafion	random	5	336000	16.6	1846 ± 0.4	15.0	2.0	13.0	NR-5-1
10	Nafion	random	5	336000	16.6	1847 ± 0.7	15.0	2.0	13.0	NR-5-2
11	Nafion	cylinder	10	427680	17.7	1756 ± 0.6	15.6	2.0	13.6	NC-10-1
12	Hyflon	random	10	356000	16.6	1751 ± 0.4	23.9	6.0	17.9	HR-10-1
13	Hyflon	random	10	356000	16.6	1752 ± 0.5	22.5	6.0	16.5	HR-10-2

Table 1: Simulation systems.

In contrast with the study by Voth and Knox [2], we did not consider the randomicity of the comonomeric sequence; the side chains are periodically separated by 15 CF_2 groups ($x = 7$ in eq. 1). According to eq. 1, it corresponds to an EW of about 1150 $g \cdot mol^{-1}$. In the language of the study of the effect of the monomeric sequence by Jang et al. [90], which suggests an influence on the nanophase segregation, our model corresponds to the "dispersed" sequence.

We used the software GROMACS [129] in its parallelized form and the Cray XT6 machine of the University of Duisburg-Essen that became available in June 2010. With 96 CPUs, 20 ns of simulation of a Nafion system of about 400000 atoms took about two-three days. A plot of the performance (in $ns \cdot day^{-1}$) as a function of the number of CPUs, for a system (similar to Nafion, i.e. $C_4F_9SO_3^-K^+$, in water) of about 200000 atoms, is shown in fig. 10. Based on this test, we choose to simulate the Nafion systems with 96 CPUs, and 24 of the 96 CPUs were attributed to the particle mesh Ewald summation [162, 163], an algorithm similar to the Ewald summation [144], but with an improved performance on the reciprocal sum (more specifically the computation cost of the Ewald and the particle mesh Ewald summation increases as $N^{3/2}$ and $N \log N$, respectively).

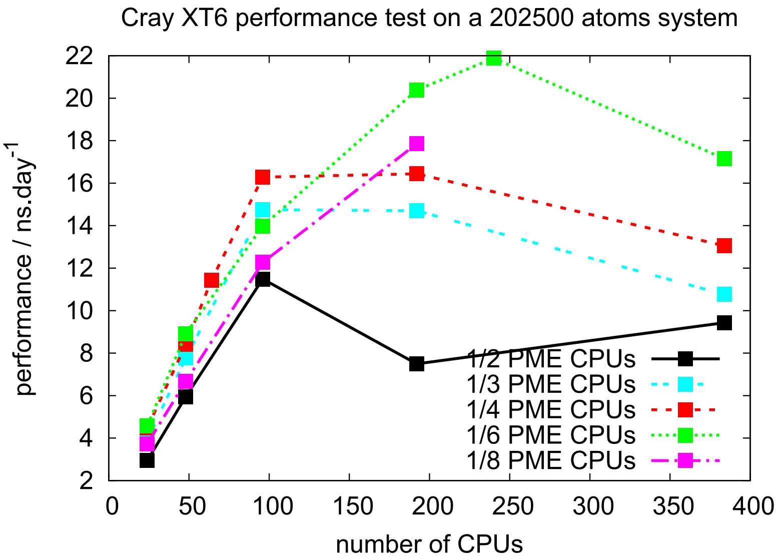


Figure 10: Simulation performance test on the Cray XT6 machine of the University of Duisburg-Essen, for a system similar to the Nafion/Water systems we simulated. In the legend, '1/2 PME CPUs' means that half of the CPUs are attributed to the particle-mesh Ewald summation [162, 163].

5.2.2 Force Field

We have used the inter- and intramolecular potential functions described in section 3.1.1 by eqs. 6, 7, 8, 9, 10, and 11. For Nafion, we used the parameters, and in particular the partial charges, proposed in [93]. All the Nafion parameters (ε_{ii}, σ_{ii}, q_i, k_b, k_α, k_ϕ, r_0, and α_0) are listed in tables 2, 3, and 4 in Appendix 5.A. The SPC water [164] was used and the counterion K^+ was represented by the oplsaa force field [135, 165].

In many simulations studies of Nafion [2, 27, 28, 58, 90, 166], a modified version of the DREIDING force field [90,107,108] was used together with the flexible F3C water model and an explicit hydronium model [90, 109]. This combination was found to reproduce the experimental density satisfactorily [2]. This is also true for the force field that we used for our simulations: the densities obtained after equilibration (see below) are also in agreement with the experimental ones [97]. We note in passing that Dupuis et al. [18], using on one hand the DREIDING [108] and F3C [109] force fields with the DL_POLY [167] simulation code, and on the other hand the AMBER [168, 169] and SPC/E water [99] force fields with the NWChem [170,171] code, found that the general features of the Nafion morphology and the molecular transport in Nafion membranes appear to be independent of the force field and the code used.

Furthermore, our force field also reproduces satisfactorily the frequencies (or at least the frequency range) of the vibrational modes in the Nafion (or Hyflon) molecule, and also of the hindered translational and rotational (librational) motions of the water molecule. This will be shown in chapter 6 dedicated to the results. A difference with other Nafion force fields is that, in our force field, there are no partial electric charges on the carbon (C), fluorine (F), and ether oxygen (Oe) atoms. The sulfonate atoms only: the sulfurs (S) and the sulfur oxygens (Os), carry such charges. Together with the choice of the K^+-ion instead of hydronium, the computation with the present force field should be faster than the ones with force field involving more partial charges.

5.2.3 Simulation Details

In this work, all systems were simulated in the (NpT) ensemble, at 1 bar and 300K. Therefore, the Berendsen thermostat [147] and barostat [147] implemented in gromacs were used to keep the temperature and the pressure constant. We used a coupling time constant of 1 ps for both the thermostat and the barostat. For the long-range interactions, we used the particle mesh Ewald method [162,163] with a mesh spacing of 1.2 Å (see the gromacs manual [129]). The short-range interactions in the direct space were truncated beyond a cut-off distance r_c=10 Å, using the shifted-force method for the forces to decay smoothly to zero between 5 and 10 Å. To speed up the calculations, we used the Verlet neighbor list algorithm with a radius r_list=13 Å. The list was updated every 10 simulation steps.

In contrast to the water molecule (rigid SPC), no particular constraints on positions, bonds, or angles were used for the polymer. In accordance with the highest frequency of motions in the system, a 2.0 fs value for the integration time step δt (see section 3.2.5) was used. It will be shown in the section 6.3 that such a value is compatible with a 'flexible' polymer and a rigid water. However, for a flexible water model, it would be too large.

We used periodic boundary conditions (PBC) in all directions in order to get rid of surface effects and to simulate a bulk phase (see section 3.2.1). Also, the initial atomic velocities were attributed according to a Maxwell-Boltzmann distribution, and the trajectory was recorded on disk every 5000 steps (i.e. every 10 ps).

5.3 Initial Geometries

We now describe the main ideas of the algorithm that we designed for the construction of the initial geometries of our systems. The full details are found in Appendix 5.B, fig. 18. The code, which can also serve as a basis for the generation of any AB_2 polymer (e.g. CF_2, CH_2) or any AB_2 polymer with a particular side chain, is also given in this appendix. A version for the AB_2 polymer (without side chains) is given as well.

5.3.1 Random Model

The guiding principle is to grow first the polymer chains with as few constraints as possible, which entails a priori a large volume, i.e a low density. However, starting the simulation from such a low density would require very long simulation times to reach the proper density and/or the need of a strong equilibration procedure (e.g. using a high external pressure) to 'shrink' the simulation box. Furthermore, with such a fast procedure, one would risk to 'freeze' the system in a particular structure. With the present algorithm, it is possible to place the polymer at such a density that these problems can mostly be avoided. In our algorithm, detailed below, the side chains are not explicitly prevented from getting entangled. Nevertheless, as shown in an example in fig. 13, adjacent polymer strands tend not to be much entangled with their neighbors.

5.3.1.1 Growing the Polymer

We generated the CF_2 backbones from random starting points in the box, avoiding overlaps, and with random growth paths subject only to the geometrical constraints (interatomic bond lengths and angles). By using a unique 'seed' (according to the system time of the machine) to generate each random number, each chain in each system is unique, having a different starting point and a different path.

Considering the first and the last carbon atoms of the backbone as the head and the tail of the chain, and a C-C-C angle of 109.47°, a stretched polymer chain containing 20 side chains, periodically separated by 15 CF_2 groups, has a maximum length of about 19 nm. This is larger than the values that we can expect for the edge lengths of 'affordable' periodic simulation boxes (see table 1). However, in order to preserve the periodicity in our systems, the linear extension of the polymers must not exceed the size of the simulation box during the growth process. In order to limit the size of the polymer chains, we prevent the backbone chains from growing outside the box. Therefore we add, as an additional constraint, artificial rigid boundaries at the faces of the box. Thus, whenever the growing backbone approaches a face, it must orient its path in order to find its next positions inside the box. The effect of adding such a constraint on the growth process can be seen in fig. 11 with two Nafion chains: whenever a chain approaches one of the face of the simulation box, it proceeds in a u-turn.

Another way to limit the size of the polymers without adding artificial walls would have been to limit their growth (by a u-turn) at some arbitrary head-to-tail distance smaller than (or not much larger than, see the compression procedure described below) the expected edge lengths. It would also have been possible to displace individual chains randomly in one direction. This might, however, have created new overlaps.

From a computational standpoint, adding the pendant side chains once all the CF_2 backbones are positioned seems to be easier than adding them during the growth process of the backbone. However, this turned out to be practicable only at very low densities, yet starting from such a density should be avoided (see above). Therefore, the growth process of each backbone was interrupted every 15 CF_2 groups to add a side chain. The growth process of a polymer chain is illustrated in fig. 11 with two Nafion chains.

Growing the polymer chains at low densities is not a problem, as the probability for the current growing chain to overlap with other chains, or with itself, is low. If it by chance nevertheless happens, there is enough free space in the box to find an alternative random growth path without overlaps. However, at higher densities, sooner or later a growing chain does not find enough free space to keep growing without overlapping with other chains or with itself. To face this problem, we shortened after a maximum number of growth attempts the chain to one of its previous point by removing monomeric sequences (i.e. we removed backbone atoms up to the last side chain). A new growth path, starting from this point, was then attempted. This way of proceeding was found to be adequate to generate random polymer structures without overlaps and at densities below, but close enough to the experimental one. Furthermore, we can easily generate as many different initial structures with identical properties as we wish.

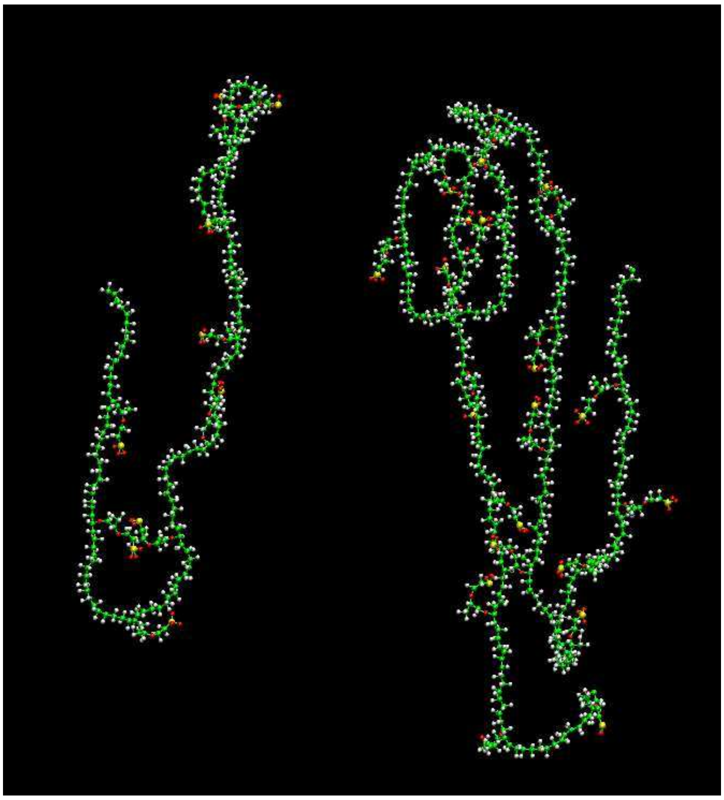


Figure 11: Random growth process of the CF_2 backbones and side chains: two Nafion molecules, the left one still growing while the right one was grown in a previous step and is already finished. The C-, F-, O-, and S-atoms are shown in green, white, red, and yellow, respectively. In this example, the molecules are not entangled, which, however, is not prevented by the algorithm. Both chain made u-turns near the box boundary.

In our random Nafion systems, 200 polymer chains were thus placed in this way in a $20 \times 20 \times 20$ nm^3 cube, leading to a polymer density of about 950 kg $\cdot$ m^{-3}, which is about half the experimental density of dry Nafion. For the Hyflon, a $30 \times 30 \times 30$ nm^3 cube was used as a test, even though, with the shorter side chains, the problem of entanglement should be less severe here. Compared e.g. to the NR-10 Nafion systems, under equal conditions (p=1 bar, T=300 K), a longer equilibration phase was, however, required to reach the final stable density. This is shown at the end of this chapter.

The 200 Nafion (or Hyflon) chains placed in the simulation box as described above were then submitted to a standard energy minimization in order to remove the 'defects' induced by our geometrical construction. An example of a structure obtained before and after the energy minimization is shown in fig. 12 for a Nafion chain. After the energy minimization, the chain seems 'chemically' more realistic.

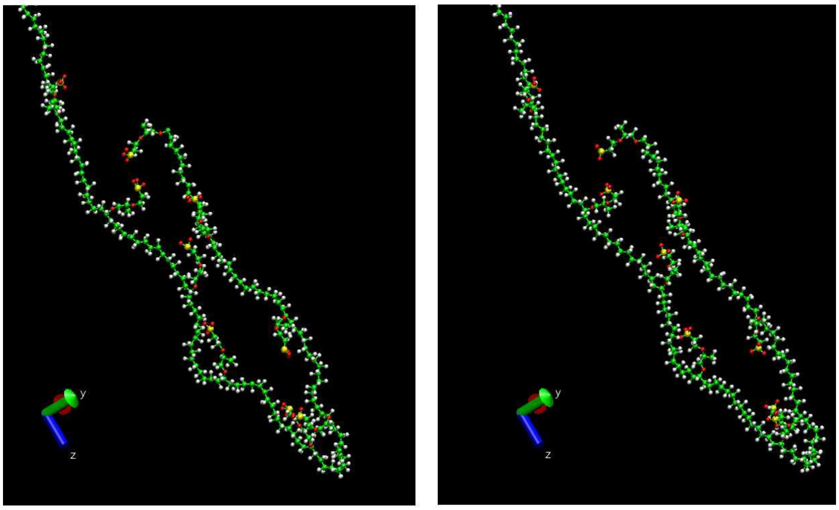


Figure 12: Structures of a Nafion chain before (left) and after an energy minimization run. The C-, F-, O-, and S-atoms are shown in green, white, red, and yellow, respectively.

A resulting structure of 200 polymer chains in a $20 \times 20 \times 20$ nm^3 box (after the same energy minimization) is shown in fig. 13. Looking onto the xy plane (fig. 13, left), one does not see any particular geometry but rather a random network of chains that are more or less entangled. In contrast, when looking at the faces in the xz (fig. 13, center) and yz (fig. 13, right) planes, one sees a more compact structure of elongated chains along the z direction, with the chains arranged like 'laces'. Such an arrangement is due to the narrow u-turns imposed for the backbone chains when approaching the boundaries in the xy plane (see above).

Before adding the water molecules and the K$^+$-ions, the polymer was simulated for 0.5 ns in the (NpT) ensemble, at room temperature and pressure. This resulted in slightly higher densities, shrinking the boxes (imposing a cubic box) from $20 \times 20 \times 20$ nm^3 to about $19 \times 19 \times 19$ nm^3 for Nafion and from $30 \times 30 \times 30$ nm^3 to about $29.9 \times 29.9 \times 29.9$ nm^3 for Hyflon. This mild compression did not, as expected, perturb the initial random structure in an appreciable fashion.

5.3.1.2 Placing Water Molecules and Counterions

We then filled the volume of the box not occupied by polymer particles with water molecules. We used for this purpose the GROMACS software, for details, see Ap-

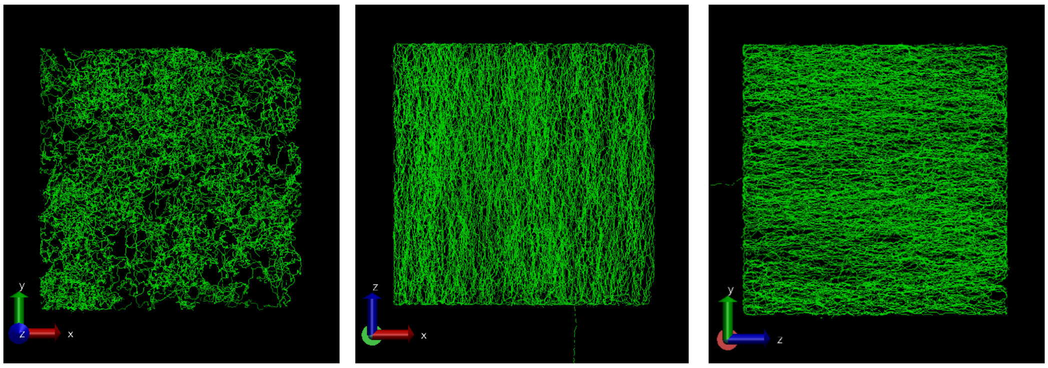


Figure 13: 200 Nafion chains in a $20 \times 20 \times 20$ nm^3 cube. The x, y and z axes are shown at the bottom left. The C atoms of the backbone are shown only.

pendix 5.B. The drawback of this approach is that this code is able to fill the box homogeneously with water molecules only at the density of liquid water. The number of water molecules to be generated can be limited (e.g. according to the desired λ-value), however, in this case, voids will appear in the box between domains with bulk density. We thus first filled the entire box with as many water molecules as it took. We then removed water molecules, randomly and homogeneously over the box, until the desired number (e.g. 40000 at λ=10) remained. 4000 more water molecules were retained and transformed into K$^+$-ions. An initial distribution of water molecules and K$^+$-ions in the xy plane of the simulation box is shown in fig. 14. No significant difference is observed in the xz and yz planes.

Such a distribution is typical for the systems hydrated at λ=10 (i.e. the systems NR-10-1, NR-10-2, NR-10-3, NR-10-4, NR-10-5, NR-10-6, HR-10-1, HR-10-2). Fig. 14 shows domains where the density of water molecules and K$^+$-ions is zero (black regions). This is a direct consequence of placing the polymer first. Except this constraint, no particular structure can be seen: the water molecules and the K$^+$-ions are distributed randomly around the polymer. Furthermore, despite the domains where the average water density is very low, i.e. the polymer, the network formed by the water molecules and the K$^+$-ions seems to percolate.

After adding to the simulation box the 4000 K$^+$ and 40000 water molecules (for the systems with λ=10), the density increased to about 14328.8 kg $\cdot$ m^{-3}, which is lower than the experimental density ($\approx$ 1700 kg $\cdot$ m^{-3} [97]). To generate the systems with a lower water content (NR-7.5-1, NR-7.5-2, NR-5-1, NR-5-2), we removed in the same way the appropriate number of water molecules from the last configurations of the simulations of the systems NR-10-1 and NR-10-2.

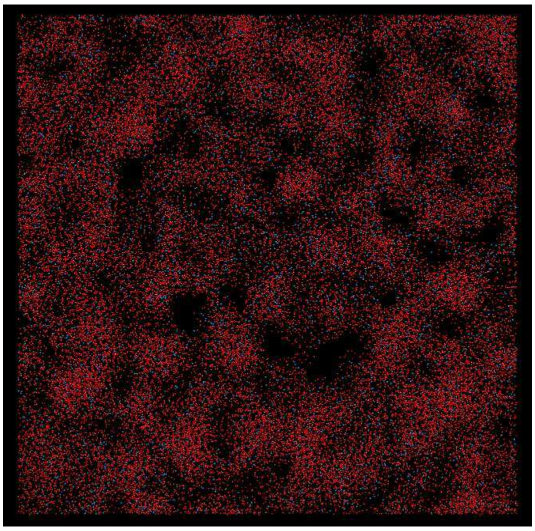


Figure 14: Random distribution of 40000 water molecules and 4000 K^+ in a system containing 200 polymer chains. The snapshot is taken in the xy plane, no significant difference is observed in the xz and yz planes. The box is about $19 \times 19 \times 19$ nm^3 large. The water oxygens and K^+ are shown only, in red and blue, respectively.

5.3.2 Cylindrical Model

In contrast to the random model, we built our cylindrical initial system at a density close to the experimental one: Starting the simulation from a low density would have probably 'destroyed' the initial cylindrical structure when compressing the system. We first placed 8 Nafion chains in a $6 \times 6 \times 6$ nm^3 box with the same growth algorithm as for the random systems but with the additional constraint that the backbone of the polymer can not come a distance r_{cyl} closer to the center of the box in the xy plane, r_{cyl} being the radius of the cylinder along the z direction. This is illustrated in fig. 15. The polymer density of this small system is ≈ 1406 kg $\cdot$ m^{-3}. According to Schmidt-Rohr and Chen [3], the water channels have diameters of between 1.8 and 3.5 nm with an average of 2.4 nm. In our model we have chosen 1.5 nm as r_{cyl}-value.

This small system was then submitted to a standard energy minimization and replicated in the x, y, and z directions in order to construct a system 27 times larger (i.e. 216 polymer chains in an $18 \times 18 \times 18$ nm^3 box). This large system is shown in fig. 16. As the faces in the xz and yz planes look similar, we show only one face in the xz plane (fig. 16, right). Nine identical 'cylinders', or canals, along the z direction, can be seen by looking at the xy plane (fig. 16, left).

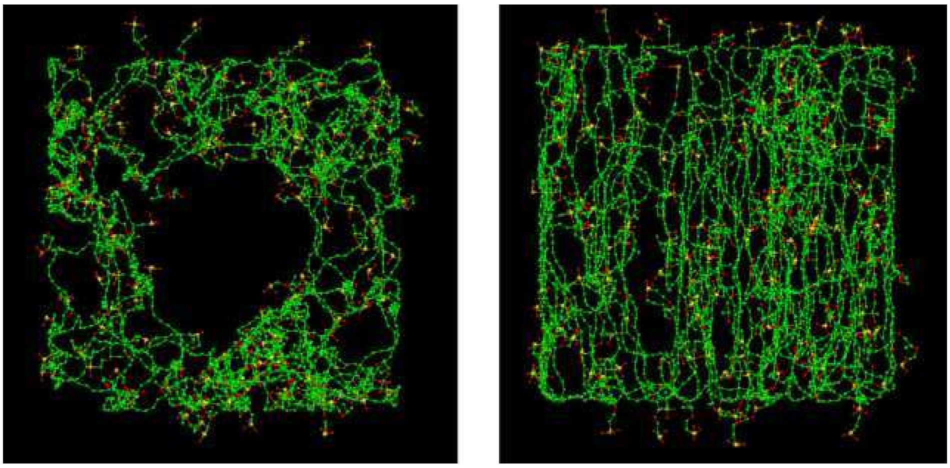


Figure 15: Small cylindrical system (8 Nafion chains in a $6 \times 6 \times 6$ nm^3 box). Left: face in the xy plane; right: face in the xz plane; The faces in the yz plane look similar to the ones in the xz plane. The C-, O-, and S-atoms are shown only, in green, red, and yellow, respectively.

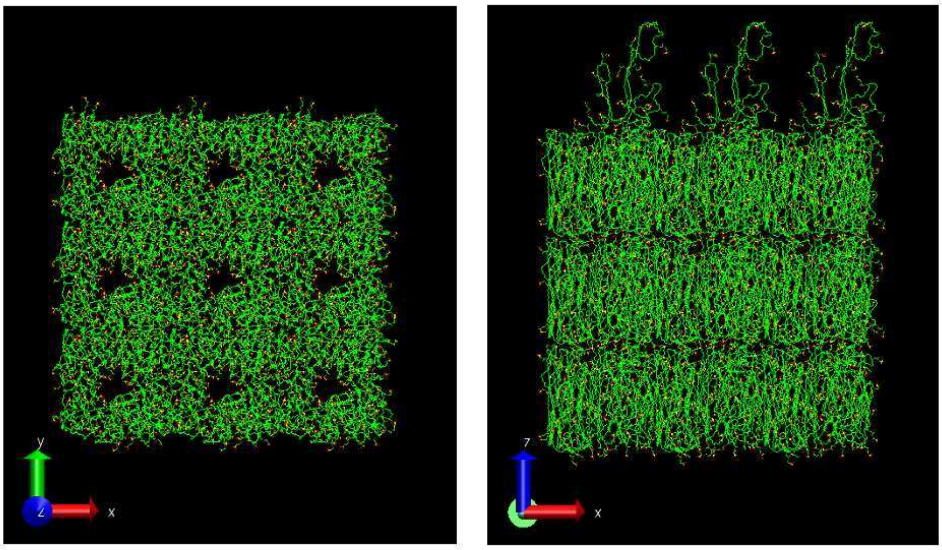


Figure 16: Initial positions of the polymer in the cylindrical system (216 Nafion chains in a $18 \times 18 \times 18$ nm^3 box). The x, y, and z axes are shown at the bottom left. The C-, O-, and S-atoms are shown only, in green, red, and yellow, respectively.

We then filled the $18 \times 18 \times 18$ nm^3 simulation box with water (at λ=10) and K$^+$-ions in the same way as for the random model, and we expected a density for the water molecules and the K$^+$-ions higher inside than outside the canals. 43200 water molecules and 4320 K$^+$-ions were thus added to the system, leading to a total density of ≈ 1674 kg $\cdot$ m^{-3}. In average, ≈ 40 % of the water molecules and ≈ 34 % of the K$^+$-ions are found inside the canals, which leads to a local density of the water molecules and K$^+$-ions ≈ 423 kg $\cdot$ m^{-3}, while the average density is ≈ 269 kg $\cdot$ m^{-3}.

5.4 Equilibration and Data Collection

All systems were then simulated in the (NpT) ensemble at room temperature and pressure. The simulation lengths for each system are given in table 1. Fig. 17 shows for various systems the evolution of the density of the Nafion and Hyflon systems (see table 1) during the equilibration and simulation. The curves for the NR-10 systems are similar, therefore we show here only one curve for these systems. Since the NR-10-1 system has the longest simulation length, we show the curve for the system NR-10-1. Similarly, there is no significant difference when comparing with each others the NR-7.5 systems, the NR-5 systems, and the HR-10 systems, and we show only one curve for each of these typical systems. The curves not shown here (i.e. for the systems NR-10-2, NR-10-3, NR-10-4, NR-10-5, NR-10-6, NR-7.5-2, NR-5-2, and HR-10-2) are shown in Appendix 5.C, fig. 19. The final (average) density of each system is given in table 1.

After a short equilibration phase (the first nanoseconds), the densities of the random Nafion systems stopped increasing and became stable, within less than 1 %, for the rest of the simulation (see fig. 17). For an equilibrated (in the (NpT)-ensemble) system, the density should oscillate around a constant value. However, as it can be seen from the zoom on the curve for the NR-10-1 system, the densities were still slightly increasing even after the short equilibration phase, and until the end of the simulation. This is almost certainly due to the long, entangled, and glassy backbones. Equilibrating the whole system completely would unfortunately require very long simulation times: some microseconds or even seconds.

As expected, the density is larger when the water content is lower. This is seen in fig. 17 by comparing the NR-10 with the NR-7.5 and the NR-5 systems. After this short equilibration phase, the present ρ-values are in agreement with the ones obtained experimentally [97].

The densities of the Hyflon systems (HR-10-1 and HR-10-2), starting from a larger volume, only became stable after about 5 ns. Despite the shorter side chains, Hyflon has the same density as Nafion, after equilibration. Fig. 17 also shows that the Nafion system with the cylindrical morphology (NC-10-1) and the Nafion systems with the random morphologies (NR-10) have similar densities. As expected, system NC-10-1 started from a density close to the experimental one, and the equilibration phase was even shorter than for the Nafion random systems.

For each system, once the density was stable and in agreement with the experiment, data were collected over the rest of the simulation. The decomposition for all systems of the simulation lengths into the time for equilibration and data collection phases, respectively, is shown in table 1. t_{sim}, t_{equi}, and t_{col} in table 1 are the simulation, equilibration, and data collection lengths, respectively.

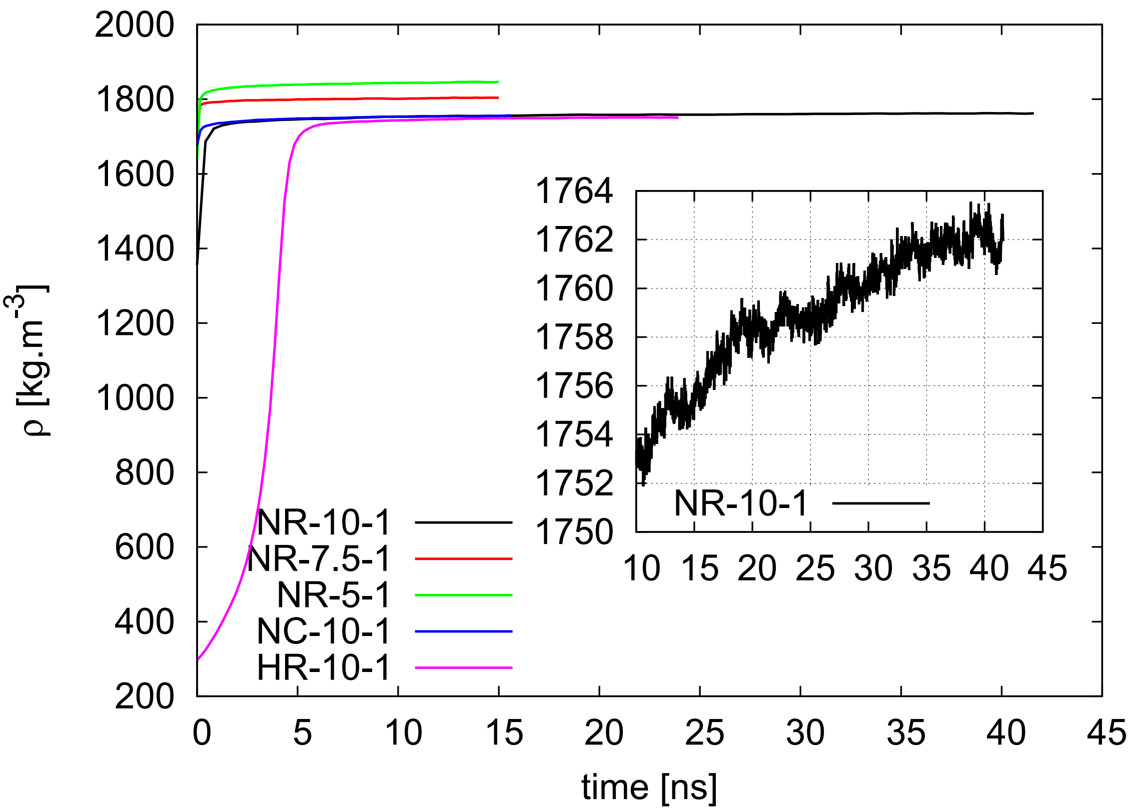


Figure 17: Total density vs. the simulation time. The curves have been smoothed with a Bezier function. The insert shows a zoom at times after equilibration.

5.A Appendix: Force Field Parameters

	ε [kJ $\cdot$ mol^{-1}]	σ [Å]	q [e]
C	0.3981	3.473	0.00
F	0.3035	3.093	0.00
Oe	0.7117	3.070	0.00
S	1.0465	3.550	1.19
Os	0.8372	3.150	-0.73

Table 2: Lennard-Jones parameters and partial charges.

	k_b [kJ $\cdot$ mol^{-1} $\cdot$ Å^{-2}]	r_0 [Å]
C-C	2928.8	1.54
C-Oe	2928.8	1.54
C-F	2928.8	1.37
C-S	2928.8	1.80
S-Os	2928.8	1.49

Table 3: Stretch parameters.

	k_α [kJ $\cdot$ mol^{-1} $\cdot$ rad^{-2}]	α_0 [deg]
C-C-C	471.45	109.47122
C-C-F	472.04	109.47122
F-C-F	470.70	109.47122
S-C-C	490.91	109.47122
S-C-F	478.15	109.47122
Os-S-C	456.31	109.47122
Os-S-Os	509.36	109.47122
Oe-C-C	470.70	109.47122
C-Oe-C	470.70	109.47122
Oe-C-F	470.70	109.47122

Table 4: Bend parameters.

All torsion terms: $k_\phi = 0.9297$ kJ $\cdot$ mol^{-1} $\cdot$ rad^{-2}

5.B Appendix: Algorithm and Codes for the Initial Geometries

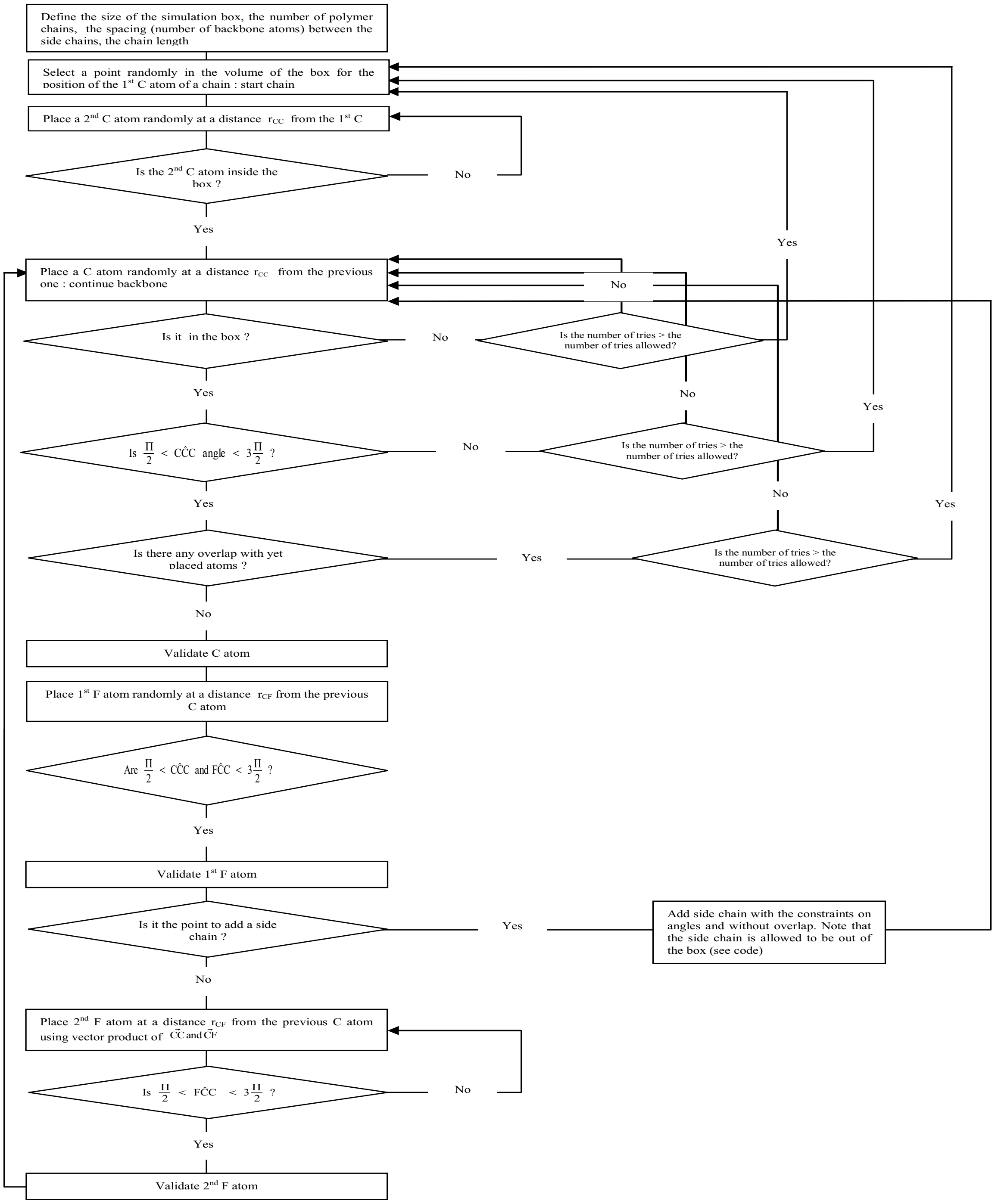

Define the size of the simulation box, the number of polymer chains, the spacing (number of backbone atoms) between the side chains, the chain length
Select a point randomly in the volume of the box for the position of the 1st C atom of a chain : start chain
Place a 2nd C atom randomly at a distance rCC from the 1st C
Is the 2nd C atom inside the box ?
No
Yes
Place a C atom randomly at a distance rCC from the previous one : continue backbone
No
Is it in the box ?
No
Is the number of tries > the number of tries allowed?
Yes
Yes
No
Is Π/2 < CĈC angle < 3Π/2 ?
No
Is the number of tries > the number of tries allowed?
Yes
Yes
No
Yes
Is there any overlap with yet placed atoms ?
Yes
Is the number of tries > the number of tries allowed?
No
Validate C atom
Place 1st F atom randomly at a distance rCF from the previous C atom
Are Π/2 < CĈC and FĈC < 3Π/2 ?
Yes
Validate 1st F atom
Is it the point to add a side chain ?
Yes
Add side chain with the constraints on angles and without overlap. Note that the side chain is allowed to be out of the box (see code)
No
Place 2nd F atom at a distance rCF from the previous C atom using vector product of CC and CF
Is Π/2 < FĈC < 3Π/2 ?
No
Yes
Validate 2nd F atom

Figure 18: algorithm for the growth process of a Nafion/Hyflon chain

code for the building of the Nafion/Hyflon chains

```
/////////////////////////////////////////////////////////////////////////////////////////////////////////////
//                                                                                                         //
//   This code generates a configuration of Nafion polymer chains (default choice is to place 200 Nafion   //
//   chains in a 20 * 20 * 20 nm box; change BOX and moleculemax values to desired values).                //
//   The polymer chains have random starting points and each chain has a unique random path                //
//   (see the algorithm in the thesis manuscript for more details).                                        //
//   The coordinates are saved in the specific formats configuration .xyz and .gro files                   //
//   Executing x times the code will give x different (in the positions of the atoms) configurations.      //
//                                                                                                         //
/////////////////////////////////////////////////////////////////////////////////////////////////////////////

#include <cstdio>
#include <stdlib.h>
#include <math.h>
#include <time.h>
#include <iostream>

using namespace std;

float BOX=200.0; // edge of the cubic box (angstrom)
int side_chain_carbon=48; // number of C backbone atoms between side chains * 3
int sidechainmax=20; // number of side chains per chain
int atommax=9+sidechainmax*side_chain_carbon; // number of backbone atoms per chain
int side_chain_per_molecule= atommax/side_chain_carbon;

int moleculemax= 200; // number of nafion chains
float X[3][1600][200]; // coordinates
float end_to_end[200]; // end-to-end distances
float rcc=1.540; // bond between two C
float rcf=1.370; // bond between C and F
float rco=1.540; // C-O bond
float rcs=1.800; // C-S bond
float ros=1.490; // S-O bond
int outofthebox; // 0 if atom in the box, 1 if atom outside the box
int overlap; // 0 if no overlap, 1 if any overlap

// give random coordinates to first atom of a chain
int randomize_first_atom (int atom, int molecule){
  for (int coordinate=0 ; coordinate <3 ; coordinate++){
    X[coordinate][atom][molecule]= ( rand() / (float)RAND_MAX ) * (BOX-0.0) + 0.0;
  }
  return(0);
}

// place atom1 randomly at a distance rho from atom2
int place_atom (int atom1, int atom2, int molecule, float rho){
  float phi= ( rand() / (float)RAND_MAX ) * ( M_PI - 0.0 ) + 0.0;
  float teta= ( rand() / (float)RAND_MAX ) * ( 2.*M_PI - 0.0 ) + 0.0;
  X[0][atom1][molecule]= X[0][atom2][molecule] + rho*sin(phi)*cos(teta);
  X[1][atom1][molecule]= X[1][atom2][molecule] + rho*sin(phi)*sin(teta);
  X[2][atom1][molecule]= X[2][atom2][molecule] + rho*cos(phi);
  return(0);
}

// check whether atom is inside or outside the box
int is_it_in_the_box(int atom, int molecule){
  outofthebox=0;
  if ( (X[0][atom][molecule]<0.0) || (X[0][atom][molecule]>BOX) ||
       (X[1][atom][molecule]<0.0) || (X[1][atom][molecule]>BOX) ||
       (X[2][atom][molecule]<0.0) || (X[2][atom][molecule]>BOX) ){
    outofthebox=1;
  }
  return(0);
}
```

```
// compute angle using scalar product
float scalar_product(int atom1, int atom2, int atom3, int molecule){
  float vec1x= X[0][atom1][molecule]-X[0][atom2][molecule];
  float vec1y= X[1][atom1][molecule]-X[1][atom2][molecule];
  float vec1z= X[2][atom1][molecule]-X[2][atom2][molecule];
  float vec2x= X[0][atom3][molecule]-X[0][atom2][molecule];
  float vec2y= X[1][atom3][molecule]-X[1][atom2][molecule];
  float vec2z= X[2][atom3][molecule]-X[2][atom2][molecule];
  float dist1=sqrt(vec1x*vec1x + vec1y*vec1y + vec1z*vec1z);
  float dist2=sqrt(vec2x*vec2x + vec2y*vec2y + vec2z*vec2z);
  float ralpha = acos( (vec1x*vec2x + vec1y*vec2y + vec1z*vec2z) / (dist1*dist2) );
  return (ralpha);
}

// place atom using vector product
float vector_product(int atom1, int atom2, int atom3, int atom4, int atom5,
                     int molecule, float ralphaccf1, float rho){
  float vec1x= X[0][atom1][molecule]-X[0][atom2][molecule];
  float vec1y= X[1][atom1][molecule]-X[1][atom2][molecule];
  float vec1z= X[2][atom1][molecule]-X[2][atom2][molecule];
  float dist1=sqrt(vec1x*vec1x + vec1y*vec1y + vec1z*vec1z);
  float vec2x= X[0][atom3][molecule]-X[0][atom2][molecule];
  float vec2y= X[1][atom3][molecule]-X[1][atom2][molecule];
  float vec2z= X[2][atom3][molecule]-X[2][atom2][molecule];
  float dist2=sqrt(vec2x*vec2x + vec2y*vec2y + vec2z*vec2z);
  float vec3x= X[0][atom5][molecule]-X[0][atom2][molecule];
  float vec3y= X[1][atom5][molecule]-X[1][atom2][molecule];
  float vec3z= X[2][atom5][molecule]-X[2][atom2][molecule];
  X[0][atom4][molecule]= (vec1y*vec2z - vec1z*vec2y)/(dist1*dist2*sin(ralphaccf1))*rho
  + X[0][atom2][molecule];
  X[1][atom4][molecule]= (-vec1x*vec2z + vec1z*vec2x)/(dist1*dist2*sin(ralphaccf1))*rho
  + X[1][atom2][molecule];
  X[2][atom4][molecule]= (vec1x*vec2y - vec1y*vec2x)/(dist1*dist2*sin(ralphaccf1))*rho
  + X[2][atom2][molecule];
  //take minus the cross product (i.e. opposite direction) if (atom) and (atom-1) are too close
  float vec4x= X[0][atom4][molecule]-X[0][atom2][molecule];
  float vec4y= X[1][atom4][molecule]-X[1][atom2][molecule];
  float vec4z= X[2][atom4][molecule]-X[2][atom2][molecule];
  float dist3=sqrt(vec3x*vec3x + vec3y*vec3y + vec3z*vec3z);
  float dist4=sqrt(vec4x*vec4x + vec4y*vec4y + vec1z*vec4z);
  float ralphaccf3 = acos( (vec3x*vec4x + vec3y*vec4y + vec3z*vec4z) / (dist3*dist4) );
  if ( (ralphaccf3 < M_PI*0.5) || (ralphaccf3 > M_PI*1.5) ){
    X[0][atom4][molecule]= (-vec1y*vec2z + vec1z*vec2y)/(dist1*dist2*sin(ralphaccf1))*rho
    + X[0][atom2][molecule];
    X[1][atom4][molecule]= (vec1x*vec2z - vec1z*vec2x)/(dist1*dist2*sin(ralphaccf1))*rho
    + X[1][atom2][molecule];
    X[2][atom4][molecule]= (-vec1x*vec2y + vec1y*vec2x)/(dist1*dist2*sin(ralphaccf1))*rho
    + X[2][atom2][molecule];
  }
  return (0);
}

// check whether there is any overlap
int count_overlap(int molecule, int atom1, int atom2, int atom3, float rvdw){
  float rvdw2=rvdw*rvdw;
  for (int j=molecule; j>-1; j--){
    for (int i=atom2; i>=atom3; i=i-3){
      float rx=X[0][atom1][molecule]-X[0][i][j];
      float ry=X[1][atom1][molecule]-X[1][i][j];
      float rz=X[2][atom1][molecule]-X[2][i][j];
      float r2 = rx*rx + ry*ry + rz*rz;
      if (r2<rvdw2){
        overlap++;
      }
    }
  }
  return(0);
}

//compute end-to-end distance for each molecule
int end_to_end_distance(int molecule){
  float rx=X[0][atommax][molecule]-X[0][6][molecule];
  float ry=X[1][atommax][molecule]-X[1][6][molecule];
  float rz=X[2][atommax][molecule]-X[2][6][molecule];
  end_to_end[molecule]=sqrt(rx*rx + ry*ry + rz*rz);
  return(0);
}

//print end-to-end distances in a file
int print_end_to_end(){
  FILE * pFile3;
  pFile3 = fopen ("end_to_end.dat","w");
  fprintf (pFile3, "#molecule  end-to-end distance\n");
  for (int i=0; i<moleculemax; i++){
    fprintf (pFile3, "%8d %8.3f\n", i, end_to_end[i]);
  }
  fclose (pFile3);
  return(0);
}

//print coordinates in a .xyz file
int print_xyz_file(){
  FILE * pFile;
  pFile = fopen ("geometry.xyz","w");
  fprintf(pFile, "%d\n", moleculemax*1360);
  fprintf(pFile, "%d\n", moleculemax*1360);
  for (int molecule=0; molecule<moleculemax; molecule++){
    int sidechain=1;
    for (int atom=9 ; atom<atommax ; atom+=3){
      fprintf (pFile, "C %8.3f%8.3f%8.3f\n", X[0][atom-3][molecule],
       X[1][atom-3][molecule], X[2][atom-3][molecule]);
      fprintf (pFile, "F %8.3f%8.3f%8.3f\n", X[0][atom-2][molecule],
       X[1][atom-2][molecule], X[2][atom-2][molecule]);
      if (atom==6+side_chain_carbon*sidechain){
        fprintf (pFile, "O %8.3f%8.3f%8.3f\n",X[0][atom-1][molecule],
                 X[1][atom-1][molecule],X[2][atom-1][molecule]);
        sidechain++;
      }
      else{
        fprintf (pFile, "F %8.3f%8.3f%8.3f\n",X[0][atom-1][molecule],
                 X[1][atom-1][molecule],X[2][atom-1][molecule]);
      }
    }
  }
  fclose (pFile);
  return(0);
}

//print coordinates in a .gro file
int print_gro_file(){
  FILE * pFile2;
  pFile2 = fopen ("geometry.gro","w");
  fprintf(pFile2, "%d\n", moleculemax*1360);
  fprintf(pFile2, "%d\n", moleculemax*1360);
  int residueindex=1;
  char residue_name[] = "na";
  char end_residue_name[1];
  char atomtype_C[] = "C";
  char atomtype_F[]="F";
  char atomtype_Oe[]="Oe";
  char atomtype_Os[]="Os";
  char atomtype_S[]="S";
```

```
for (int molecule=0; molecule<moleculemax; molecule++){
  int sidechain=1;
  int Cindex=1;
  int Findex=1;
  int Oeindex=1;
  int Osindex=1;
  int Sindex=1;
  int atomindex=1;
  for (int atom=9 ; atom<atommax ; atom+=3){
    if (sidechain==1){
      end_residue_name[0]='l';
    }
    else if (sidechain==side_chain_per_molecule){
      end_residue_name[0]='r';
    }
    else{
      end_residue_name[0]='f';
    }
    if (atom==9){
      fprintf(pFile2,"%5d%4s%1c%4s%1c%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
              end_residue_name[0], atomtype_C,'l',atomindex, X[0][atom-3][molecule]*0.1,
              X[1][atom-3][molecule]*0.1, X[2][atom-3][molecule]*0.1);
      Cindex++;
    }
    else if ((sidechain==side_chain_per_molecule) && (Cindex==16)){
      fprintf(pFile2,"%5d%4s%1c%4s%1c%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
              end_residue_name[0], atomtype_C, 'r', atomindex, X[0][atom-3][molecule]*0.1,
              X[1][atom-3][molecule]*0.1, X[2][atom-3][molecule]*0.1);
      Cindex++;
    }
    else{
      if (Cindex<10){
        fprintf(pFile2,"%5d%4s%1c%4s%1d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
                end_residue_name[0], atomtype_C, Cindex, atomindex, X[0][atom-3][molecule]*0.1,
                X[1][atom-3][molecule]*0.1, X[2][atom-3][molecule]*0.1);
        Cindex++;
      }
      else{
        fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
                end_residue_name[0], atomtype_C, Cindex, atomindex, X[0][atom-3][molecule]*0.1,
                X[1][atom-3][molecule]*0.1, X[2][atom-3][molecule]*0.1);
        Cindex++;
      }
    }
    if (Findex<10){
      fprintf(pFile2,"%5d%4s%1c%4s%1d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
              end_residue_name[0], atomtype_F, Findex, atomindex, X[0][atom-2][molecule]*0.1,
              X[1][atom-2][molecule]*0.1, X[2][atom-2][molecule]*0.1);
      Findex++;
    }
    else{
      fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
              end_residue_name[0], atomtype_F, Findex, atomindex, X[0][atom-2][molecule]*0.1,
              X[1][atom-2][molecule]*0.1, X[2][atom-2][molecule]*0.1);
      Findex++;
    }
    if (atom==6+side_chain_carbon*sidechain){
      fprintf(pFile2,"%5d%4s%1c%4s%1d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
              end_residue_name[0], atomtype_Oe, Oeindex, atomindex, X[0][atom-1][molecule]*0.1,
              X[1][atom-1][molecule]*0.1, X[2][atom-1][molecule]*0.1);
      Oeindex++;
      for (int l=0; l<18 ; l+=3){
        if (l==0){
          fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
                  end_residue_name[0], atomtype_C, Cindex, atomindex,
                  X[0][atommax+(sidechain*21)+l][molecule]*0.1,
          X[1][atommax+(sidechain*21)+l][molecule]*0.1,
          X[2][atommax+(sidechain*21)+l][molecule]*0.1);
  Cindex++;
  fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
          end_residue_name[0], atomtype_F, Findex, atomindex,
          X[0][atommax+(sidechain*21)+l+1][molecule]*0.1,
          X[1][atommax+(sidechain*21)+l+1][molecule]*0.1,
          X[2][atommax+(sidechain*21)+l+1][molecule]*0.1);
  Findex++;
  fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
          end_residue_name[0], atomtype_F, Findex, atomindex,
          X[0][atommax+(sidechain*21)+l+2][molecule]*0.1,
          X[1][atommax+(sidechain*21)+l+2][molecule]*0.1,
          X[2][atommax+(sidechain*21)+l+2][molecule]*0.1);
  Findex++;
}
else if (l==3){
  fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
          end_residue_name[0], atomtype_C, Cindex, atomindex,
          X[0][atommax+(sidechain*21)+l][molecule]*0.1,
          X[1][atommax+(sidechain*21)+l][molecule]*0.1,
          X[2][atommax+(sidechain*21)+l][molecule]*0.1);
  Cindex++;
  fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
          end_residue_name[0], atomtype_F, Findex, atomindex,
          X[0][atommax+(sidechain*21)+l+1][molecule]*0.1,
          X[1][atommax+(sidechain*21)+l+1][molecule]*0.1,
          X[2][atommax+(sidechain*21)+l+1][molecule]*0.1);
  Findex++;
  fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
          end_residue_name[0], atomtype_C, Cindex, atomindex,
          X[0][atommax+(sidechain*21)+l+2][molecule]*0.1,
          X[1][atommax+(sidechain*21)+l+2][molecule]*0.1,
          X[2][atommax+(sidechain*21)+l+2][molecule]*0.1);
  Cindex++;
  fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
          end_residue_name[0], atomtype_F, Findex, atomindex,
          X[0][atommax+(sidechain*21)+l+4][molecule]*0.1,
          X[1][atommax+(sidechain*21)+l+4][molecule]*0.1,
          X[2][atommax+(sidechain*21)+l+4][molecule]*0.1);
  Findex++;
  fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
          end_residue_name[0], atomtype_F, Findex, atomindex,
          X[0][atommax+(sidechain*21)+l+5][molecule]*0.1,
          X[1][atommax+(sidechain*21)+l+5][molecule]*0.1,
          X[2][atommax+(sidechain*21)+l+5][molecule]*0.1);
  Findex++;
  fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
          end_residue_name[0], atomtype_F, Findex, atomindex,
          X[0][atommax+(sidechain*21)+19][molecule]*0.1,
          X[1][atommax+(sidechain*21)+19][molecule]*0.1,
          X[2][atommax+(sidechain*21)+19][molecule]*0.1);
  Findex++;
}
else if (l==6){
  fprintf(pFile2,"%5d%4s%1c%4s%1d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
          end_residue_name[0], atomtype_Oe, Oeindex, atomindex,
          X[0][atommax+(sidechain*21)+l][molecule]*0.1,
          X[1][atommax+(sidechain*21)+l][molecule]*0.1,
          X[2][atommax+(sidechain*21)+l][molecule]*0.1);
  Oeindex++;
}
else if (l==9){
  fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
          end_residue_name[0], atomtype_C, Cindex, atomindex,
          X[0][atommax+(sidechain*21)+l][molecule]*0.1,
          X[1][atommax+(sidechain*21)+l][molecule]*0.1,
       X[2][atommax+(sidechain*21)+l][molecule]*0.1);
   Cindex++;
   fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
       end_residue_name[0], atomtype_F, Findex, atomindex,
       X[0][atommax+(sidechain*21)+l+1][molecule]*0.1,
       X[1][atommax+(sidechain*21)+l+1][molecule]*0.1,
       X[2][atommax+(sidechain*21)+l+1][molecule]*0.1);
   Findex++;
   fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
       end_residue_name[0], atomtype_F, Findex, atomindex,
       X[0][atommax+(sidechain*21)+l+2][molecule]*0.1,
       X[1][atommax+(sidechain*21)+l+2][molecule]*0.1,
       X[2][atommax+(sidechain*21)+l+2][molecule]*0.1);
   Findex++;
  }
  else if (l==12){
   fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
       end_residue_name[0], atomtype_C, Cindex, atomindex,
       X[0][atommax+(sidechain*21)+l][molecule]*0.1,
       X[1][atommax+(sidechain*21)+l][molecule]*0.1,
       X[2][atommax+(sidechain*21)+l][molecule]*0.1);
   Cindex++;
   fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
       end_residue_name[0], atomtype_F, Findex, atomindex,
       X[0][atommax+(sidechain*21)+l+1][molecule]*0.1,
       X[1][atommax+(sidechain*21)+l+1][molecule]*0.1,
       X[2][atommax+(sidechain*21)+l+1][molecule]*0.1);
   Findex++;
   fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
       end_residue_name[0], atomtype_F, Findex, atomindex,
       X[0][atommax+(sidechain*21)+l+2][molecule]*0.1,
       X[1][atommax+(sidechain*21)+l+2][molecule]*0.1,
       X[2][atommax+(sidechain*21)+l+2][molecule]*0.1);
   Findex++;
  }
  else if (l==15){
   fprintf(pFile2,"%5d%4s%1c%4s%1d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
       end_residue_name[0], atomtype_S, Sindex, atomindex,
       X[0][atommax+(sidechain*21)+l][molecule]*0.1,
       X[1][atommax+(sidechain*21)+l][molecule]*0.1,
       X[2][atommax+(sidechain*21)+l][molecule]*0.1);
   fprintf(pFile2,"%5d%4s%1c%4s%1d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
       end_residue_name[0], atomtype_Os, Osindex, atomindex,
       X[0][atommax+(sidechain*21)+l+1][molecule]*0.1,
       X[1][atommax+(sidechain*21)+l+1][molecule]*0.1,
       X[2][atommax+(sidechain*21)+l+1][molecule]*0.1);
   Osindex++;
   fprintf(pFile2,"%5d%4s%1c%4s%1d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
       end_residue_name[0], atomtype_Os, Osindex, atomindex,
       X[0][atommax+(sidechain*21)+l+2][molecule]*0.1,
       X[1][atommax+(sidechain*21)+l+2][molecule]*0.1,
       X[2][atommax+(sidechain*21)+l+2][molecule]*0.1);
   Osindex++;
   fprintf(pFile2,"%5d%4s%1c%4s%1d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
       end_residue_name[0], atomtype_Os, Osindex, atomindex,
       X[0][atommax+(sidechain*21)+l+3][molecule]*0.1,
       X[1][atommax+(sidechain*21)+l+3][molecule]*0.1,
       X[2][atommax+(sidechain*21)+l+3][molecule]*0.1);
  }
 }
 sidechain++;
 residueindex++;
 Cindex=1;
 Findex=1;
 Oeindex=1;
 Osindex=1;
}
```

```
      else{
        if (Findex<10){
          fprintf(pFile2,"%5d%4s%1c%4s%1d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
                  end_residue_name[0], atomtype_F, Findex, atomindex,
                  X[0][atom-1][molecule]*0.1, X[1][atom-1][molecule]*0.1, X[2][atom-1][molecule]*0.1);
          Findex++;
        }
        else{
          fprintf(pFile2,"%5d%4s%1c%3s%2d%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name,
                  end_residue_name[0], atomtype_F, Findex, atomindex,
                  X[0][atom-1][molecule]*0.1, X[1][atom-1][molecule]*0.1, X[2][atom-1][molecule]*0.1);
          Findex++;
        }
      }
    }
  }
  fprintf (pFile2, "%8.3f%8.3f%8.3f\n", BOX*0.1, BOX*0.1, BOX*0.1);
  fclose (pFile2);
  return(0);
}

int main(){
  int atom, aatom; // current atom index
  int sidechain; // current side chain index
  int atomsd; // atom of the side chain
  int atomsdmax=21; // number of atoms per side chain
  int molecule; // current molecule index
  int iterationvdw, iterationangle, iterationbox; // tries to fullfill conditions: vdw (overlaps), angle, in the box
  int tolerancebox=20; // number of allowed tries to put the atom in the box before to go rewind in the chain building
  int tolerancevdw=20; // number of allowed tries to put the atom without overlaps with the other atoms
  int toleranceangle=20; // number of permitted tries for an angle

  // set all coordinates to 0
  for (molecule=0; molecule<moleculemax; molecule++){
    for (atom=0; atom<atommax; atom++ ){
      for (int coordinate=0; coordinate<3; coordinate++){
        X[coordinate][atom][molecule]=0.0;
      }
    }
  }

  srand (time (NULL)) ; // random seed taken from system date

  for (molecule=0; molecule<moleculemax; molecule++){
    cout << "molecule: " << molecule << endl;
    REWIND2: // rewind point 2
    randomize_first_atom(0, molecule); // place randomly inside the box the 1st C backbone atom of the chain
    int sidechain=1; // first side chain

    //place 2nd C backbone atom inside the box and at a distance rcc of 1st C
    outofthebox=1; // atom is out of the box
    while (outofthebox!=0){
      float rho=rcc;
      place_atom(3, 0, molecule, rho);
      is_it_in_the_box(0, molecule); // check whether atom is inside or outside the box
    }

    aatom=6;
    REWIND: // rewind point
    // generate C backbone chains
    for (atom=aatom; atom<atommax; atom+=3){
      outofthebox=1; // atom is out of the box
      iterationbox=0;
      while ( outofthebox!=0  ){
        iterationbox++;
        // cout << "not in the box! " << iterationbox << " " << atom << " " << molecule << endl;
        if (iterationbox>tolerancebox){ // number of tries > number of tries allowed
    if (sidechain>=2){
      aatom=6+side_chain_carbon*(sidechain-1);
      sidechain--;
      goto REWIND; // go rewind in the chain buiding process
    }
    else if (sidechain==1){
      goto REWIND2; // go rewind in the chain building process
    }
  }

  float ralphaccc=0.0; // C C C angle
  iterationangle=0;
  while ( (ralphaccc< (0.5*M_PI)) || (ralphaccc> (1.5*M_PI)) ){ // C C C angle must be in the range [pi/2; 3pi/2]
    iterationangle++;
    // cout << "bad angle! " << iterationangle << " " <<  atom << " " << molecule << endl;
    if (iterationangle>toleranceangle){ // number of tries > number of tries allowed
      if (sidechain>=2){
        aatom=6+side_chain_carbon*(sidechain-1);
        sidechain--;
        goto REWIND;
      }
      else if (sidechain==1){
        goto REWIND2;
      }
    }

    overlap=1; // atom overlaps
    iterationvdw=0;
    while(overlap!=0){
      iterationvdw++;
      // cout << "overlap! " << iterationvdw << " " << atom << " " << molecule << endl;
      if (iterationvdw>tolerancevdw){ // number of tries > number of tries allowed
        if (sidechain>=2){
          aatom=6+side_chain_carbon*(sidechain-1);
          sidechain--;
          goto REWIND;
        }
        else if (sidechain==1){
          goto REWIND2;
        }
      }
      place_atom(atom, atom-3, molecule, rcc);
      is_it_in_the_box(atom, molecule);
      ralphaccc = scalar_product(atom, atom-3, atom-6, molecule); // compute ralphaccc angle
      overlap=0; // atom does not overlap
      if (atom>=9){ // check for overlap(s)
        count_overlap(molecule, atom, atom-9, 6, 4.1);
        count_overlap(molecule, atom, atommax+(sidechain*21), atommax, 2.6);
      }
    }
  }
}
// C backbone placed!

//add 1st of the two F on each C atom of the backbone
float ralphaccf1=0.0; // C(atom-6)-C(atom-3)-F(atom-2) angle
float ralphaccf2=0.0; // C(atom)-C(atom-3)-F(atom-2) angle
while ( (ralphaccf1 < 0.5*M_PI) || (ralphaccf1 > 1.5*M_PI) || // angles must be in the range [pi/2; 3pi/2]
        (ralphaccf2 < 0.5*M_PI) || (ralphaccf2 > 1.5*M_PI) ){
  place_atom(atom-2, atom-3, molecule, rcf); // place F atom at a distance rcf from C backbone atom
  ralphaccf1 = scalar_product(atom-2, atom-3, atom-6, molecule); // compute ralphaccf1 angle
  ralphaccf2 = scalar_product(atom-2, atom-3, atom, molecule); // compute ralphaccf2 angle
}

//  place side chain periodically on the backbone chain
if (atom==6+side_chain_carbon*sidechain){
  vector_product(atom-6, atom-3, atom-2, atom-1, atom, molecule, ralphaccf1, rco); // place 1st O ether of
```

```
// the side chain at a distance rco of the C backbone atom, using vector product
for (int l=0 ; l<18 ; l+=3){ // build side chain
  float rho=rcc;
  int atom1=atommax+sidechain*21+l-3;
  int atom2=atommax+sidechain*21+l-6;
  if (l==0){
    rho=rco;
    atom1=atom-1;
    atom2=atom-3;
  }
  else if (l==3){
    rho=rcc;
    atom1=atommax+sidechain*21+l-3;
    atom2=atom-1;
  }
  else if ( (l==6) || (l==9) ){
    rho=rco;
    atom1=atommax+sidechain*21+l-3;
    atom2=atommax+sidechain*21+l-6;
  }
  else if ( (l==12) || (l==18) ){
    rho=rcc;
    atom1=atommax+sidechain*21+l-3;
    atom2=atommax+sidechain*21+l-6;
  }
  else if (l==15){
    rho=rcs;
    atom1=atommax+sidechain*21+l-3;
    atom2=atommax+sidechain*21+l-6;
  }

  outofthebox=1;
  iterationbox=0;
  while (outofthebox!=0){
    // printf("not in the box!\t%d\n", iterationbox);
    iterationbox++;
    if (iterationbox>tolerancebox){
      if (sidechain>=2){
        aatom=6+side_chain_carbon*(sidechain-1);
        sidechain--;
        goto REWIND;
      }
      else if (sidechain==1){
        goto REWIND2;
      }
    }

    float ralpha=0.0;
    iterationangle=0;
    while ( (ralpha < M_PI*2.0/3.0) || (ralpha > M_PI*4.0/3.0) ){
      // printf("bad angle!\t%d\n", iterationangle);
      iterationangle++;
      if (iterationangle>toleranceangle){
        if (sidechain>=2){
          aatom=6+side_chain_carbon*(sidechain-1);
          sidechain--;
          goto REWIND;
        }
        else if (sidechain==1){
          goto REWIND2;
        }
      }

      overlap=1;
      iterationvdw=0;
      while (overlap!=0){
        // printf("overlap!\t%d\n", iterationvdw);
      iterationvdw++;
      if (iterationvdw>tolerancevdw){
        if (sidechain>=2){
          aatom=6+side_chain_carbon*(sidechain-1);
          sidechain--;
          goto REWIND;
        }
        else if (sidechain==1){
          goto REWIND2;
        }
      }
      place_atom(atommax+sidechain*21+l, atom1, molecule, rho);
      is_it_in_the_box(atommax+sidechain*21+l, molecule);
      ralpha = scalar_product(atommax+sidechain*21+l, atom1, atom2, molecule);
      overlap=0;
      count_overlap(molecule, atommax+sidechain*21+l, atom, 6, 2.6);
      count_overlap(molecule, atommax+sidechain*21+l, atommax+sidechain*21+l-6, atommax, 2.6);
    }
  }
}
if (l==3){ // place F atoms on C atom of the side chain
  float ralphaccf1=0.0; // C(atom-6)-C(atom-3)-F(atom-2) angle
  float ralphaccf2=0.0; // C(atom)-C(atom-3)-F(atom-2) angle
  while ( (ralphaccf1 < 0.5*M_PI) || (ralphaccf1 > 1.5*M_PI) ||
          (ralphaccf2 < 0.5*M_PI) || (ralphaccf2 > 1.5*M_PI) ){
    place_atom(atommax+sidechain*21+l-2, atommax+sidechain*21+l-3, molecule, rcf);
    ralphaccf1 = scalar_product(atommax+sidechain*21+l-2, atommax+sidechain*21+l-3, atom-1, molecule);
    ralphaccf2 = scalar_product(atommax+sidechain*21+l-2, atommax+sidechain*21+l-3,
                                atommax+sidechain*21+l, molecule);
    vector_product(atom-1, atommax+sidechain*21+l-3, atommax+sidechain*21+l-2,
            atommax+sidechain*21+l-1, atommax+sidechain*21+l, molecule, ralphaccf1, rcf);
  }
}
else if (l==6){ // place F atoms on C atom of the side chain
  float ralphaccf1=0.0; // C(atom-6)-C(atom-3)-F(atom-2) angle
  float ralphaccf2=0.0; // C(atom)-C(atom-3)-F(atom-2) angle
  float ralphaccf3=0.0;
  float ralphafcf=0.0;
  while ( (ralphaccf1 < 0.5*M_PI) || (ralphaccf1 > 1.5*M_PI) ||
          (ralphaccf2 < 0.5*M_PI) || (ralphaccf2 > 1.5*M_PI) ||
          (ralphaccf3 < 0.5*M_PI) || (ralphaccf3 > 1.5*M_PI) ){
    place_atom(atommax+sidechain*21+l-2, atommax+sidechain*21+l-3, molecule, rcf);
    ralphaccf1 = scalar_product(atommax+sidechain*21+l-2, atommax+sidechain*21+l-3,
         atommax+sidechain*21+l-6, molecule);
    ralphaccf2 = scalar_product(atommax+sidechain*21+l-2, atommax+sidechain*21+l-3,
         atommax+sidechain*21+l, molecule);
    vector_product(atommax+sidechain*21+l-6, atommax+sidechain*21+l-3, atommax+sidechain*21+l-2,
            atommax+sidechain*21+l-1, atommax+sidechain*21+l, molecule, ralphaccf1, rcc);
    place_atom(atommax+sidechain*21+l+1, atommax+sidechain*21+l-1, molecule, rcf);
    ralphaccf3 = scalar_product(atommax+sidechain*21+l+1, atommax+sidechain*21+l-1,
         atommax+sidechain*21+l-3, molecule);
    vector_product(atommax+sidechain*21+l-3, atommax+sidechain*21+l-1, atommax+sidechain*21+l+1,
            atommax+sidechain*21+l+2, atommax+sidechain*21+l-3, molecule, ralphaccf3, rcf);
    ralphafcf=scalar_product(atommax+sidechain*21+l+1, atommax+sidechain*21+l-1,
              atommax+sidechain*21+l+2, molecule);
    vector_product(atommax+sidechain*21+l+1, atommax+sidechain*21+l-1, atommax+sidechain*21+l+2,
            atommax+sidechain*21+19, atommax+sidechain*21+l-3, molecule, ralphafcf, rcf);
  }
}
else if (l==12){
  float ralphaccf1=0.0; // C(atom-6)-C(atom-3)-F(atom-2) angle
  float ralphaccf2=0.0; // C(atom)-C(atom-3)-F(atom-2) angle
  while ( (ralphaccf1 < 0.5*M_PI) || (ralphaccf1 > 1.5*M_PI) ||
          (ralphaccf2 < 0.5*M_PI) || (ralphaccf2 > 1.5*M_PI) ){
    place_atom(atommax+sidechain*21+l-2, atommax+sidechain*21+l-3, molecule, rcf);
    ralphaccf1 = scalar_product(atommax+sidechain*21+l-2, atommax+sidechain*21+l-3,
         atommax+sidechain*21+l-6, molecule);
```

```
              ralphaccf2 = scalar_product(atommax+sidechain*21+l-2, atommax+sidechain*21+l-3,
                   atommax+sidechain*21+l, molecule);
              vector_product(atommax+sidechain*21+l-6, atommax+sidechain*21+l-3, atommax+sidechain*21+l-2,
                      atommax+sidechain*21+l-1, atommax+sidechain*21+l, molecule, ralphaccf1, rcf);
            }
          }
          else if (l==15){
            float ralphaccf1=0.0; // C(atom-6)-C(atom-3)-F(atom-2) angle
            float ralphaccf2=0.0; // C(atom)-C(atom-3)-F(atom-2) angle
            while ( (ralphaccf1 < 0.5*M_PI) || (ralphaccf1 > 1.5*M_PI) ||
                    (ralphaccf2 < 0.5*M_PI) || (ralphaccf2 > 1.5*M_PI) ){
              place_atom(atommax+sidechain*21+l-2, atommax+sidechain*21+l-3, molecule, rcf);
              ralphaccf1 = scalar_product(atommax+sidechain*21+l-2, atommax+sidechain*21+l-3,
                   atommax+sidechain*21+l-6, molecule);
              ralphaccf2 = scalar_product(atommax+sidechain*21+l-2, atommax+sidechain*21+l-3,
                   atommax+sidechain*21+l, molecule);
              vector_product(atommax+sidechain*21+l-6, atommax+sidechain*21+l-3, atommax+sidechain*21+l-2,
                      atommax+sidechain*21+l-1, atommax+sidechain*21+l, molecule, ralphaccf1, rcf);
            }
            float ralphacso=0.0; // C S Os angle
            while ( (ralphacso < 0.5*M_PI) || (ralphacso > 1.5*M_PI) ){
              place_atom(atommax+(sidechain*21)+l+1, atommax+(sidechain*21)+l, molecule, ros);
              ralphacso = scalar_product(atommax+(sidechain*21)+l+1, atommax+(sidechain*21)+l,
                  atommax+(sidechain*21)+l-3, molecule);
              vector_product(atommax+(sidechain*21)+l-3, atommax+(sidechain*21)+l, atommax+(sidechain*21)+l+1,
                      atommax+(sidechain*21)+l+2, atommax+(sidechain*21)+l+3, molecule, ralphacso, ros);
            }
            float ralphacso1=0.0;
            while ( (ralphacso1 < 0.5*M_PI) || (ralphacso1 > 1.5*M_PI) ){
              place_atom(atommax+(sidechain*21)+l+3, atommax+(sidechain*21)+l, molecule, ros);
              ralphacso1 = scalar_product(atommax+(sidechain*21)+l+3, atommax+(sidechain*21)+l,
                   atommax+(sidechain*21)+l-3, molecule);
            }
          }
        }
        sidechain++; // side chain placed!
      }
      else{ // continue backbone: place 2nd F on C backbone atom using vector product
        vector_product(atom-6, atom-3, atom-2, atom-1, atom, molecule, ralphaccf1, rcf);
      }
    }
    end_to_end_distance(molecule); // compute end-to-end distance of the finished chain
  }

  print_xyz_file();
  print_gro_file();
  print_end_to_end();

  return (0);
}
```

code for AB_2-like chains (like Teflon, $CF_2 - CF_2$)

```
////////////////////////////////////////////////////////////////////////////////////////////////////
//                                                                                                //
//   This code generates a configuration of an AB2-like polymer, like Teflon (CF2-CF2).           //
//   The polymer chains have random starting points and each chain has a unique random path       //
//   (see the algorithm in the thesis manuscript for more details).                               //
//   The coordinates are saved in the specific formats configuration .xyz and .gro files          //
//   Executing x times the code will give x different (in the positions of the atoms) configurations.  //
//                                                                                                //
////////////////////////////////////////////////////////////////////////////////////////////////////

#include <cstdio>
#include <stdlib.h>
#include <math.h>
#include <time.h>
#include <iostream>

using namespace std;

float BOX=100.0; // edge of the cubic box (angstrom)
int moleculemax= 2; // number of AB2 chains
int atommax= 3000; // number of atoms per chain (multiple of 3)
float X[3][3000][2]; // coordinates
float end_to_end[2]; // end-to-end distances
float raa=1.5; // bond between two A atoms (angstrom)
float rab=1.0; // bond between A and B
int outofthebox; // 0 if atom in the box, 1 if atom outside the box
int overlap; // 0 if no overlap, 1 if any overlap

// give random coordinates to first atom of a chain
int randomize_first_atom (int atom, int molecule){
  for (int coordinate=0 ; coordinate <3 ; coordinate++){
    X[coordinate][atom][molecule]= ( rand() / (float)RAND_MAX ) * (BOX-0.0) + 0.0;
  }
  return(0);
}

// place atom1 randomly at a distance rho from atom2
int place_atom (int atom1, int atom2, int molecule, float rho){
  float phi= ( rand() / (float)RAND_MAX ) * ( M_PI - 0.0 ) + 0.0;
  float teta= ( rand() / (float)RAND_MAX ) * ( 2.*M_PI - 0.0 ) + 0.0;
  X[0][atom1][molecule]= X[0][atom2][molecule] + rho*sin(phi)*cos(teta);
  X[1][atom1][molecule]= X[1][atom2][molecule] + rho*sin(phi)*sin(teta);
  X[2][atom1][molecule]= X[2][atom2][molecule] + rho*cos(phi);
  return(0);
}

// check whether atom is inside or outside the box
int is_it_in_the_box(int atom, int molecule){
  outofthebox=0;
  if ( (X[0][atom][molecule]<0.0) || (X[0][atom][molecule]>BOX) ||
       (X[1][atom][molecule]<0.0) || (X[1][atom][molecule]>BOX) ||
       (X[2][atom][molecule]<0.0) || (X[2][atom][molecule]>BOX) ){
    outofthebox=1;
  }
  return(0);
}

// compute angle using scalar product
float scalar_product(int atom1, int atom2, int atom3, int molecule){
  float vec1x= X[0][atom1][molecule]-X[0][atom2][molecule];
  float vec1y= X[1][atom1][molecule]-X[1][atom2][molecule];
  float vec1z= X[2][atom1][molecule]-X[2][atom2][molecule];
  float vec2x= X[0][atom3][molecule]-X[0][atom2][molecule];
  float vec2y= X[1][atom3][molecule]-X[1][atom2][molecule];
  float vec2z= X[2][atom3][molecule]-X[2][atom2][molecule];
  float dist1=sqrt(vec1x*vec1x + vec1y*vec1y + vec1z*vec1z);
  float dist2=sqrt(vec2x*vec2x + vec2y*vec2y + vec2z*vec2z);
  float ralpha = acos( (vec1x*vec2x + vec1y*vec2y + vec1z*vec2z) / (dist1*dist2) );
  return (ralpha);
}

// place atom using vector product
float vector_product(int atom1, int atom2, int atom3, int atom4, int atom5, int molecule, float ralphaccf1, float rho){
  float vec1x= X[0][atom1][molecule]-X[0][atom2][molecule];
  float vec1y= X[1][atom1][molecule]-X[1][atom2][molecule];
  float vec1z= X[2][atom1][molecule]-X[2][atom2][molecule];
  float dist1=sqrt(vec1x*vec1x + vec1y*vec1y + vec1z*vec1z);
  float vec2x= X[0][atom3][molecule]-X[0][atom2][molecule];
  float vec2y= X[1][atom3][molecule]-X[1][atom2][molecule];
  float vec2z= X[2][atom3][molecule]-X[2][atom2][molecule];
  float dist2=sqrt(vec2x*vec2x + vec2y*vec2y + vec2z*vec2z);
  float vec3x= X[0][atom5][molecule]-X[0][atom2][molecule];
  float vec3y= X[1][atom5][molecule]-X[1][atom2][molecule];
  float vec3z= X[2][atom5][molecule]-X[2][atom2][molecule];
  X[0][atom4][molecule]= (vec1y*vec2z - vec1z*vec2y)/(dist1*dist2*sin(ralphaccf1))*rho  + X[0][atom2][molecule] ;
  X[1][atom4][molecule]= (-vec1x*vec2z + vec1z*vec2x)/(dist1*dist2*sin(ralphaccf1))*rho + X[1][atom2][molecule] ;
  X[2][atom4][molecule]= (vec1x*vec2y - vec1y*vec2x)/(dist1*dist2*sin(ralphaccf1))*rho  + X[2][atom2][molecule] ;
  // take minus the cross product (i.e. opposite direction) if (atom) and (atom-1) are too close
  float vec4x= X[0][atom4][molecule]-X[0][atom2][molecule];
  float vec4y= X[1][atom4][molecule]-X[1][atom2][molecule];
  float vec4z= X[2][atom4][molecule]-X[2][atom2][molecule];
  float dist3=sqrt(vec3x*vec3x + vec3y*vec3y + vec3z*vec3z);
  float dist4=sqrt(vec4x*vec4x + vec4y*vec4y + vec1z*vec4z);
  float ralphaccf3 = acos( (vec3x*vec4x + vec3y*vec4y + vec3z*vec4z) / (dist3*dist4) );
  if ( (ralphaccf3 < M_PI*0.5) || (ralphaccf3 > M_PI*1.5) ){
    X[0][atom4][molecule]= (-vec1y*vec2z + vec1z*vec2y)/(dist1*dist2*sin(ralphaccf1))*rho + X[0][atom2][molecule];
    X[1][atom4][molecule]= (vec1x*vec2z - vec1z*vec2x)/(dist1*dist2*sin(ralphaccf1))*rho + X[1][atom2][molecule];
    X[2][atom4][molecule]= (-vec1x*vec2y + vec1y*vec2x)/(dist1*dist2*sin(ralphaccf1))*rho + X[2][atom2][molecule];
  }
  return (0);
}

// check whether there is any overlap
int count_overlap(int molecule, int atom1, int atom2, int atom3, float rvdw){
  float rvdw2=rvdw*rvdw;
  for (int j=molecule; j>-1; j--){
    for (int i=atom2; i>=atom3; i=i-3){
      float rx=X[0][atom1][molecule]-X[0][i][j];
      float ry=X[1][atom1][molecule]-X[1][i][j];
      float rz=X[2][atom1][molecule]-X[2][i][j];
      float r2 = rx*rx + ry*ry + rz*rz;
      if (r2<rvdw2){
        overlap++;
      }
    }
  }
  return(0);
}

// compute end-to-end distance for each molecule
int end_to_end_distance(int molecule){
  float rx=X[0][atommax][molecule]-X[0][6][molecule];
  float ry=X[1][atommax][molecule]-X[1][6][molecule];
  float rz=X[2][atommax][molecule]-X[2][6][molecule];
  end_to_end[molecule]=sqrt(rx*rx + ry*ry + rz*rz);
  return(0);
```

```
}

// print end-to-end distances in a file
int print_end_to_end(){
  FILE * pFile3;
  pFile3 = fopen ("end_to_end.dat","w");
  fprintf (pFile3, "#molecule  end-to-end distance\n");
  for (int i=0; i<moleculemax; i++){
    fprintf (pFile3, "%8d %8.3f\n", i, end_to_end[i]);
  }
  fclose (pFile3);
  return(0);
}

// print coordinates in a .xyz file
int print_xyz_file(){
  FILE * pFile;
  pFile = fopen ("geometry.xyz","w");
  fprintf(pFile, "%d\n", moleculemax*(atommax-9));
  fprintf(pFile, "%d\n", moleculemax*(atommax-9));
  for (int molecule=0; molecule<moleculemax; molecule++){
    for (int atom=9 ; atom<atommax ; atom+=3){
      fprintf (pFile, "A %8.3f%8.3f%8.3f\n", X[0][atom-3][molecule], X[1][atom-3][molecule], X[2][atom-3][molecule]);
      fprintf (pFile, "B %8.3f%8.3f%8.3f\n", X[0][atom-2][molecule], X[1][atom-2][molecule], X[2][atom-2][molecule]);
fprintf (pFile, "B %8.3f%8.3f%8.3f\n", X[0][atom-1][molecule], X[1][atom-1][molecule],X[2][atom-1][molecule]);
    }
  }
  fclose (pFile);
  return(0);
}

// print coordinates in a .gro file
int print_gro_file(){
  FILE * pFile2;
  pFile2 = fopen ("geometry.gro","w");
  fprintf(pFile2, "%d\n", moleculemax*(atommax-9));
  fprintf(pFile2, "%d\n", moleculemax*(atommax-9));
  int residueindex=1; // molecule index
  char residue_name[] = "res"; // residue name
  char atomtype_A[]= "A";
  char atomtype_B[]= "B";

  for (int molecule=0; molecule<moleculemax; molecule++){
    int Aindex=1;
    int Bindex=1;
    int atomindex=1;
    for (int atom=9 ; atom<atommax ; atom+=3){
      fprintf(pFile2,"%5d%4s%4s%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name, atomtype_A, atomindex, X[0][atom-3][molecule
      Aindex++;
      fprintf(pFile2,"%5d%4s%4s%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name, atomtype_B, atomindex, X[0][atom-2][molecule
      Bindex++;
      fprintf(pFile2,"%5d%4s%4s%5d%8.3f%8.3f%8.3f\n", residueindex, residue_name, atomtype_B, atomindex, X[0][atom-1][molecule
      Bindex++;
    }
    residueindex++;
  }
  fprintf(pFile2, "%8.3f%8.3f%8.3f\n", BOX*0.1, BOX*0.1, BOX*0.1);
  fclose(pFile2);
  return(0);
}

int main(){
  int atom, aatom; // current atom index
  int molecule; // current molecule index
int iterationvdw, iterationangle, iterationbox; // tries to fullfill conditions: vdw (overlaps), angle, in the box
int tolerancebox=20; // number of allowed tries to put the atom in the box before to go rewind in the chain building
int tolerancevdw=20; // number of allowed tries to put the atom without overlaps with the other atoms
int toleranceangle=20; // number of permitted tries for an angle

// set all coordinates to 0
for (molecule=0; molecule<moleculemax; molecule++){
  for (atom=0; atom<atommax; atom++ ){
    for (int coordinate=0; coordinate<3; coordinate++){
      X[coordinate][atom][molecule]=0.0;
    }
  }
}

srand (time (NULL)) ; // random seed taken from system date

for (molecule=0; molecule<moleculemax; molecule++){
  cout << "molecule: " << molecule << endl;
  randomize_first_atom(0, molecule); // place randomly inside the box the 1st A backbone atom of the chain

  //place 2nd A backbone atom inside the box and at a distance raa of 1st A
  outofthebox=1; // atom is out of the box
  while (outofthebox!=0){
    float rho=raa;
    place_atom(3, 0, molecule, rho);
    is_it_in_the_box(0, molecule); // check whether atom is inside or outside the box
  }

   aatom=6;
   // generate the A backbone chain
   for (atom=aatom; atom<atommax; atom+=3){
     REWIND:
     outofthebox=1; // atom is out of the box
     iterationbox=0;
     while ( outofthebox!=0  ){
       iterationbox++;
       //cout << "not in the box! " << iterationbox << " " << atom << " " << molecule << endl;
       if (iterationbox>tolerancebox){ // number of tries > number of tries allowed
        atom=atom-6;
        goto REWIND; // go one A atom rewind in the chain building process
       }

       float ralphaccc=0.0; // A A A angle
       iterationangle=0;
       while ( (ralphaccc< (0.5*M_PI)) || (ralphaccc> (1.5*M_PI)) ){ // A A A angle must be in the range [pi/2; 3pi/2]
        iterationangle++;
        //cout << "bad angle! " << iterationangle << " " <<  atom << " " << molecule << endl;
        if (iterationangle>toleranceangle){ // number of tries > number of tries allowed
        atom=atom-6;
          goto REWIND;
        }

        overlap=1; // atom overlaps
        iterationvdw=0;
        while(overlap!=0){
          iterationvdw++;
          //cout << "overlap! " << iterationvdw << " " << atom << " " << molecule << endl;
          if (iterationvdw>tolerancevdw){ // number of tries > number of tries allowed
            atom=atom-6;
            goto REWIND;
          }
          place_atom(atom, atom-3, molecule, raa);
          is_it_in_the_box(atom, molecule);
          ralphaccc = scalar_product(atom, atom-3, atom-6, molecule); // compute ralphaccc angle
```

```
            overlap=0; // atom does not overlap
            if (atom>=9){ // check for overlap(s)
              count_overlap(molecule, atom, atom-9, 6, 4.1);
            }
          }
         }
       }
       // A backbone placed!

       // add 1st of the two B on each A atom of the backbone
       float ralphaccf1=0.0; // A(atom-6)-A(atom-3)-B(atom-2) angle
       float ralphaccf2=0.0; // A(atom)-A(atom-3)-B(atom-2) angle
       while ( (ralphaccf1 < 0.5*M_PI) || (ralphaccf1 > 1.5*M_PI) ||
               (ralphaccf2 < 0.5*M_PI) || (ralphaccf2 > 1.5*M_PI) ){// ralphaccf1 and ralphaccf2 must be in the range [pi/2; 3
               place_atom(atom-2, atom-3, molecule, rab); // place B atom at a distance rab from A backbone atom
               ralphaccf1 = scalar_product(atom-2, atom-3, atom-6, molecule); // compute ralphaccf1 angle
               ralphaccf2 = scalar_product(atom-2, atom-3, atom, molecule); // compute ralphaccf2 angle
              }

       // place 2nd B on A backbone atom using vector product
       vector_product(atom-6, atom-3, atom-2, atom-1, atom, molecule, ralphaccf1, rab);

       // compute end-to-end distance of the finished chain
       end_to_end_distance(molecule);
    }
  }

  print_xyz_file();
  print_gro_file();
  print_end_to_end();

  return (0);
}
```

5.C Appendix: Densities

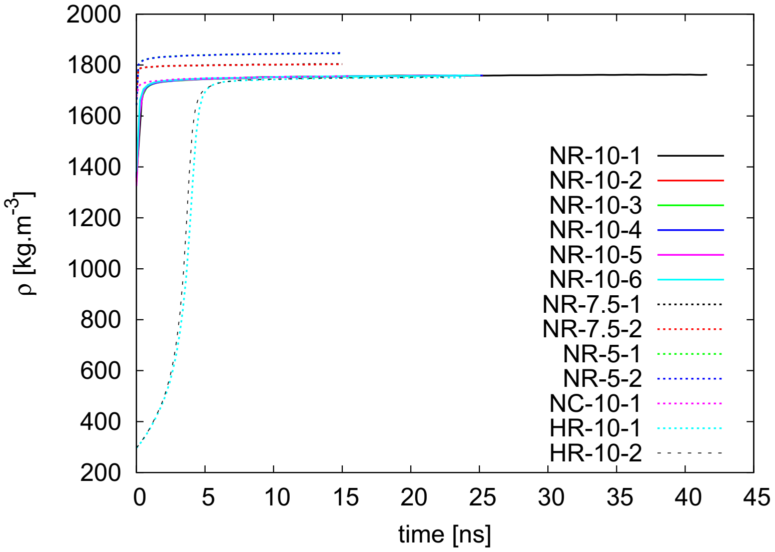


Figure 19: Total density vs. the simulation time, for all systems. The curves have been smoothed with a Bezier function.

6 Results and Discussions

6.1 Structure

6.1.1 VMD Snapshots

SO_3^--K^+**-ion pairs:**
The K^+-ions and SO_3^--groups, they are mostly located in the aqueous domains (see section 6.1.5), are very often associated into K^+-SO_3^--ion pairs. This is seen in fig. 20 for the system NR-10-1 at t=40 ns, where the K^+-ions 4.5 Å closer of at least one S-atom are represented in red and the ones that are not 4.5 Å closer of a S-atom in blue. This can also be seen in Appendix 6.A (figs. 76, 77, 78, 79, 80, and 81), for various systems and times. The higher the water content λ, the higher the ratio of dissociated ions over associated ion pairs, as expected (compare figs. 20, 78, and 79). Since there are no electric charges on the polymer backbone, the dielectric constant ϵ (if such a macroscopic concept applies here) is controlled by that of the water. The dielectric constant ϵ is expected to grow with the size of the domains; for water the full bulk value ($\epsilon \approx 81$) is not reached unless a domain has dimensions of at least 10 Å [172,173].

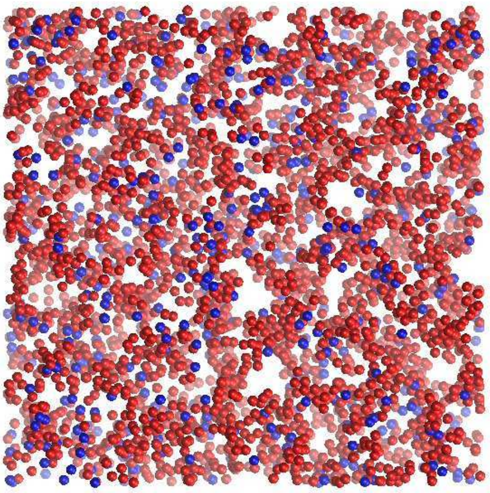


Figure 20: NR-10-1 system at t=40 ns. Only the K^+-ions are shown. In red: closer than 4.5 Å from at least one S-atom; in blue: more than 4.5 Å from any S-atom. Equivalent representations for other systems at various times are given in Appendix 6.A.

6.1.2 Radial Distribution Functions and Coordination Numbers

The radial distribution function (rdf) $g(r)$ and its integral $n(r)$ have been mentioned in section 4.4. We compute them here for the S-S, S-Ow, and Ow-Ow, over the last 10 ns of the simulation of each run, as justified below for the statistically least favorable case, $g_{\mathrm{S-S}}/n_{\mathrm{S-S}}$.

Sulfur-Sulfur (S-S)
The mean sulfonate-sulfonate distance strongly influences how rapidly protons can hop, according e.g. to the Grotthuss mechanism, between sulfonate groups [2, 20]. Despite charge repulsion, the sulfonate groups aggregate because of their attraction to protons and water. Protons need, on one hand, a short distance from one $\mathrm{SO_3^-}$-group to the next to diffuse [93, 174]. On the other hand, a too short sulfonate-sulfonate distance acts as a proton-trap [21, 175–177].

The time-evolution of the average number of S-atoms around each S-atom, i.e. $n_{\mathrm{S-S}}(r,t)$ during equilibration and simulation is shown in fig. 21. In our computation, we counted S-atoms that are closer to one another than 0.8 nm, i.e. the position of the first minimum of the $g_{\mathrm{S-S}}(r)$ (see fig. 22). Since there is no significant difference between the six NR-10 systems, we show here only the curve for NR-10-1. For the same reason, we show only the curve for the first of the NR-7.5, NR-5, and HR-10 systems. The curves for the systems not shown here (i.e., NR-10-2, NR-10-3, NR-10-4, NR-10-5, NR-10-6, NR-7.5-2, NR-5-2, HR-10-2) are given in Appendix 6.B (fig. 87).

It is seen in fig. 21 that $n_{\mathrm{S-S}}(0.8\text{ nm}, t\text{ large})$ is not yet quite stable: some curves are still evolving even after several 10 ns, see e.g. the curve for NR-10-1. We have attempted to estimate the limiting values by fitting a function of the form $a \cdot exp(-b \cdot t) + c$ for different time windows (from the very beginning and starting at $t = 4.0$ ns). These fits (red and black curves in the figure) show that at larger t, the curves seem to converge to similar values, which can be used as an estimate of the real limiting value of n and its error. The n-values obtained for the last 10 ns are within these error bars and can thus be taken as representative of the equilibrium.

Fig. 22 shows the rdf $g_{\mathrm{S-S}}(r)$ computed over the last 10 ns of each run. Again the curves for the systems not shown here are given in Appendix 6.B (fig. 88). All systems except NR-5-1 and NR-5-2 show two characteristic maxima around, roughly, 0.6 and 0.9 nm. The fact that there is no well defined second maximum for the systems hydrated at λ=5 means that at low water content, the distribution of the sulfonate groups is more dispersed. Our curves are in close agreement with e.g. the ones in [2, 90]. The position of the first peak shifts to larger distances when increasing λ, as expected [91]: 0,61, 0.62, and 0.63 nm for λ=5, 7.5, and 10, respectively. This is due to the swelling of the membrane: the size of the aqueous domains increases when the water content is increased [1] and so does the mean S-S distance. Accordingly, the number of neighbors is decreased when λ is increased (compare NR-10-1, NR-7.5.1, and NR-5-1 in fig. 21). This is in keeping with an increasing solvation of the S-atoms when increasing λ (see the S-Ow rdfs below).

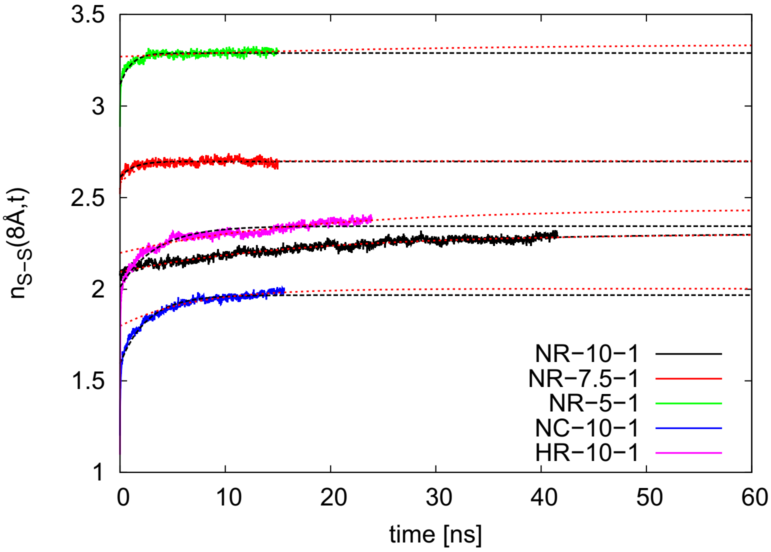


Figure 21: Average number of S-atoms closer than 0.8 nm to an S-atom vs. the simulation time, and fitted functions (red and black, see text), for various systems. The curves for the systems not shown here are given in Appendix 6.B.

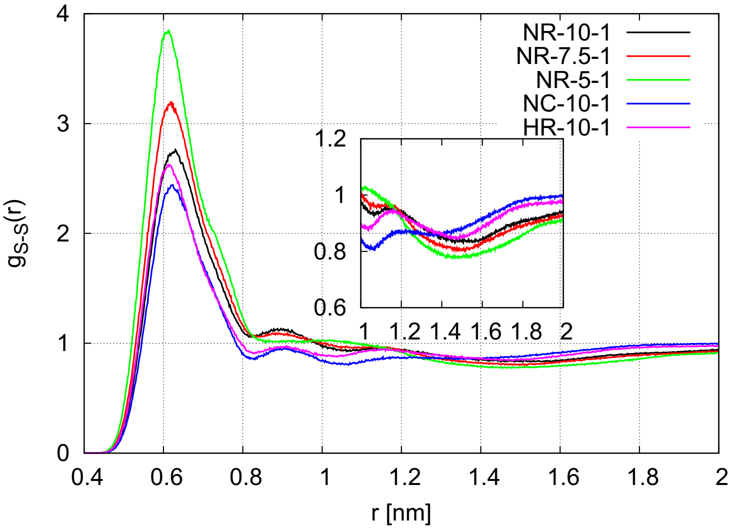


Figure 22: $g_{\mathrm{S-S}}(r)$ over the last 10 ns, see text, for various systems. The curves for the systems not shown here are given in Appendix 6.B.

The $g_{\mathrm{S-S}}(r)$-curves for NR-10-1 and HR-10-1 are very similar. Nonetheless, shortening the side chain length (Nafion/Hyflon) leads to shorter mean S-S distances. This is seen in fig. 22 when comparing the positions of the first maxima. Accordingly, it is seen in fig. 21 that, on average, an S-atom has more S-neighbors in Hyflon than in Nafion: roughly 2.4 vs. roughly 2.2. One may thus expect that the sulfonate groups have less Ow-neighbors in Hyflon than in Nafion (see the S-Ow rdfs below). In the simulation study by Brandell et al. [58], 3.1 and 2.6 neighbors were found for Hyflon and Nafion, respectively, with the same distance criterion (8 Å). These authors therefore suggested that Hyflon promotes the formation of larger S-clusters than Nafion.

Furthermore, the number of clusters and the histogram of number of atoms per cluster (we call this the cluster size distribution) were computed using the standard cluster analysis (the method is described in section 4.4, more results are found in 6.1.5). This is shown in figs. 23 and 24 for the systems NR-10-1, NR-10-2, HR-10-1, and HR-10-2. In these computations, we considered only S-atoms and we used a distance criterion d=0.8 nm. With similar arguments as above, the number of S-clusters (fig. 23), (even though still decreasing toward the end of the simulation) can be considered to be larger in Nafion than in Hyflon: roughly 900 vs. roughly 800. The clusters must thus be, on average, larger in Hyflon than in Nafion. This is the case, as shown in fig. 24: the number of clusters with, say, less than 30 atoms, is larger in Nafion than in Hyflon, and the number of clusters with, say, more than 30 atoms, is larger in Hyflon.

One can thus conclude, like Brandell et al. [58], that Hyflon promotes the formation of larger S-clusters than Nafion. This might have an effect on the diffusion of the water molecules and the cations. In their simulation study [58], Brandell et al. found that the self-diffusion coefficients of the water molecules and the hydronium ions are slightly higher in Hyflon than in Nafion. The self-diffusion coefficients of the water molecules and the K^+-ions in our systems will be discussed later in this chapter.

The effect of changing the morphology (compare NR-10-1 and NC-10-1 in fig. 22) on the position of the first peak is not significant. However, fig. 21 shows that the coordination number is lower for NC-10-1 than for NR-10-1: roughly 2.0 vs. roughly 2.3. Accordingly, the number of clusters is higher in NC-10-1 than in NR-10-1 (see fig. 23), and the clusters are smaller (see fig. 24). This means that the spatial distribution of the S-atoms remains less homogeneous in the cylindrical system than in the random ones, at equivalent water content. In this sense, the initial setup is preserved (see section 5.3). Hence a lower fraction of the S-atoms must be in the aqueous domains.

Also, the zoom at large distances on the $g_{\mathrm{S-S}}(r)$ (see insert in fig. 22) shows a dip around, say, 1.5 nm. This dip results from the lower density of (and, hence, the lower probability of finding) S-atoms between S-clusters. Above, say, 2 nm, the distribution of the S-atoms is random.

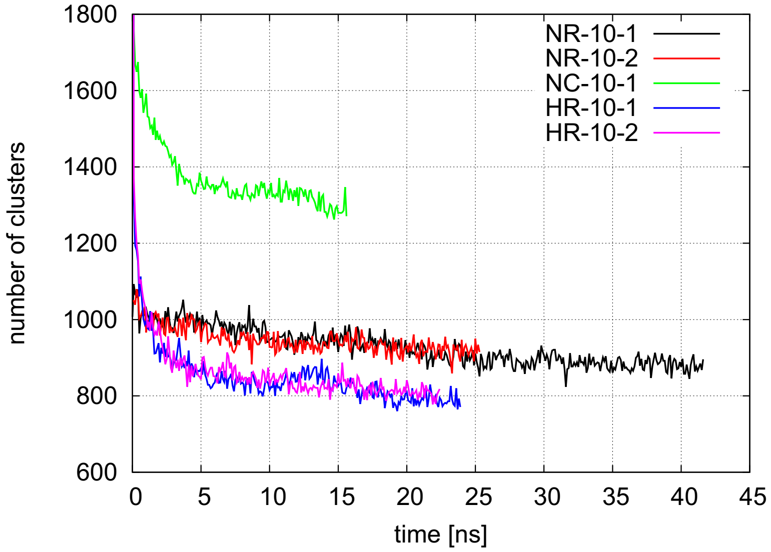


Figure 23: Number of S-clusters vs. the simulation time, for various systems. The distance criterion is 8.0 Å.

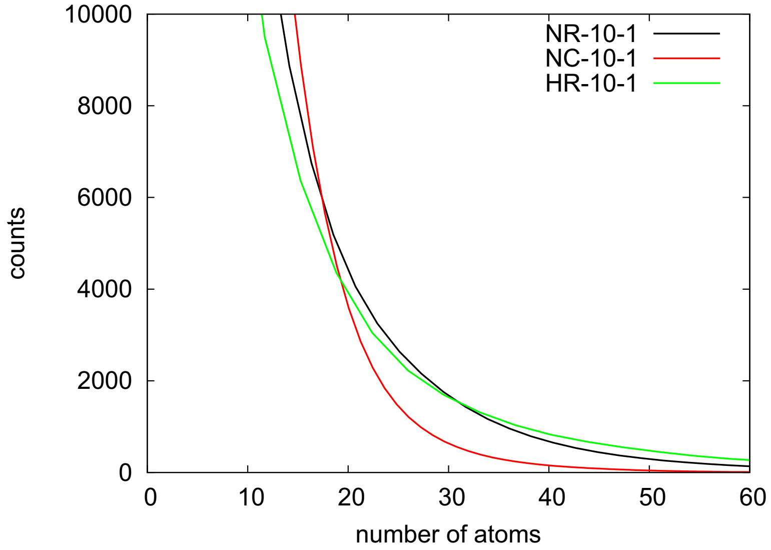


Figure 24: Distribution of the number of S-atoms per cluster, averaged over the last 10 ns (computed every 100 ps, hence averaged over 100 configurations), for various systems. The curves have been smoothed with a Bezier function.

Sulfur–Water-Oxygen (S-Ow)

Fig. 25 shows the rdfs $g_{\mathrm{S-Ow}}(r)$ (left axis, full lines) and their integrals $n_{\mathrm{S-Ow}}(r)$ (right axis, dashed lines) over the last 10 ns, for typical systems. We show here only the curves for the typical systems (all the curves are given in Appendix 6.B, fig. 89). It is seen that $g_{\mathrm{S-Ow}}(r) > 1$ up to about 1 nm, for all systems. This means that the local density of water around the sulfonate groups is larger than the average up to about 1 nm, and lower than the average above 1 nm. In other words the water density around the sulfonates is inhomogeneous at the nm scale, in keeping with the existence of water aggregates with typical sizes of this order (see also the Ow-Ow rdfs below).

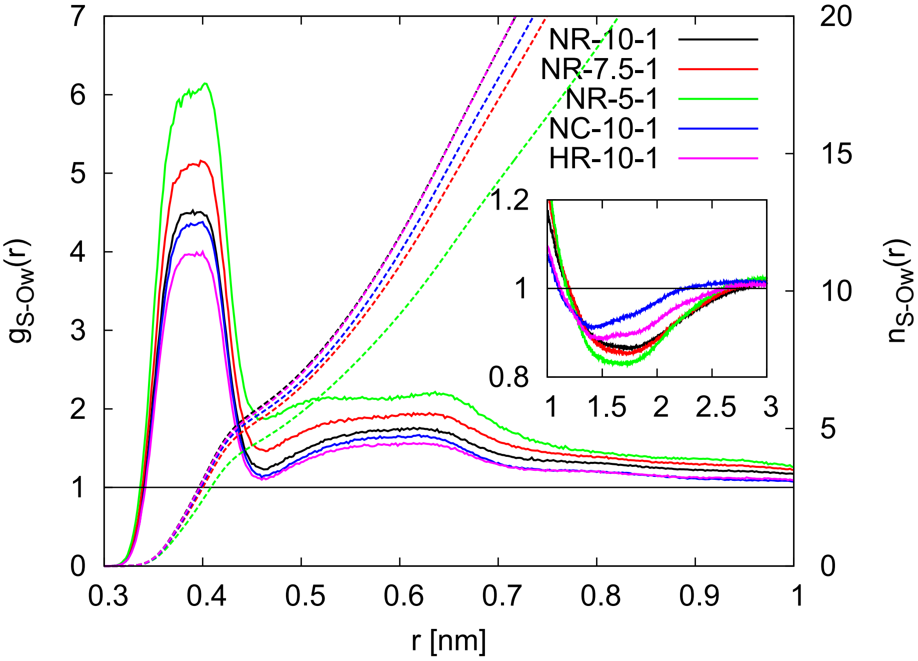


Figure 25: $g_{\mathrm{S-Ow}}(r)$ (left axis) and $n_{\mathrm{S-Ow}}(r)$ (right axis) over the last 10 ns of the simulation (computed every 100 ps, hence averaged over 100 configurations), for various systems.

The level of solvation of a given species (call it solute) can be judged either with respect to the average density of the solvent (i.e. from the rdf $g(r)$) or from the absolute number of solvent molecules in the vicinity of the solute (i.e. from $n(r)$). Here we have chosen the second definition. The $n_{\mathrm{S-Ow}}(r)$-curves in fig. 25 and 89 shows that the average number of water molecules around the sulfonates decreases when decreasing λ, as expected. In particular, the S-atoms in the systems hydrated at $\lambda > 5$ are better solvated than those at $\lambda = 5$: $n_{\mathrm{S-Ow}}(0.46 \text{ nm}) \approx 4.5$ and 5.5 for $\lambda = 5$ and $\lambda > 5$, respectively. This is in accordance with the decreasing $n_{\mathrm{S-S}}(0.8 \text{ nm})$ when increasing λ (see above). For comparison: in their simulation study, Dupuis et al. [91] found $n_{\mathrm{S-Ow}}(0.46 \text{ nm}) = 3.66$ and 5.62, for $\lambda = 6$ and 11, respectively.

We note, however, that the (relative) inhomogeneity in the water density around the sulfonate groups increases when decreasing λ (compare the heights of the $g_{\mathrm{S-Ow}}(r)$-curves in fig. 25 for NR-10-1, NR-7.5-1, and NR-5-1). No significant difference on the peak positions of the rdfs in fig. 25 occurs between the systems: the first $g_{\mathrm{S-Ow}}(r)$-minimum is at ≈ 4.6 Å. It reveals the first solvation shell of water and ions surrounding each sulfonate. The peak locations and shapes resemble those of previous studies [2, 27, 91]. Also, all systems exhibit a broad dip around 1.5 nm (see the insert in fig. 25), like for the $g_{\mathrm{S-S}}(r)$-curves (see above) and the $g_{\mathrm{Ow-Ow}}(r)$-curves (see below).

Changing the morphology of Nafion (NR-10-1 (random) vs. NC-10-1 (cylindrical)) leads to a slightly lower $n_{\mathrm{S-Ow}}(r)$ in the cylindrical system (the curves for NR-10-1 and HR-10-1 overlap in fig. 25). This may still be due to the initial construction of the systems, see section 5.3. Also, the $n_{\mathrm{S-Ow}}$-curves for the Nafion (NR-10) and Hyflon (HR-10) systems seem superimposed (see figs. 25 and 89). No difference in the solvation number of the sulfonate groups in Hyflon and in Nafion can thus be detected, in contrast to the above expectations. Similar results have been obtained in [58].

Water-Oxygen–Water-Oxygen (Ow-Ow)
Fig. 26 shows the rdfs $g_{\mathrm{Ow-Ow}}(r)$ and their integrals $n_{\mathrm{Ow-Ow}}(r)$ over the last 10 ns, for typical systems (the curves for the systems not shown here are given in Appendix 6.B, fig. 90). An obvious function to compare with is the $g_{\mathrm{O-O}}(r)$ of liquid water at ambient conditions, also shown in fig. 26. We see striking differences. First: there are no well defined minima (hence no well defined hydration shells) for the $g_{\mathrm{Ow-Ow}}(r)$-curves at 0.33 nm and 0.58 nm, in contrast to bulk water. Furthermore, $g_{\mathrm{Ow-Ow}}(r) > 1$ up to large r-values for which $g_{\mathrm{O-O}}(r) \approx 1$. Second: there is a dip (see the insert in fig. 25) around about 2 nm (for all systems excepted the cylindrical one), and around 4-5 nm for the cylindrical (NC-10-1) one. In the simulation study by Voth and Knox [2], a similar dip was also observed around 3 nm. These dips are related to the 'ionomer peak' in the structure factors; this will be discussed below. The contribution to the ionomer peak of the S-S (dips in fig. 22) and S-Ow (dips in fig. 25) pairs will also be discussed.

These characteristic differences with bulk water appear for all Nafion and Hyflon systems investigated here. These have also been observed e.g. in [2, 91]. They clearly demonstrate the inhomogeneity at the nanometer scale. We note in passing that the position of the dip for the non-cylindrical systems is shifted to higher distances when increasing the water content (compare NR-10-1, NR-7.5-1, and NR-5-1 in the insert in fig. 25). This is due to the swelling behavior of the membranes: the average sizes of the water aggregates increase when increasing λ. We shall come back to this when discussing the aqueous clusters 6.1.5.

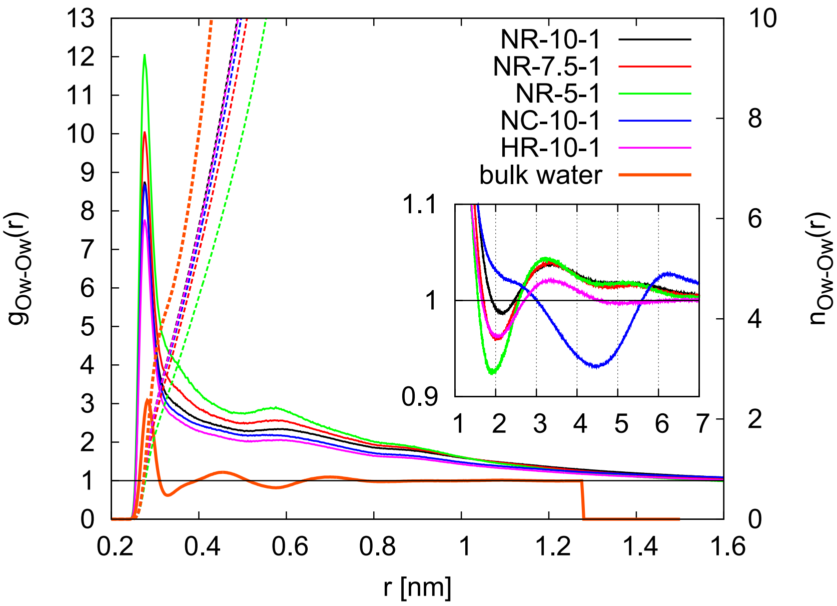


Figure 26: $g_{\mathrm{Ow-Ow}}(r)$ (left axis) and $n_{\mathrm{Ow-Ow}}(r)$ (right axis) over the last 10 ns of the simulation (computed every 100 ps, hence averaged over 100 configurations), for various systems.

The differences in $g_{\mathrm{Ow-Ow}}(r)$ between the various systems are nevertheless comparatively minor. In particular, there are no significant differences between the peak positions of the rdf-curves in fig. 26: for all systems, a sharp peak is seen at ≈ 2.8 Å, in keeping with e.g. [2, 58, 91]. This distance is in agreement with typical hydrogen bond distances in aqueous environments [178]. This first peak is followed by a broad second peak around ≈ 6 Å, in agreement with [2].

Due to the overall lower water density, $n_{\mathrm{Ow-Ow}}(r)$ in fig. 26 (like $n_{\mathrm{S-Ow}}(r)$, see above) is lower for the systems hydrated at $\lambda = 5$: At $\lambda > 5$ there are between 3.6 and 3.9 neighboring water molecules around each water molecule (defined at $r = 3.5$ Å) while for the systems at $\lambda = 5$, there are only about 3. For comparison, there are about 5 in bulk water, at this distance. The difference in $n_{\mathrm{S-Ow}}(r)$ between NR-10-1 and NR-7.5-1 is not significant compared to the difference between NR-10-1 (or NR-7.5-1) and NR-5-1. This is in keeping with the results by Dupuis et al. [91] who found, respectively, 2.56, 3.38, and 3.73 water molecules at $\lambda = 6$, 11, and 16, with the same distance criterion and with a λ-value of 5-6 for the percolation threshold, as suggested by these authors.

The difference in $n_{\mathrm{S-Ow}}(r)$ between NR-10-1 and NC-10-1 is negligible and the curves for NR-10-1 and HR-10-1 superimpose. There is thus, in this distance range, no effect on the solvation number of water of changing the morphology and the polymer side chain length.

6.1.3 Structure Factors

A way to compare the simulated structure with experimental data is to calculate one of the so-called total structure factor $S(q)$ (see section 4.4). It is an additional advantage of the simulation over the experiment that $S(q)$ can be decomposed as a sum of partial contributions $S_{\alpha\beta}(q)$ involving the atom types α and β. We compute here the $S(q)$s (neutron- and X-ray-weighted) and various $S_{\alpha\beta}(q)$s, for selected configurations of each run and also average over configurations.

Total Structure Factors $S(q)$
The characterization of the Nafion morphology via light scattering techniques has been reported in section 2. We recall here that, experimentally, a peak is usually observed at, say, $\approx 0.1 - 0.15$ Å^{-1} [1,5,63,65,179] (depending on the water content). It is called the ionomer peak and is interpreted as a consequence of a mean separation distance between aqueous clusters of $2\pi/q$ or, say, 4-6 nm. This peak shifts to lower q-values (longer distances) as the water content increases. Also, there is a broad, lower q ($\approx$ 0.03 Å^{-1}) region ("matrix knee") that has been assigned to semicrystalline polymer regions and a smooth "Porod region" at higher q ($\approx$ 0.3 Å^{-1}), where the intensity is expected to decay obeying a q^{-4} power law [1,5]. However, treating the "matrix knee" region via atomistic computer simulation would require much larger simulation boxes than the ones in this study, which is currently not achievable.

Figs. 27 and 28 show some $S(q)$s for neutron and X-ray scattering. The curves for the NR-5-2 and NR-7.5-2 systems are not shown here since there is no significant difference with NR-5-1 and NR-7.5-1 (all curves are given in Appendix 6.C, figs. 91 and 92). In our computation, we consider first for each system the last configuration only. We see no clear difference when averaging over the last 10 ns. This is shown in figs. 29 (neutron) and 30 (X-ray) for one system as an example. This means that the structure is sufficiently stable at least for the last 10 ns of the simulation, see also above in section 6.1.2. The time-evolution of the structure at earlier times will be discussed in section 6.2.

Only minor differences on the peak positions and relative intensities can be seen when comparing the curves for the neutron and X-ray scattering (compare figs. 27 and 28): both figures exhibit a maximum in the ionomer peak region for most of our systems, independently of the morphology, the side chain length, and the water content (see e.g. NR-10-1, HR-10-1, NC-10-1, NR-7.5-1). The systems for which there is no clear maximum in this region (see e.g. NR-10-5, NR-5-1) exhibit nonetheless a shoulder or a less well defined maximum. Furthermore, all the curves decay smoothly to zero at large q-values. They are thus likely to obey the experimentally observed power law behavior in the Porod region (the asymptotic behavior at large q-values will be examined in one example below). Our results are in keeping with the ones by Voth and Knox [2], i.e. that almost all models, even though they have very different structures, reproduce fairly well the ionomer peak and the power law behavior in the Porod regime. This means that neither the ionomer peak nor the power law behavior at high q is a strong indicator of the geometry. In other words, $S(q)$ calculations and scattering experiments in general are unable to elucidate the true structure of Nafion. Therefore more detailed measurements are needed.

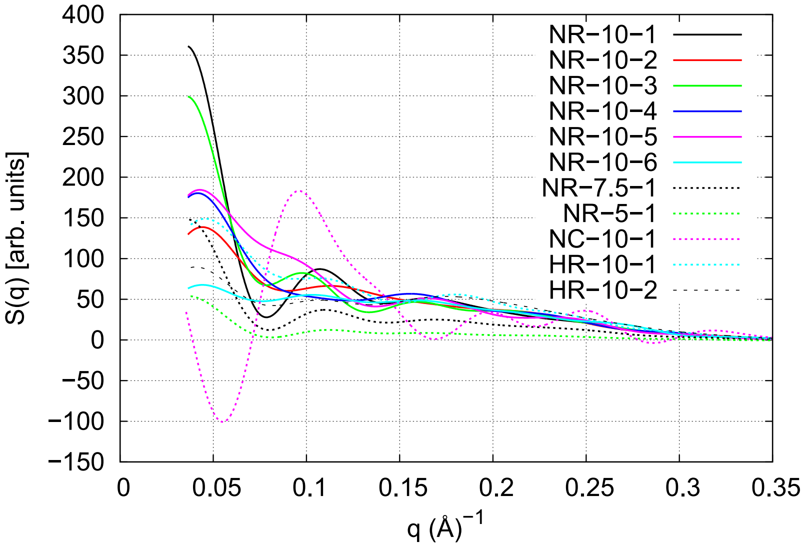


Figure 27: S(q) (neutron) of the last configuration of various systems. The curves not shown here are given in Appendix 6.C (fig. 91).

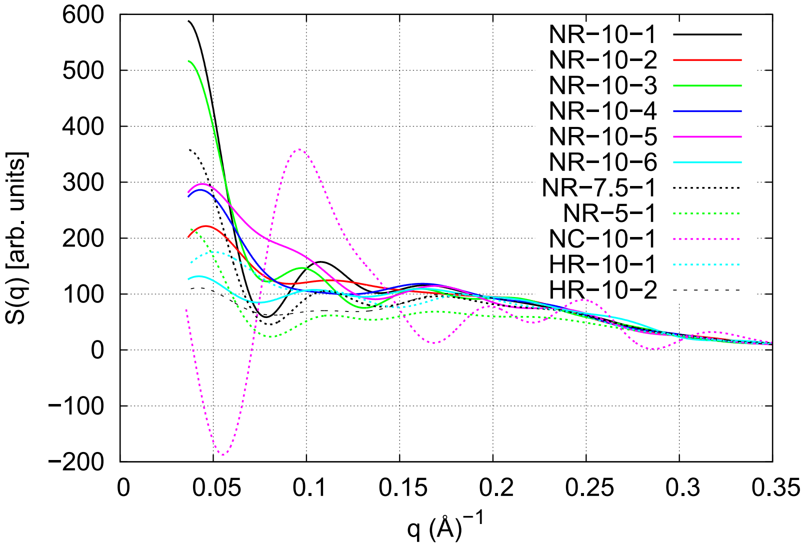


Figure 28: S(q) (X-ray) of the last configuration of various systems. The curves not shown here are given in Appendix 6.C (fig. 92).

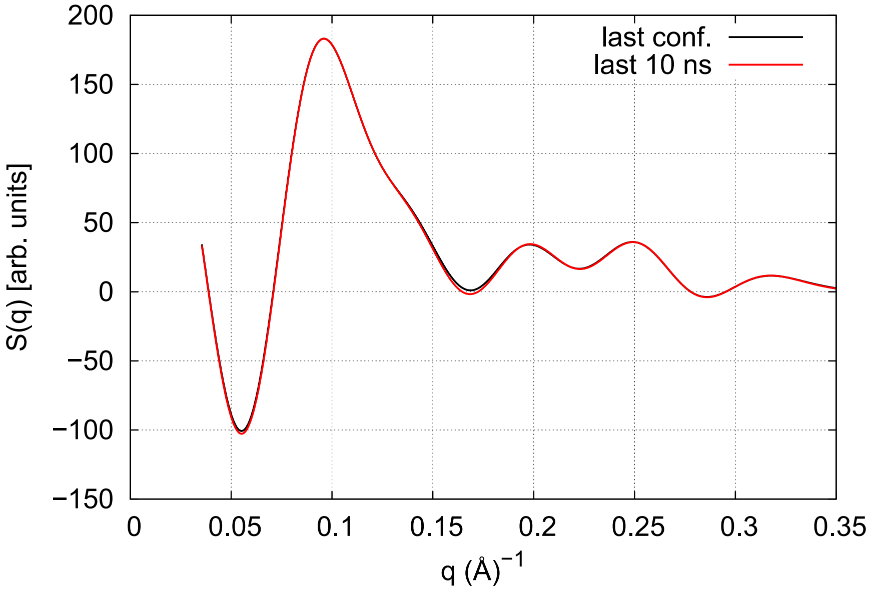


Figure 29: S(q) (neutron), for the last configuration and the last 10 ns (computed every ns, hence averaged over 10 configurations) of the NC-10-1 simulation.

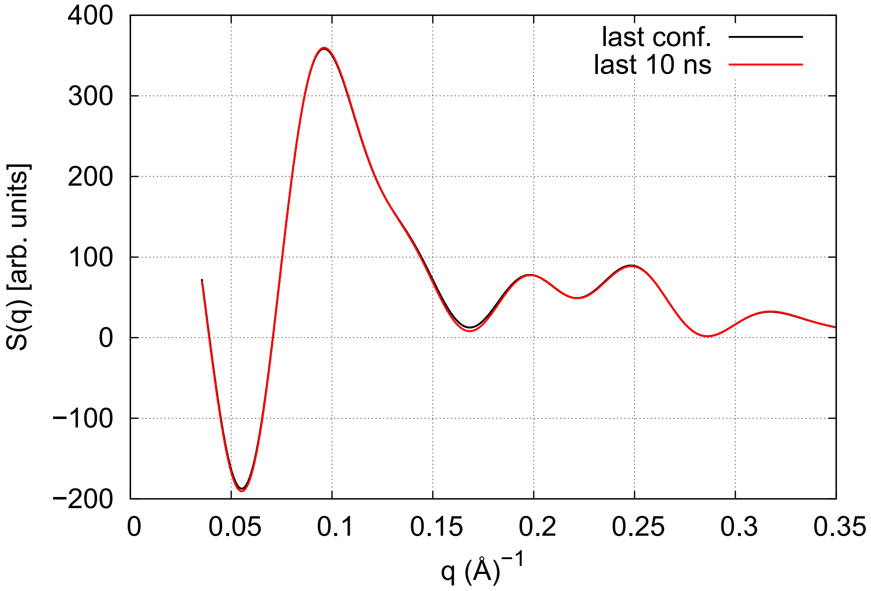


Figure 30: S(q) (X-ray), for the last configuration and the last 10 ns (computed every ns, hence averaged over 10 configurations) of the NC-10-1 simulation.

Figs. 27 and 28 also show that the position of the ionomer peak is not constant for the six NR-10 systems, in spite of the fact that they stem from the same initial geometry (random) and have the same water content ($\lambda = 10$). This means that they have slightly different apparent periodicities. This must be due to the differing initial positions of the atoms in the six NR-10 systems (see section 5.2). We can thus conclude that generating systems with similar properties, but different initial positions, leads to similar, but measurably different, configurations.

Furthermore, the position of the ionomer peak shifts to lower q-values when increasing the water content (compare NR-5-1, NR-7.5-1, and NR-10-1 in figs. 27 and 28), as expected (see above). This is in keeping with the position of the dips seen in $g_{\mathrm{Ow-Ow}}$ at large r-values (see insert in fig. 26 above) that shifts to larger r-values. Since i) water is one of the dominant species in the systems, and ii) the ionomer peak is believed to result from periodic aqueous clusters, this is not unexpected. Surprisingly, however, the systems with the deepest dips in $g_{\mathrm{Ow-Ow}}(r)$ do not exhibit the highest ionomer peaks, except for NC-10-1.

The ionomer peak is found at a lower q-value (longer distance) for NC-10-1 than for the other systems. This is again in accordance with the difference in the dip position for the $g_{\mathrm{Ow-Ow}}(r)$ (see section 6.1.2 above). This means that the average spacing between the aqueous domains is larger for NC-10-1 than for the other systems. This is not unexpected, since the water molecules have been placed initially with a density higher inside than outside the cylindrical channels (see section 5.3). The position of the ionomer peak for the Hyflon systems (HR-10-1, HR-10-2) is in the same q-range as that of other Nafion systems, no significant difference could be detected.

Fig. 31 shows a typical experimental scattering profile of a Nafion membrane soaked in water obtained from SAXS and USAXS measurements [5], together with the computed $S(q)$-profile of NR-10-1 (after equilibration). As mentioned above, we compare only the regions for which q is large enough, say, $q > 0.08$ Å^{-1}. It is seen in fig. 31 that the computed scattering profile is consistent with the experimental one: it reproduces both the ionomer peak around 0.1 Å^{-1} and the q^{-4} power law decay. This is, however, only true when the system has been equilibrated, as shown in fig. 32 with the scattering profile of NR-10-1 at t=0: the maximum around 0.1 Å^{-1} is shifted to lower q-values and the q^{-4} power law decay is not reproduced.

We can thus conclude that an MD simulation of a random model of Nafion soaked in water yields a structure in agreement with available experimental data. This is in some contrast with the conclusion by Voth and Knox [2], i.e. that it would be computationally not feasible, starting with a random structure, to reproduce the structure of true Nafion. To the best of our knowledge, this is the first time that a random model (in the sense of our setup, see section 5) of Nafion does reproduce the experimental scattering profile.

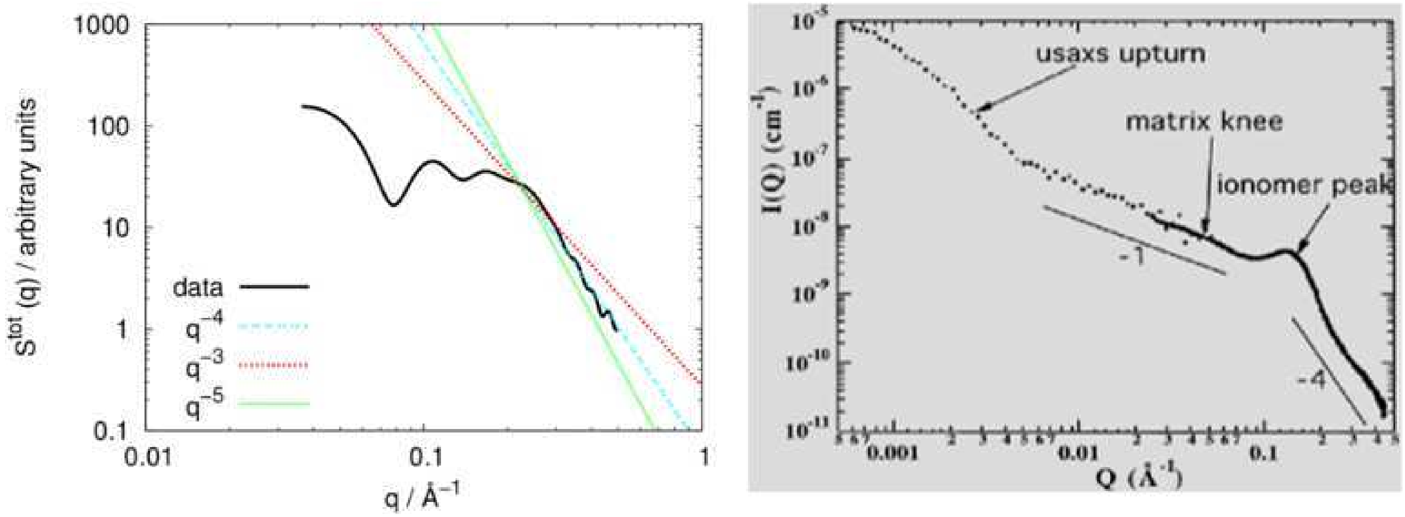


Figure 31: Log-log representation of the small-angle X-ray scattering curve obtained from: left: the random morphological model after equilibration; right: experiment [5]. The colored curves in the left frame are fits in the region 0.2 Å$^{-1} < q < 0.5$ Å^{-1}.

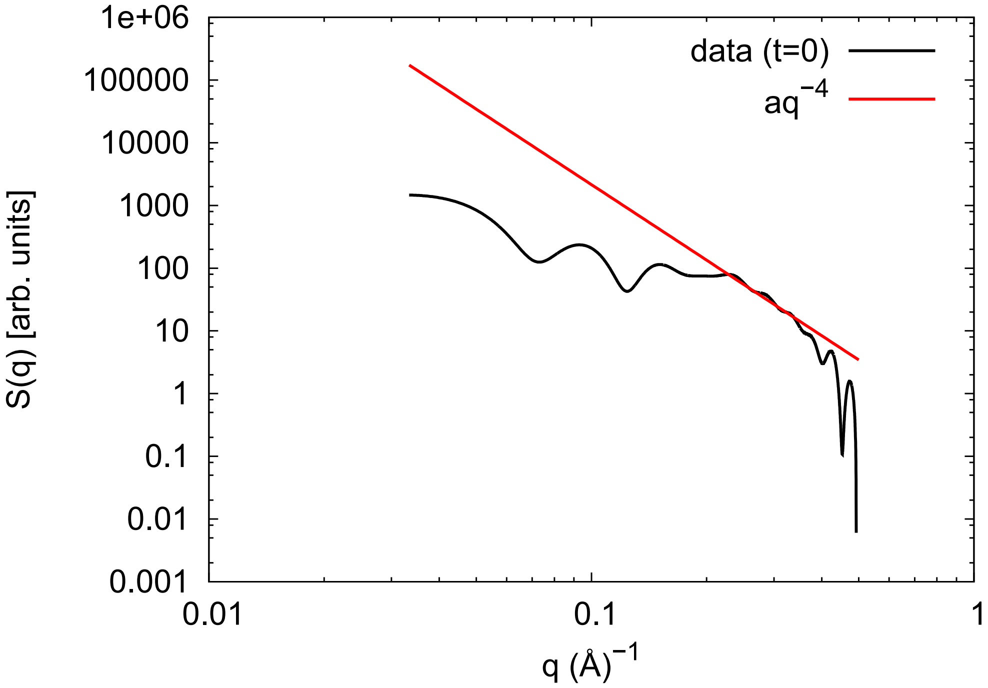


Figure 32: Log-log representation of small-angle X-ray scattering curve obtained from the random morphological model at t=0 (i.e. before equilibration).

Partial Structure Factors $S_{\alpha\beta}(q)$

The contributions of the (unweighted, see eq. 27, section 4.4) partials $S_{\alpha\beta}(q)$ are shown for various pairs in fig. 33 (NR-10-1) and 34 (NC-10-1). The curves for the C-C, C-F, and F-F pairs (left) have been multiplied by 10 in order to compare them with those for the Ow-Ow, Ow-Hw, and Hw-Hw pairs. It is clear from fig. 33 and 34 that the ionomer peak in the total $S(q)$ appears in the partials $S_{\alpha\beta}(q)$ involving both the water molecules and the CF_2 matrix. This is, however, not true for the S-S, S-K, and K-K pairs, for which the contributions to the ionomer peak in the total $S(q)$ can be considered as negligible (see fig. 33 and 34, right). The ionomer peak thus results from a periodicity between polymeric and aqueous domains, the water contribution being the dominant one.

The curves for the S-S, S-K, and K-K pairs show a peak at larger (≈ 0.25 Å^{-1}) q. Hence, there is a periodicity involving the S-atoms and K^+-ions at a length scale smaller than for the hydrophobic/hydrophilic periodicity. Assuming that all (or almost all) the K^+-ions are located in the aqueous domains and that the S-atoms are located either near the polymer/water interface or in the aqueous domains (as shown in section 6.1.5), the S-atoms and the K^+-ions must thus form some sort of 'intraclusters' [66] inside the aqueous clusters.

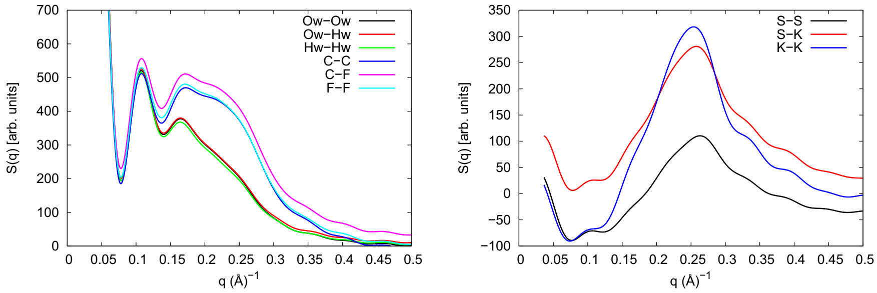


Figure 33: Partial $S_{\alpha\beta}(q)$s for different pairs, for the NR-10-1 system. The curves for the C-C, C-F, and F-F pairs have been multiplied by 10.

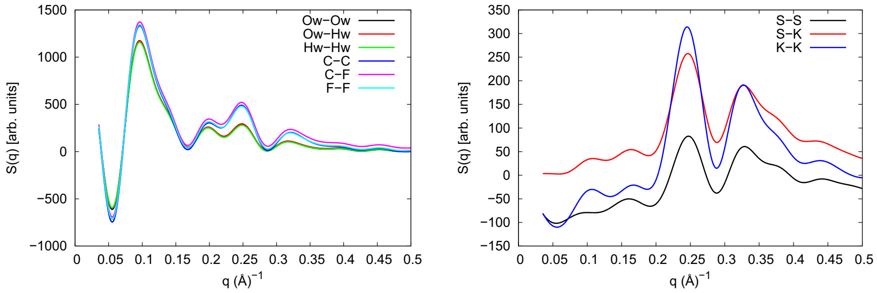


Figure 34: Partial $S_{\alpha\beta}(q)$s for different pairs, for the NC-10-1 system. The curves for the C-C, C-F, and F-F pairs have been multiplied by 10.

6.1.4 End-to-End Distances

The end-to-end distance (see section 4.4), $d_{\mathrm{n-n}}$, is a rough measure of the 'compactness' of a structure. We have investigated $d_{\mathrm{n-n}}$ of the polymer chains. Also, since shorter side chains may favor end-of-chain interactions and inhibit chain entanglement and intrachain interactions, thus affecting the resulting structure and dynamics [2], we also compare the $d_{\mathrm{n-n}}$-values of the side chains, $d^{\mathrm{side}}_{\mathrm{n-n}}$, in Nafion and Hyflon. We nevertheless expect longer side chains to lead to larger $d^{\mathrm{side}}_{\mathrm{n-n}}$-values.

Polymer Chain
Considering the first and the last C-atoms of the backbone as the head and the tail of the chain, and a C-C-C angle of 109.47°, a stretched polymer chain containing 20 side chains, periodically separated by 15 CF_2 groups, has a maximum $d_{\mathrm{n-n}}$-value of about 19 nm. Fig. 35 shows, for various systems, the evolution of $d_{\mathrm{n-n}}$ (averaged over all chains) during the equilibration (the first 1-6 ns, depending on the system, see table 1 in section 5.4) and simulation of the system. There is no significant difference when comparing the 10 random Nafion systems. Therefore we show here only the $d_{\mathrm{n-n}}$-curve for the NR-10-1 system, the curves for the other systems are given in Appendix 6.D. The size of the polymer chains is hence independent of the water content (see Appendix 6.D). Also, there are no significant differences between the curves for the Hyflon systems (HR-10-1, HR-10-2), again we show here only one system. All random Nafion systems, after a small decay during the first nanoseconds, have characteristic $d_{\mathrm{n-n}}$-values of about 5.5-6 nm. This is roughly three times smaller than the extremely elongated structure. This will be discussed further below.

Fig. 35 also shows that the Hyflon chains have initial $d_{\mathrm{n-n}}$-values larger than the Nafion ones with the random morphology (≈ 12.5 against ≈ 6 nm). The Hyflon chains are hence initially less entangled. This is a direct consequence of building the Hyflon systems at a lower density (see section 5.3). After roughly 5 ns of equilibration, $d_{\mathrm{n-n}}$ for the Hyflon chains become stable and oscillate around 8-9 nm, which is significantly higher than for the Nafion random chains. This may be a consequence of building the Nafion systems at a higher density: since the Nafion chains are initially more entangled and because of the long relaxation regime for the chains, the structures of the Nafion random systems might be frozen. This would be in accordance with the larger fluctuations seen in the $d_{\mathrm{n-n}}$-curves for the Hyflon systems. The small decay within the first nanoseconds seen for the Nafion random systems would thus arise from the relaxation (or entanglement) of a few elongated chains only.

In conclusion: starting from two quite different initial densities (and thus different $d_{\mathrm{n-n}}$-values) we found it computationally not feasible to obtain converged $d_{\mathrm{n-n}}$-values. It would be interesting to look at the $d_{\mathrm{n-n}}$-curve of a Nafion random system starting from the initial density chosen here for the Hyflon systems. It would also be interesting to investigate the influence of the solid box boundaries in our construction algorithm

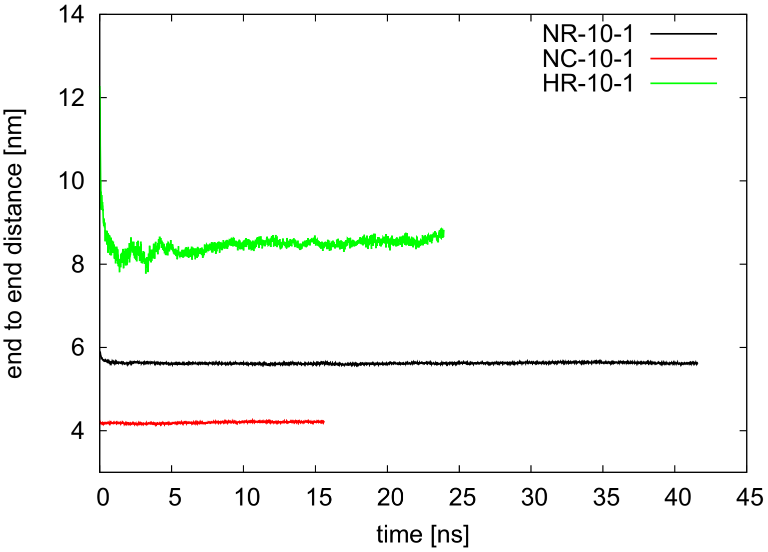


Figure 35: Average polymer chains end-to-end distance value, $d_{\mathrm{n-n}}$, vs. the simulation time, for various systems.

by using PBC instead (see section 5.3).

In contrast to the random systems, we built the cylindrical system at a density close to the experimental one (see section 5.3). This explains the lower initial $d_{\mathrm{n-n}}$-value (see fig. 35), hence the more entangled chains, for the cylindrical system: $d_{\mathrm{n-n}} \approx 4.2$ nm. Also in contrast to the random systems, there is no significant change in $d_{\mathrm{n-n}}$ within the first nanoseconds: $d_{\mathrm{n-n}}$ stays constant during the whole simulation. One could thus postulate that the initial configuration of the NC-10-1 system, resulting from our construction, is close to an equilibrated one. However, considering the very large time-scale and/or the very high energy barriers to be overcome for the backbone chains to disentangle, it is more reasonable to postulate that the structure of the polymer chains in the NC-10-1 system is akin to an entangled and frozen one.

Side Chain

For this computation, we considered the first ether oxygen atom and the sulfur atom as the head and the tail of the side chain, respectively. The evolution of $d^{\mathrm{side}}_{\mathrm{n-n}}$ during the equilibration and the simulation is shown in fig. 36, for our Nafion and Hyflon systems hydrated at $\lambda = 10$ (NR-10 and HR-10). As expected, this average end-to-end distance is larger in Nafion than in Hyflon: $d^{\mathrm{side}}_{\mathrm{n-n}} \approx 6.4$ and 3.7 Å for the NR-10 and HR-10 systems, respectively (see fig. 36).

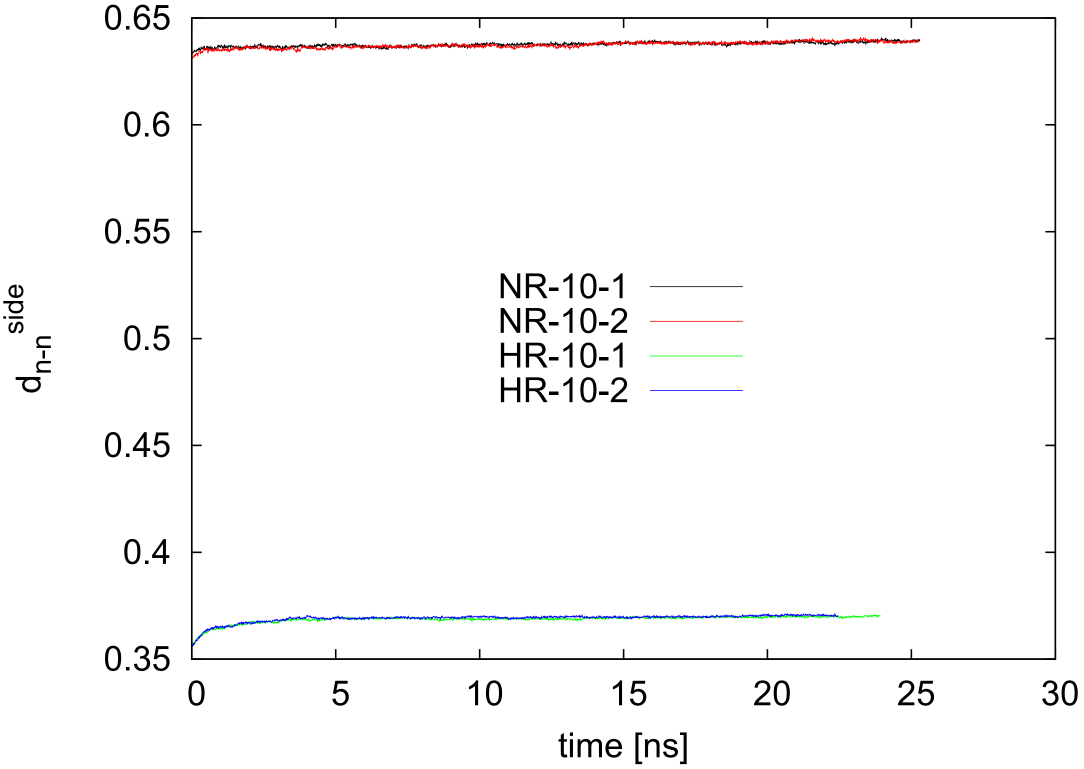


Figure 36: Average side chain end-to-end distance value, $d_{\mathrm{n-n}}^{\mathrm{side}}$, vs. the simulation time, for the Nafion and Hyflon systems.

6.1.5 Cluster Analysis

6.1.5.1 First Analysis

We have investigated the aggregation and the size distribution of the aqueous domains quantitatively using clustering algorithms (see section 4.4).

Cluster defining atoms
In our computation, we consider only the water oxygen (Ow) atoms and the K^+-ions to define the clusters. There are thus 44000 (40000 Ow and 4000 K^+), 34000 (30000 Ow and 4000 K^+), and 24000 atoms for the random systems at λ=10, 7.5, and 5, respectively. For the cylindrical system there are 43200 Ow and 4320 K^+-ions.

The sulfonate groups are known to play an important role for the properties of membranes (see section 2): they enable charge conductivity across the membrane and act as a proton (or more generally a cation) trap. The effect of adding the sulfonate groups to the 'clustering pool' will be shown below.

Distance criterion
Since there are many more water molecules than K^+-ions, we choose as the distance criterion for the cluster analysis the radial distance of the first hydration shell of the water molecules, i.e. the distance at which the $g_{\mathrm{Ow-Ow}}(r)$ curve has its first minimum, viz. 3.5 Å (see section 6.1.2). The same value was used e.g. in the analysis by Dupuis et al. [18] to determine the water percolation threshold. The effect of changing the distance criterion value will also be shown in this section.

Figs. 37 and 38 show the number of aqueous clusters and the number of atoms (in percent of total) in the largest cluster (from each configuration) as a function of the simulation time, respectively, for various systems. Since no significant differences are observed for the NR-10 systems, we show here only the curve for the NR-10-1 system. For the same reason, we show only one curve for the systems NR-7.5 and NR-5. The curves for the systems not shown here are given in Appendix 6.E (fig. 94 and 95). Typically, roughly 40 aqueous clusters are found for the NR-10 systems (fig. 37), after equilibration. Among these 40 clusters, one contains almost all (roughly 99 %) atoms (fig. 38). For liquid water at this temperature this would mean percolation (according to H-bonds), however, the structure of the water network in the biggest cluster is significantly different from that of pure bulk water, see section 6.1.2.

There is thus one dominant cluster and roughly 40 very small ones. The typical size distribution of these 40 clusters is shown in fig. 39. It shows that a small cluster can contain at maximum roughly 0.3 % of all atoms (i.e. less than 150 atoms), and clusters containing only very few atoms (say 1-10) are the most probable ones (with an exponential decay in the probability when going from 1 to 10).

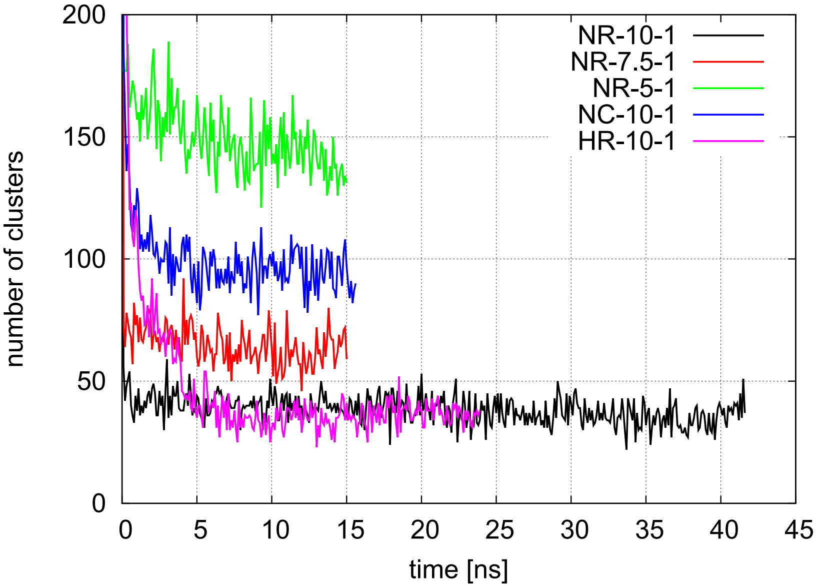


Figure 37: Number of aqueous clusters vs. the simulation time. Only Ow-atoms and K^+-ions are considered. The distance criterion is 3.5 Å. The curves for the systems not shown here are shown in Appendix 6.E.

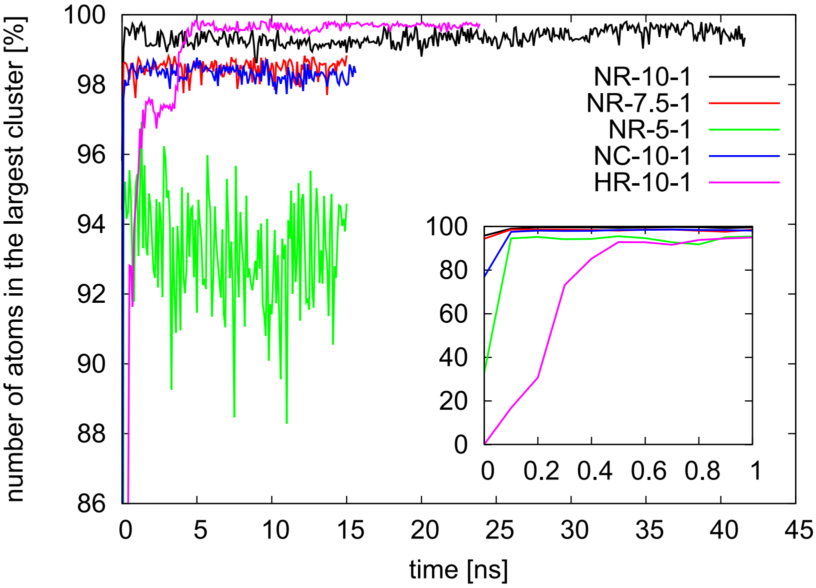


Figure 38: Number of atoms (in % of total) in the largest aqueous cluster vs. the simulation time. Only Ow-atoms and K^+-ions are considered. The distance criterion is 3.5 Å. The curves for the systems not shown here are shown in Appendix 6.E. The insert shows a zoom at short times.

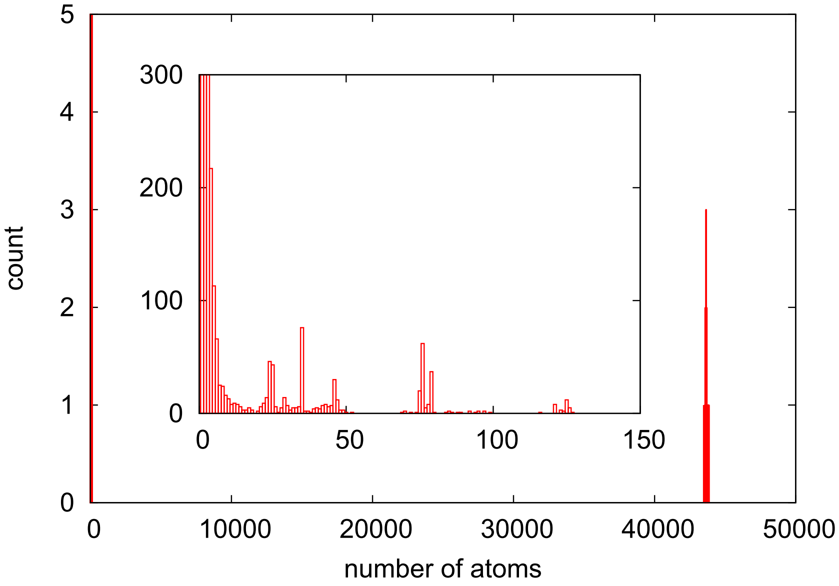


Figure 39: Histogram of the number of atoms per cluster in the NR-10-1 system, averaged over 10 ns (computed every 100 ps, hence averaged over 100 configurations). Only Ow-atoms and K^+-ions are considered; the distance criterion is 3.5 Å. The insert shows a zoom at small numbers of atoms.

We note that water percolation, according to the O-O distance criterion [18], occurs in the Nafion random systems hydrated at λ=10 (NR-10). The occurrence of water percolation was also observed at an equivalent water content e.g. in the (full atomistic) simulation study by Dupuis et al. [18], in the simulation studies (using united atom force fields) by Malek et al. [180] and by Cui et al. [181,182]. Voth and Knox [2], in their large-scale simulations to compare different morphological models, also observed the occurrence of water percolation, in all models, at a water content λ=7.4. Furthermore, in addition to observing a single large cluster of water molecules in Nafion for λ-values of 8.6 and 11.8, Cui et al. found that the water cluster size distribution is less dispersed in Nafion than in Hyflon. The effect of changing the polymer side chain length (Nafion/Hyflon) will also be shown in this section.

Variations in the cluster definition

The effect of adding or removing atom types in the 'clustering pool' on the number of atoms in the largest cluster is shown in fig. 40. More than 99 % of the atoms/ions are found in a unique cluster (we call this its population) when considering the Ow-atoms and the K^+-ions. I.e., in particular, almost all ions must be in the cluster. When using only the Ow for the analysis, the population is 97 %, which means that the water population increases when adding the ions in the analysis. Hence, the K^+-ions must play a bridging role between Ow-atoms in neighboring clusters (as defined by the Ow only).

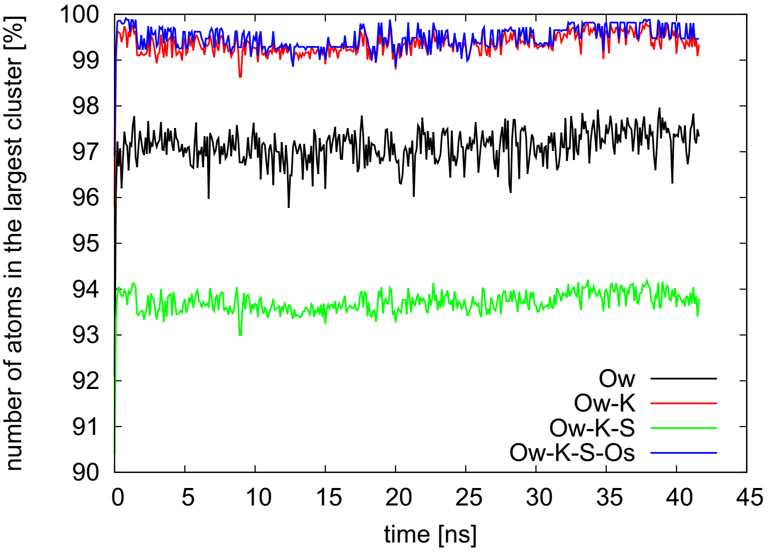


Figure 40: Number of atoms (in % of total) in the largest aqueous cluster vs. the simulation time, for various selections of atom types, in the NR-10-1 system. The distance criterion is 3.5 Å.

By the same reasoning, adding the Sulfur (S) atoms to the Ow-atoms and K^+-ions leads to a decay to 93-94 % for the population ratio. Such a ratio is obtained when roughly 75 % of 4000 S-atoms are added in the largest cluster. This result depends, of course, on the selected distance criterion. This is, however, not critical since by adding the oxygen atoms of the sulfonate groups (Os), the ratio becomes again almost 100 %. The Os-atoms must thus link the 25 % of the S-atoms found outside the largest cluster when not considering the entire SO_3^- group in the analysis. At least 25 % of the S-atoms must thus be located near the interfacial region rather than in the bulk region of the aqueous cluster.

The effect of changing the value of the distance criterion, d, on the number of clusters and on the population in the largest cluster is shown in figs. 41 and 42, respectively, for the NR-10-1 system. We have taken 3.0 and 5.0 Å as d-values, i.e. one below and one above 3.5 Å (our reference value), but still comparable to the molecular dimensions of our particles or groups. By changing the distance criterion value from 3.5 to 3.0 Å, the number of clusters increases from roughly 40 to roughly 2700-2800, and the population in the largest cluster decreases from almost all to roughly 75-80 %. On the other hand, by changing the d-value from 3.5 to 5.0 Å, the number of clusters decays from roughly 40 to roughly 5, and the largest cluster is slightly bigger. Since we do not expect a polymer chain to be present between aqueous clusters separated by (at maximum) 1.5 Å (5.0 - 3.5), we deduce that the roughly 40 small clusters found with the 3.5 Å-value are at the periphery of the very large cluster.

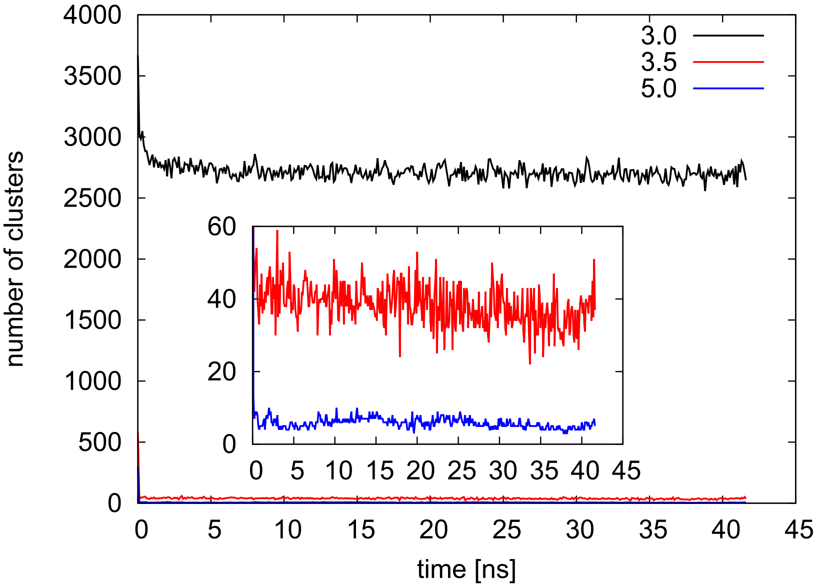


Figure 41: Effect of changing the value of the distance criterion (colors) on the number of aqueous clusters in the NR-10-1 system, as a function of time. Ow-atoms and K^+-ions are considered only. The insert shows a zoom for small numbers of clusters.

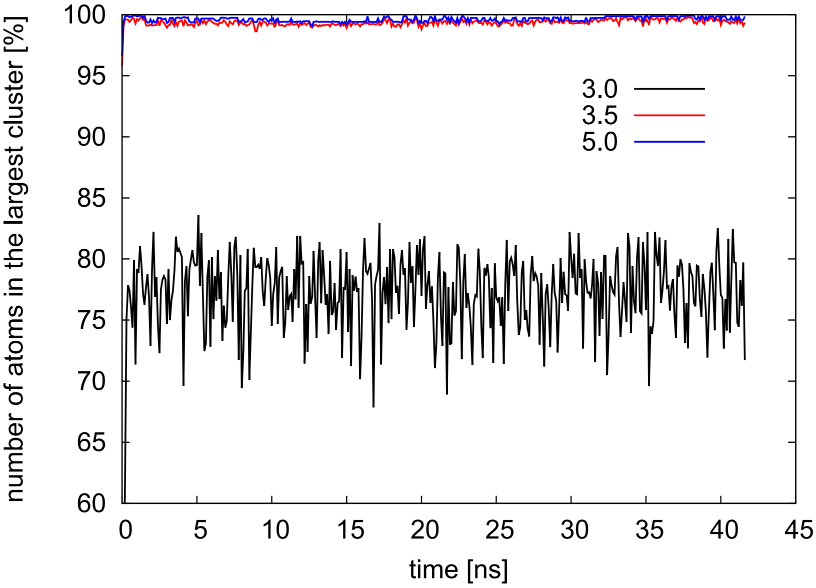


Figure 42: Effect of changing the value of the distance criterion (colors) on the number of atoms (in % of total) in the largest aqueous cluster in the NR-10-1 system. Ow-atoms and K^+-ions are considered only.

In the study by Voth and Knox [2], the same cluster analysis was performed also with 3.0 and 5.0 Å. They counted the oxygen atoms of water and hydronium to define their clusters, which is comparable with counting the oxygen atoms of water and the K^+-ions. They found for their random systems hydrated at λ=7.4 roughly 90 and 50 % of the water molecules and hydronium ions in a unique cluster with the 5.0 and 3.0 Å-values, respectively. This can be compared with the roughly 99 and 75-80 % of the water molecules and K^+-ions found in a unique cluster in our NR-10-1 system with the same values. For the systems hydrated at λ=7.5 (NR-7.5), we found, like Voth and Knox, roughly a 50 % population ratio with the 3.0 Å-value. With the NR-5 systems, not studied by Voth et al., and the 3.0 Å-value, we obtained roughly a 15 %-ratio. The number of clusters and the population in the largest cluster with a 3.0 Å-value as distance criterion are shown in Appendix 6.E (fig. 96 and 97), for all systems.

Water content λ

Decreasing the water content λ leads to an increase in the number of clusters (fig. 37), and to smaller largest clusters (fig. 38). Typically roughly 65 and 140 clusters are found for the NR-7.5 and NR-5 systems, respectively, and with largest clusters containing roughly 98.5 and 93 % of the population, with d=3.5 Å. After equilibration, water percolation thus occurs in our random Nafion systems hydrated at $\lambda \gtrsim 5$. The percolation threshold must thus occur below λ=5, for the random Nafion systems. This is below the λ=5-6 range predicted by Dupuis et al. [18]. However, this is in accordance with the λ=4 value predicted by Malek et al. [180].

Polymer chains and morphologies

Changing Nafion to Hyflon chains has only minor effects on the clustering: similar values are found for both the number of clusters and the population in the largest cluster (compare NR-10-1 and HR-10-1 in figs. 37 and 38). One difference is that the very large cluster is observed in the HR-10 system after 0.5 ns only, while after only 0.1 ns for the NR systems (see the insert in fig. 38). This is because of the lower initial density of the HR-10 systems (see section 5): at t=0, there were no clusters in the Hyflon systems with more than 1 % of all atoms. The size distributions of the clusters in the NR-10-1 and HR-10-1 systems are shown in fig. 43. They are similar, which is in contrast with the less dispersed distribution in Nafion predicted by Cui et al. [181,182].

Changing the random morphologies to the cylindrical one leads to a significant increase in the number of clusters (roughly 100 for the NC-10-1 system versus only roughly 40 for the NR-10-1 system, see fig. 37). However, like in the NR-10 and NR-7.5 systems, almost all (more than 98 %) atoms are again in a unique cluster, after equilibration (fig. 38). The roughly 100 remaining clusters in the NC-10-1 system must thus be, on average, smaller than the roughly 40 small clusters in NR-10-1. This is shown in fig. 44 with the (small) cluster size distribution of the NC-10-1 and NR-10-1 systems. It shows that the probability of having more than roughly 20 atoms in a cluster is significantly higher with the random morphology than with the cylindrical one.

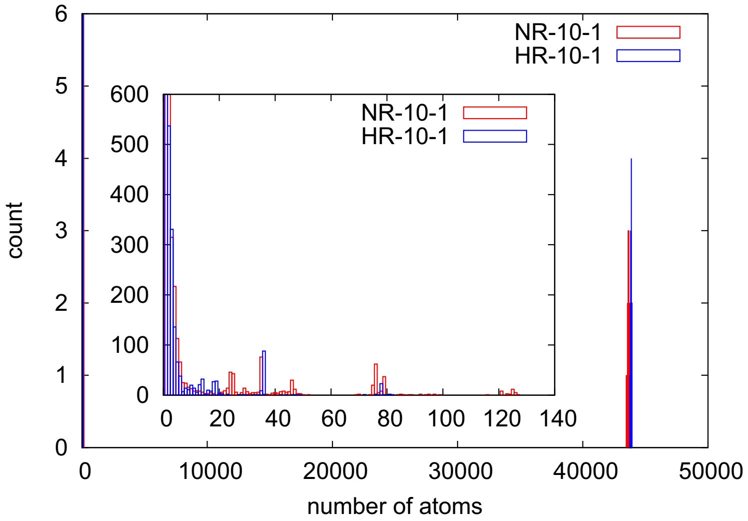


Figure 43: Histogram of the number of atoms per cluster in the NR-10-1 and HR-10-1 systems, averaged over 10 ns (computed every 100 ps, hence averaged over 100 configurations). Only Ow-atoms and K$^+$-ions are considered; the distance criterion is 3.5 Å. The insert shows a zoom at small numbers of atoms.

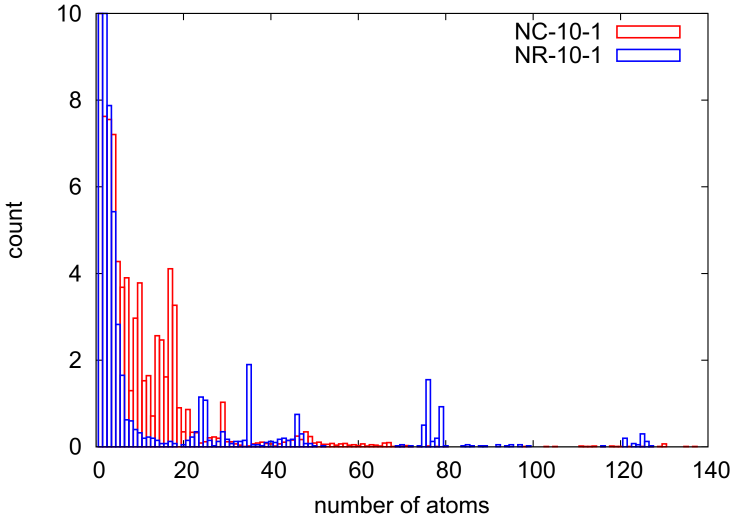


Figure 44: Histogram of the number of atoms per cluster in the NC-10-1 and NR-10-1 systems, averaged over 10 ns (computed every 100 ps, hence averaged over 100 configurations) in the data collection range. Only Ow-atoms and K$^+$-ions are considered. The distance criterion is 3.5 Å. The histograms are normalized by the number of clusters (we took 100 and 40 for the NC-10-1 and NR-10-1 systems, respectively, see fig. 37)

The larger number and smaller sizes of the isolated clusters in the NC-10-1 system is (despite the fact that we built the cylindrical system at a density higher than the random systems) probably due the fact that we built this system with regions relatively rich and relatively poor in water molecules and K^+-ions (see section 5). At t=0, 4689 clusters (for 47520 atoms) are thus found in the NC-10-1 system, while there are only 577 clusters (for 44000 atoms) in the NR-10-1 system. Their size distribution is shown in fig. 45. It is clear that there are initially many more clusters with only one atom in the NC-10-1 system than in the NR-10-1 system. The water-poor regions thus contain a considerable number of isolated clusters that do not aggregate within the period of the simulation. Nonetheless, for both morphologies, the aggregation of the largest cluster occurs within the very beginning of the equilibration (i.e. within about 0.1 ns, see the insert in fig. 38). More specifically, the largest clusters in the NC-10-1 and NR-10-1 systems contain, at t=0, roughly 77 % and 96 % of all atoms, respectively, and, for both systems, roughly 98-99 % at t=0.1 ns.

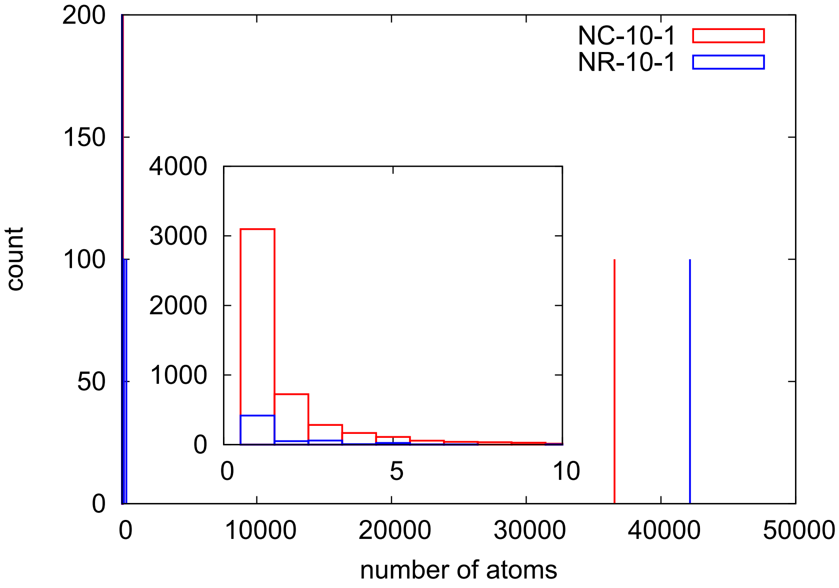


Figure 45: Histogram of the number of atoms per cluster in the NC-10-1 and NR-10-1 systems, at t=0. Only Ow-atoms and K^+-ions are considered. The distance criterion is 3.5 Å.

Summary of the first analysis

The number of clusters and the population in the largest cluster are more sensitive to λ, the water content, and to d, the distance criterion, than to the side chain length and to the morphology. The population in the largest cluster versus λ and d is shown in fig. 46. The data are averaged over the last 10 ns of the simulations and over the systems having the same water content λ. It shows that almost all atoms are in the largest cluster, except at relatively small d and λ-values.

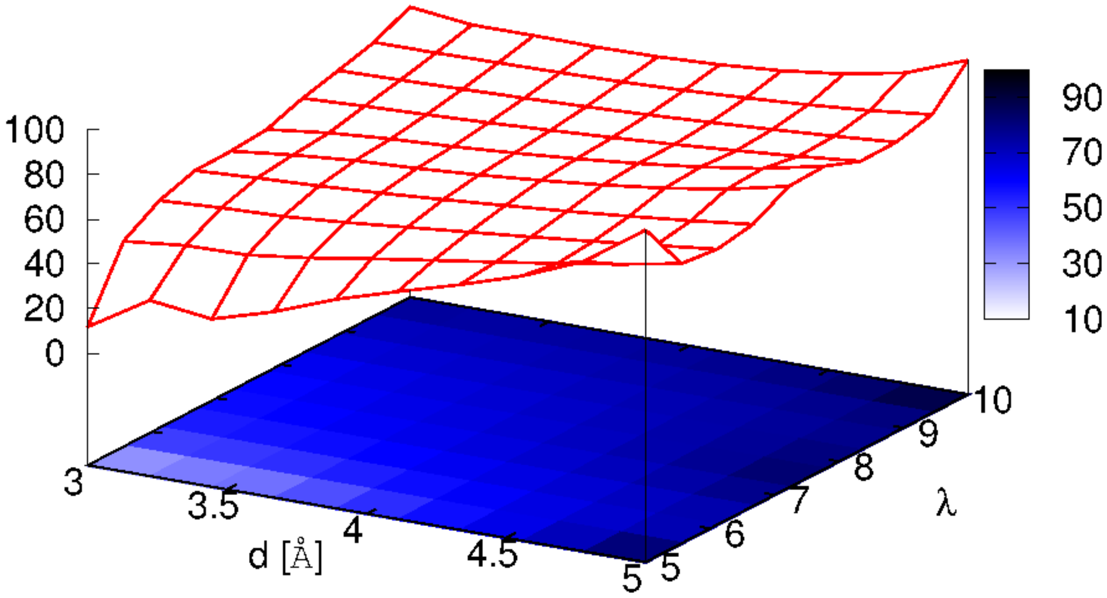


Figure 46: Population of the largest cluster (in % of all atoms) as a function of the water content, λ, and the distance criterion, d. The data are averaged over the last 10 ns and over the systems having same λ-values. Only Ow-atoms and K^+-ions are considered.

Considering only the largest cluster, i.e. in particular assuming that the small isolated clusters do not play an important role and can thus be neglected, the water percolation threshold is, according to this analysis, and with a distance criterion value > 3.0 Å, below λ=5 for both Nafion (random and cylindrical) and Hyflon polymers. Although no clear-cut transition between a percolated and a non-percolated network is seen in our data, we note that the difference between the NR-10 and NR-7.5 systems is significantly smaller than the difference between the NR-10 (or NR-7.5) and NR-5 systems. With a stricter distance criterion (i.e. 3.0 Å), not all (or almost all) atoms are found in a unique cluster. One could thus, according to an arbitrarily chosen population criterion, distinguish a percolated network from a non-percolated one. The so-obtained percolation threshold will however strongly depend on d, an arbitrary parameter.

6.1.5.2 Further Analyses of the Largest Cluster

We have extended the standard clustering routine to discretize the largest cluster from each configuration into smaller 'dense' subclusters. For this purpose we remove atoms with a 'low' coordination number. At the end we find multiple dense clusters (the algorithm is described in 4.4). In our computations, we usually remove atoms with less than and up to two neighbors (with a distance criterion d=3.5 Å). The effect of removing atoms with up to three neighbors will also be shown in this section.

Figs. 47 and 48 show the number of subclusters and the populations in the largest subcluster, as a function of the simulation time, respectively, for various systems. The population ratio of the largest subcluster is expressed relative to the total number of atoms initially in the clustering pool, and not relative to the number of atoms in the largest cluster. This is convenient since almost all atoms are found in the largest cluster (see above). Again, we show here only the curves for typical systems, the curves for all systems are given in Appendix 6.E (fig. 98 and 99). No significant difference is observed when comparing the NR-10 and HR-10 systems, the NR-7.5 systems, and the NR-5 systems.

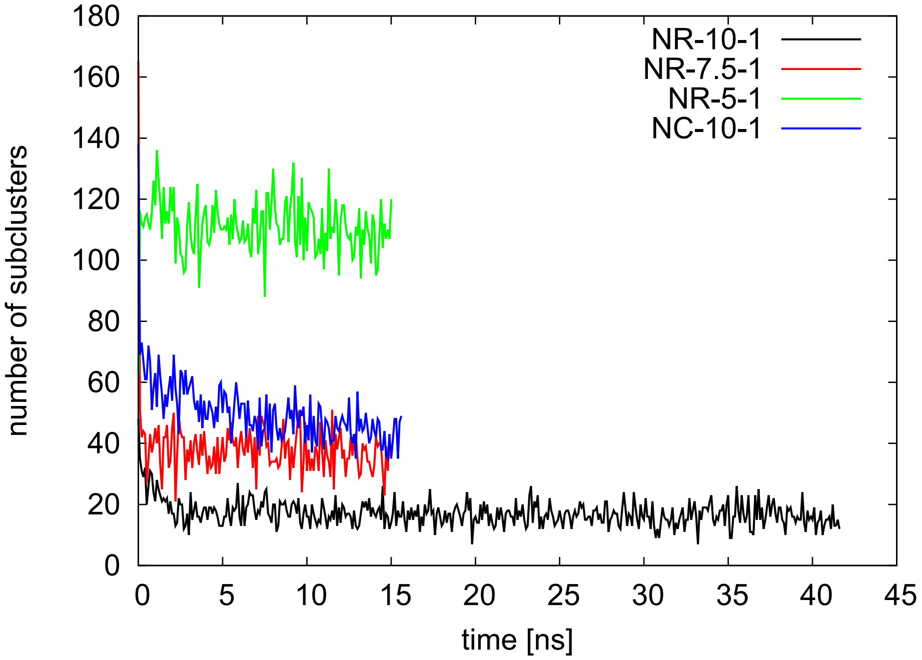


Figure 47: Number of subclusters, after iterative removal of atoms with up to two neighbors, vs. the simulation time, for various systems. Only Ow-atoms and K^+-ions are considered. The distance criterion is 3.5 Å. The curves for the systems not shown here are shown in Appendix 6.E.

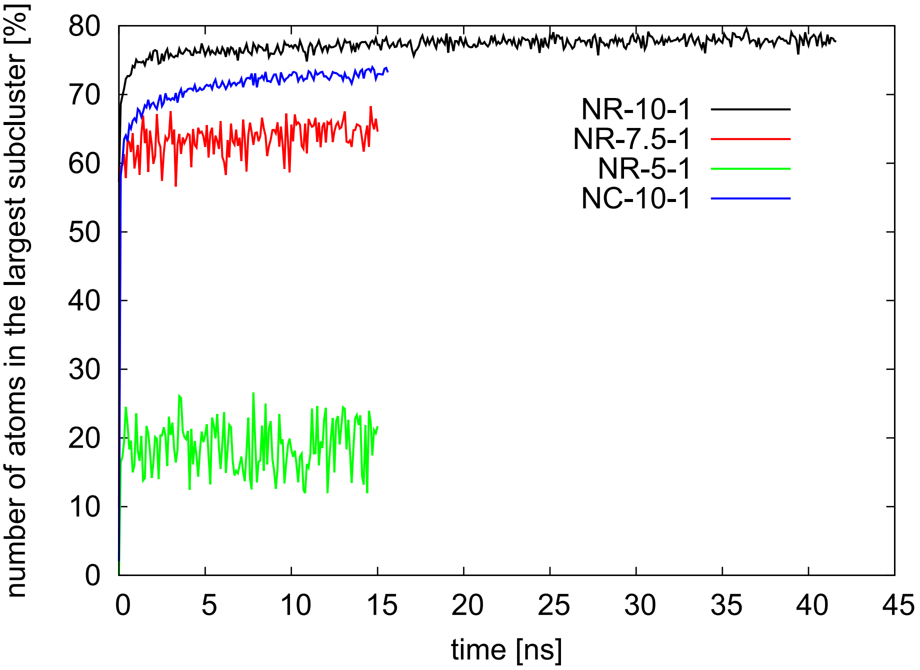


Figure 48: Number of atoms (in % of total) in the largest subcluster, after iterative removal of atoms with up to two neighbors, vs. simulation time, for various systems. Only Ow-atoms and K^+-ions are considered. The distance criterion is 3.5 Å. The curves for the systems not shown here are shown in Appendix 6.E.

For all systems, the largest cluster is subdivided by this procedure into multiple subclusters (fig. 47), yet the degree of subdivision varies strongly with λ. There are thus roughly 20, 40, and 110 subclusters for the NR-10 and HR-10, NR-7.5, and NR-5 systems, respectively. Among these subclusters, the largest one contains roughly 80, 65, and 20% of the population, respectively.

Removal of atoms with up to three neighbors

The number of subclusters and the population of the largest cluster after the removal of the atoms with less and up to three neighbors are shown in figs. 49 and 50, for all systems. The effect of removing from the largest cluster atoms with up to three neighbors is large compared to the effect of removing atoms with up to only two neighbors. The number of subclusters is consequently significantly lowered, compare figs. 47 and 49: not more than about 10-15 subclusters remain. Moreover, there are not more than 0.2 % of the population in the largest subcluster (see fig. 50). It means that most of the atoms defining the largest cluster have been removed. Hence, the few and small remaining subclusters must correspond to the densest regions in the water network.

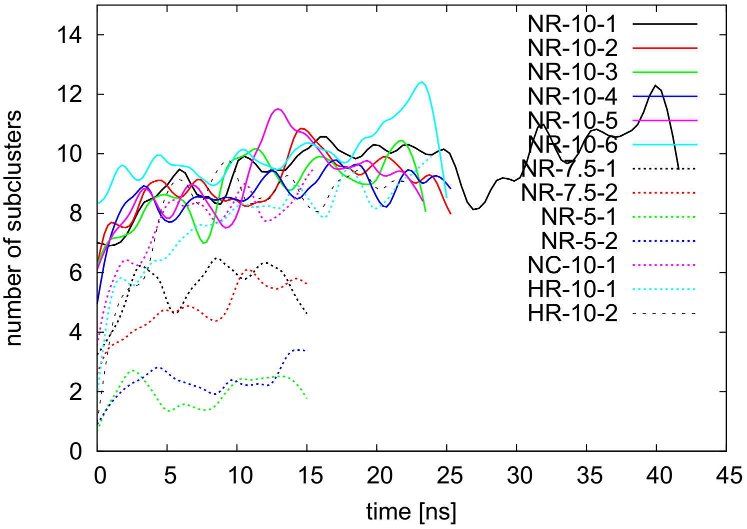


Figure 49: Number of subclusters, after iterative removal of atoms in the largest cluster with up to three neighbors, vs. the simulation time, for all systems. Only Ow-atoms and K$^+$-ions are considered. The distance criterion is 3.5 Å. The curves have been smoothed with a spline function. Compare with fig. 47.

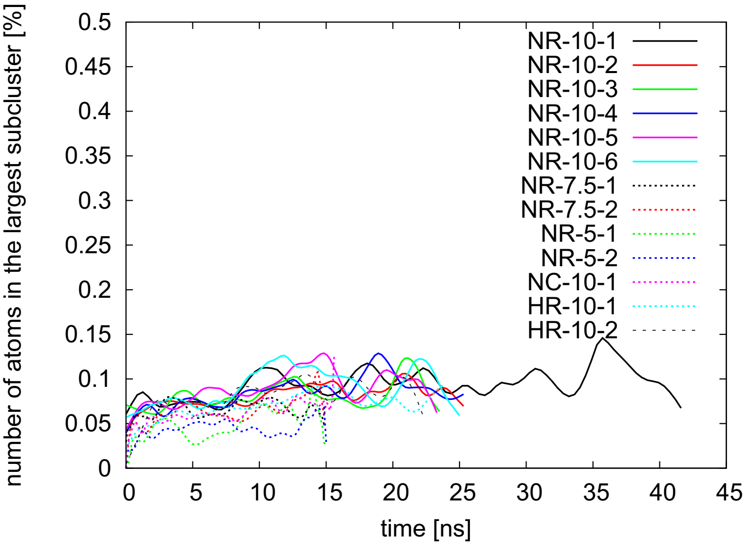


Figure 50: Number of atoms (in % of total) in the largest subcluster, after iterative removal of atoms in the largest cluster with up to three neighbors, vs. the simulation time, for all systems. Only Ow-atoms and K$^+$-ions are considered. The distance criterion is 3.5 Å. The curves have been smoothed with a spline function. Compare with fig. 48.

The spatial distributions of the Ow-atoms and K^+-ions in the largest cluster, before and after the removal of atoms with up to 2 and 3 neighbors, are shown in fig. 51. By removing atoms with up to two neighbors, the contrast between the low and high densities regions becomes more pronounced. It is, however, impossible to distinguish multiple subclusters from this picture. By removing atoms with up to three neighbors, there are only a few atoms left. These are too few to characterize the size and shape distributions of the high density regions. It is thus difficult, with the present algorithms, to distinguish aqueous domains in our systems, hence to characterize their sizes, topologies, and geometries.

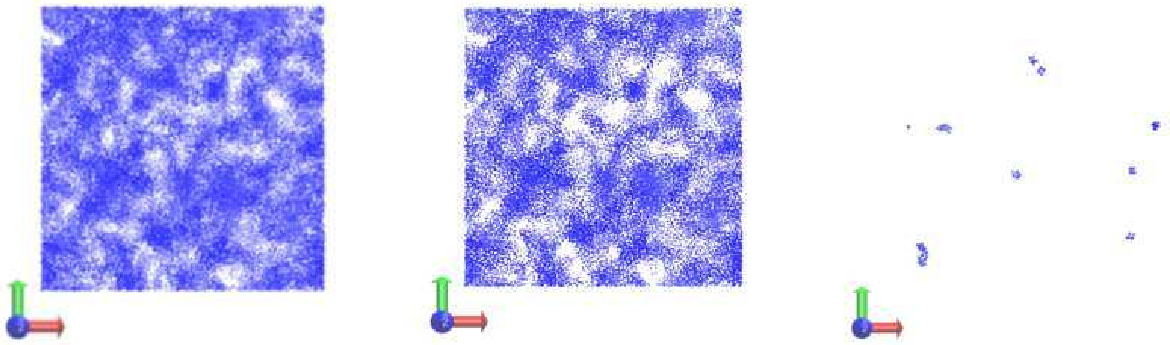


Figure 51: Spatial distribution of the Ow-atoms and K^+-ions in the largest cluster, before (left) and after removing atoms with up to two (center) and three (right) neighbors, for the last configuration of one of the NR-10 systems. The snapshots have been taken with the VMD software.

Water content λ

The increase in the number of subclusters and the decrease in the population of the largest cluster when decreasing the water content λ is shown in fig. 52 for the Nafion and Hyflon random systems. It indicates that the distribution of water molecules and of K^+-ions subdivides into more and smaller subclusters when decreasing λ. The connectivity of the water network thus seems to decreases when decreasing λ. We note in passing that the slopes for both curves are larger for $5 < \lambda < 7.5$ than for $7.5 < \lambda < 10$. This confirms that there is no detectable clear percolation threshold, yet the stronger variation in the lower λ-range seems to be, as noted above, in better agreement with [180] than with [18].

The width of the subcluster size distribution must thus increase with decreasing λ. This is the case, as seen in fig. 53, which shows the size distribution of the subclusters (averaged over the last 10 ns of the simulation) in systems NR-10-1 and NR-5-1. One can see that there is one dominant largest subcluster in both systems, and that the size distribution of the small subclusters is clearly more dispersed in NR-5-1 than in NR-10-1.

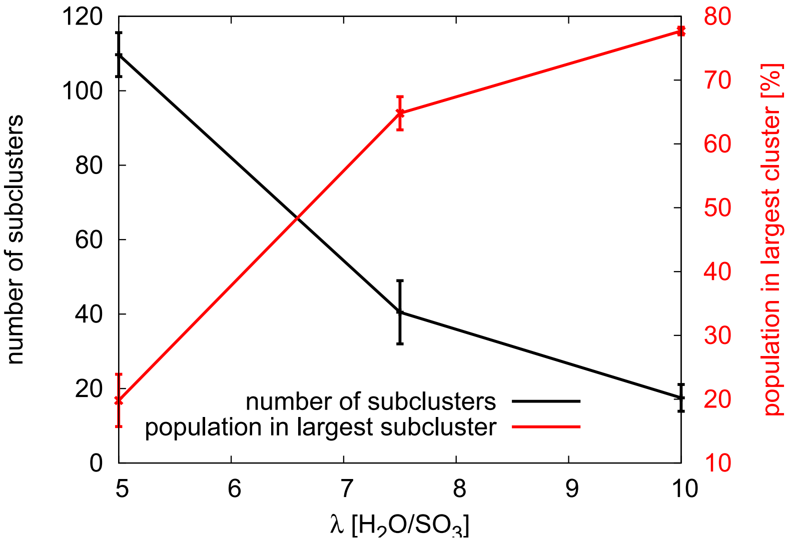


Figure 52: Number of subclusters (in black color) and number of atoms (in % of total) in the largest subcluster (in red color), after iterative removal of atoms with up to two neighbors, vs. λ. We averaged over the last 10 ns of the NR-10 and HR-10 systems, of the NR-7.5, and of the NR-5 systems. We used the root mean square deviation (RMSD, see section 4.2) function to estimate the error bars.

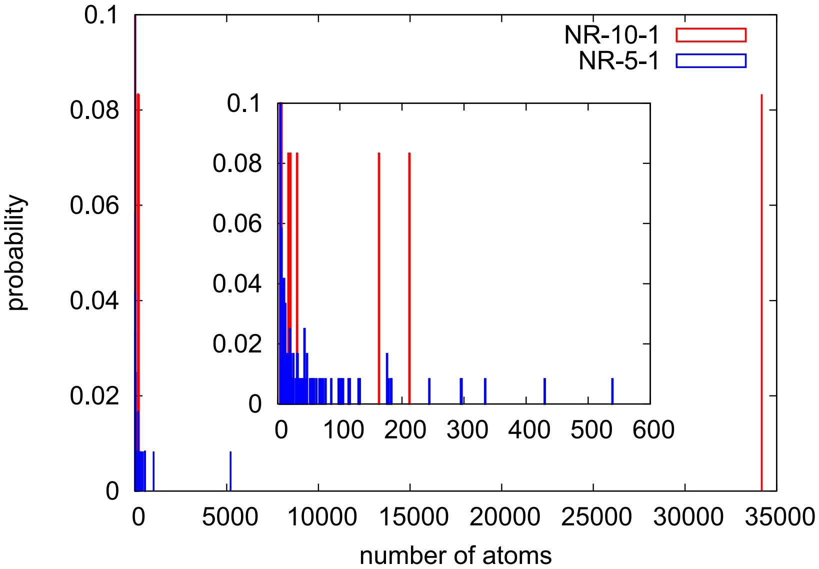


Figure 53: Probabilities of the number of atoms per subcluster in the NR-10-1 and NR-5-1 systems, averaged over the last 10 ns of the simulation (computed every 100 ps, hence averaged over 100 configurations).

Cylindrical vs. random systems
The number of subclusters in the NC-10-1 system does not seem to fully reach stability during the simulation period (see fig. 47). Nonetheless, at the end of the simulation, roughly 40 subclusters are found, like in the NR-7.5 systems. This is almost two times larger than the roughly 20 subclusters found for the systems with the random morphologies hydrated at an equivalent λ (NR-10). The larger number of subclusters is in keeping with the lower population of the largest subcluster seen in fig. 48 when comparing NR-10-1 and NC-10-1.

The populations of the largest subclusters in NR-10-1 and NC-10-1 remain nonetheless of similar magnitude: roughly 78 and 73 %, respectively. Also, the subcluster population distributions, as seen in fig. 54, are similar in both morphologies, except that there are more very small clusters in NC-10-1. This means that the spatial distributions of the water (or the connectivity of the water network) must be similar. The larger number of subclusters in NC-10-1 compared to NR-10-1 must thus arise from an only slight increase in the inhomogeneity of the water distribution. Furthermore, the number of subclusters in NC-10-1 does not seem to reach stability within the simulation: it is still decreasing. With sufficiently long simulations, one may expect that no significant differences between the two morphological systems will remain. The observed difference is thus likely due to the fact that we built NC-10-1 with a relatively heterogeneous water distribution whereas we built the NR-10 systems with an more homogeneous water distribution (see section 5).

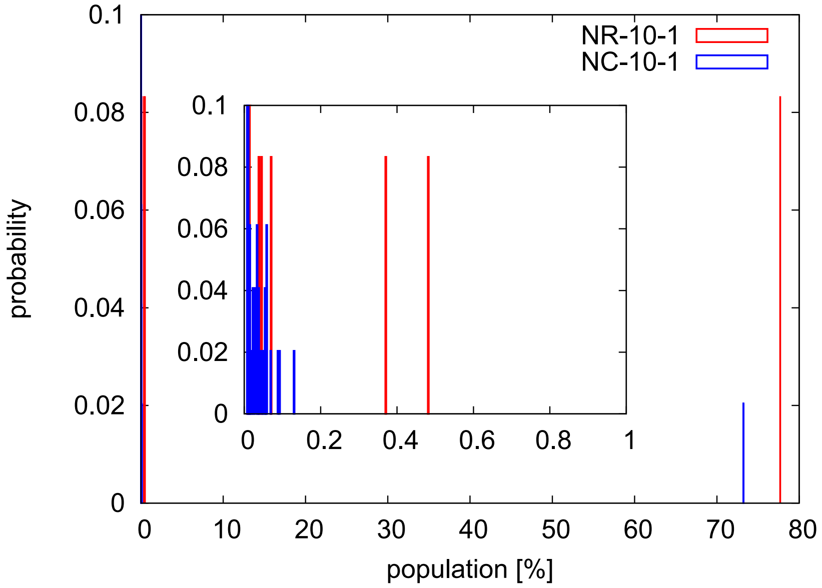


Figure 54: Probabilities of the number of atoms per subcluster in the NR-10-1 and NC-10-1 systems, averaged over the last 10 ns of the simulation (computed every 100 ps, hence averaged over 100 configurations).

6.1.5.3 Summary

We have investigated with the present analyses how water (or aqueous phase) is distributed into clusters and specifically into the largest cluster. In summary: no significant difference is observed between the systems with different random morphologies at equivalent λ. At equivalent λ, no significant difference is also observed between the Hyflon (HR-10) and Nafion (NR-10) systems. Only minor differences are observed when comparing the systems with the random morphologies with the systems with the cylindrical one: the water distribution is more inhomogeneous in NC-10-1 than in NR-10. This quantity thus seems simply not to be very sensitive, at a given temperature, to characteristics and morphologies of the polymer.

We note that, depending on the distance criterion used for the attribution of the atoms into clusters, the numerical values for the computed quantities (e.g. populations) may vary: the way to define and to determine the water percolation threshold is e.g. not unique [18]. We have checked that these conclusions are robust with respect to reasonable variations of the parameters used. We have also seen that even with the particle removal technique introduced here, it is not possible to characterize the sizes and geometries of the aqueous domains in a statistically meaningful way. Nonetheless it is clear that the lower the water content λ, the more inhomogeneous the water distribution.

6.2 Dynamics

6.2.1 VMD Snapshots

Taking 'pictures' of the simulation at different times is an intuitive way to learn about the time-evolution of its morphology. Fig. 55 shows pictures taken every 5 ns during the simulation of the NR-10-1 system. It is clear from this figure that for $t \geq 5$ ns (i.e. after equilibration), there is no global dynamics: only local structural changes occur between t=5 and t=40 ns. In other words, the simulation of the NR-10-1 system is representative of one typical configuration only. Although this configuration does reproduce in good agreement the true structure of Nafion (see section 6.1.3), we encounter the local minima problem and we do not sample the configurational space sufficiently with respect to the debate about the morphology. Considering the very long time scale of the mechanisms of interest, this was, however, expected. The fact that only local changes occur is also true for the simulations of the other systems. This is shown in Appendix 6.A (figs. 82, 83, 84, 85 and 86 for the systems NR-10-3, NR-10-6, NR-7.5-1, NC-10-6 and HR-10-1, respectively).

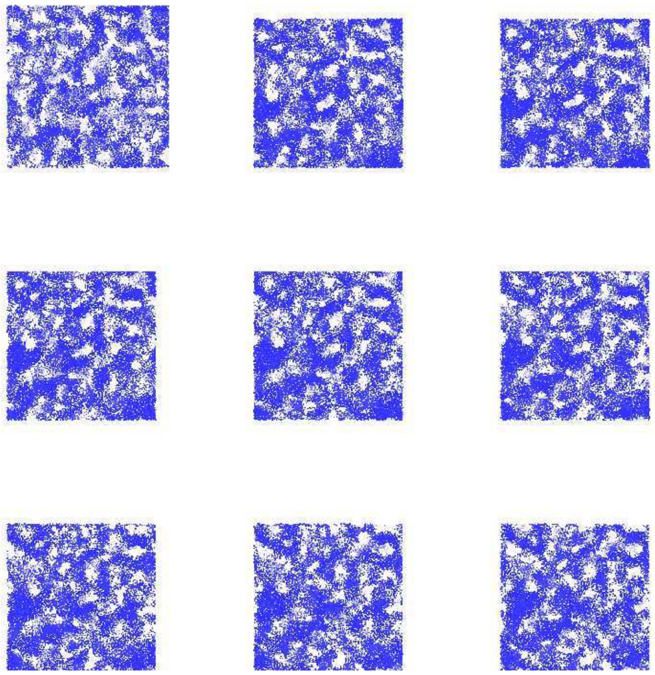


Figure 55: Pictures of the simulation of the NR-10-1 system, taken every 5 ns. Only the Ow-atoms are shown. top left: t=0; top middle: t=5 ns; top right: t=10 ns; middle left: t=15 ns; etc.; bottom right: t=40 ns. Note that the first configuration is before equilibration, thus the larger box.

The fact that no global configurational changes occur during the simulation means that conventional MD simulations are presently not able to fully sample the configurational space of such systems. We have shown that starting simulations from different initial configurations, generated with the same algorithm and differing only in the actual positions of the atoms (see section 5.2), leads, in all tested cases, to physically equivalent observables. The final configuration of these runs are, however, visibly different. This is seen when comparing the snapshots between different systems (compare e.g. figs. 55, 82, and 83). A new ensemble constructed by adding the ensembles stemming from these simulations can thus be considered a better sample of the real system's configurational space than could be obtained from a single very long simulation. This is in keeping with recent results by Elliott et al. [183], suggesting that a unified morphological description of Nafion membranes can be obtained using a combination of a model-independent approach based on the Maximum Entropy formalism and mesoscopic (coarse-grained) dissipative particle dynamics (DPD) simulations.

6.2.2 Structure Factors

The time-evolution of the structure in Nafion/Hyflon membranes can be investigated by computing the structure factor $S(q)$ at different times during the simulation. In our computation, we considered one configuration every 5 ns, starting from t=0. The time-evolution of $S(q)$ is shown in figs. 56 (neutron) and 57 (X-ray) for the NR-10-1 system, and in figs. 58 and 59 for the NR-10-2 system.

It is clear from figs. 56, 57, 58, and 59 that $S(q, t = 0)$ significantly differs from $S(q, t > 0)$. This shows the influence of the preliminary equilibration phase. At $t \geq 5$ ns, the $S(q)$s are in agreement with typical experimental scattering profiles (see section 6.1.3). There are only slight differences in the $S(q, t \geq 5 \text{ ns})$ in figs. 56, 57, 58, and 59: especially the position and the height of the maximum around 0.1 Å^{-1} (i.e. the ionomer peak) slowly converge toward the values computed for the latest time (40 ns for NR-10-1, 25 ns for NR-10-2). This means that the structures are sufficiently stable during the simulation. This is in accordance with the result from the snapshot analysis (see section 6.2.1), i.e. that after equilibration, there are only local changes but no global dynamics in the positions of the atoms.

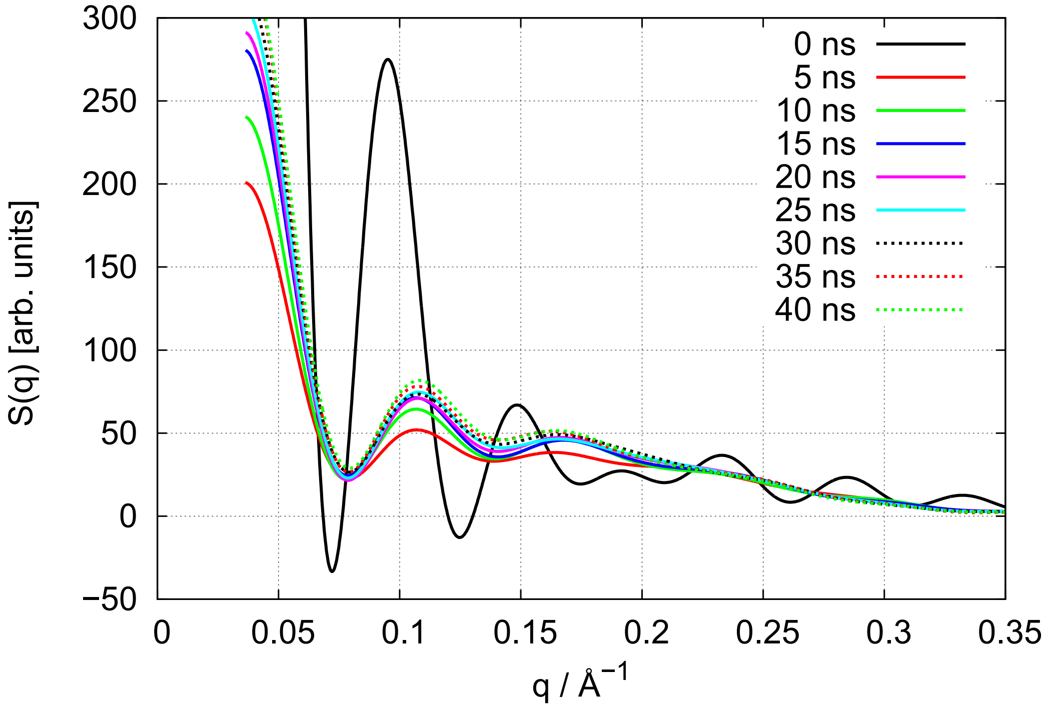


Figure 56: S(q) (neutron), computed every 5 ns of the NR-10-1 system.

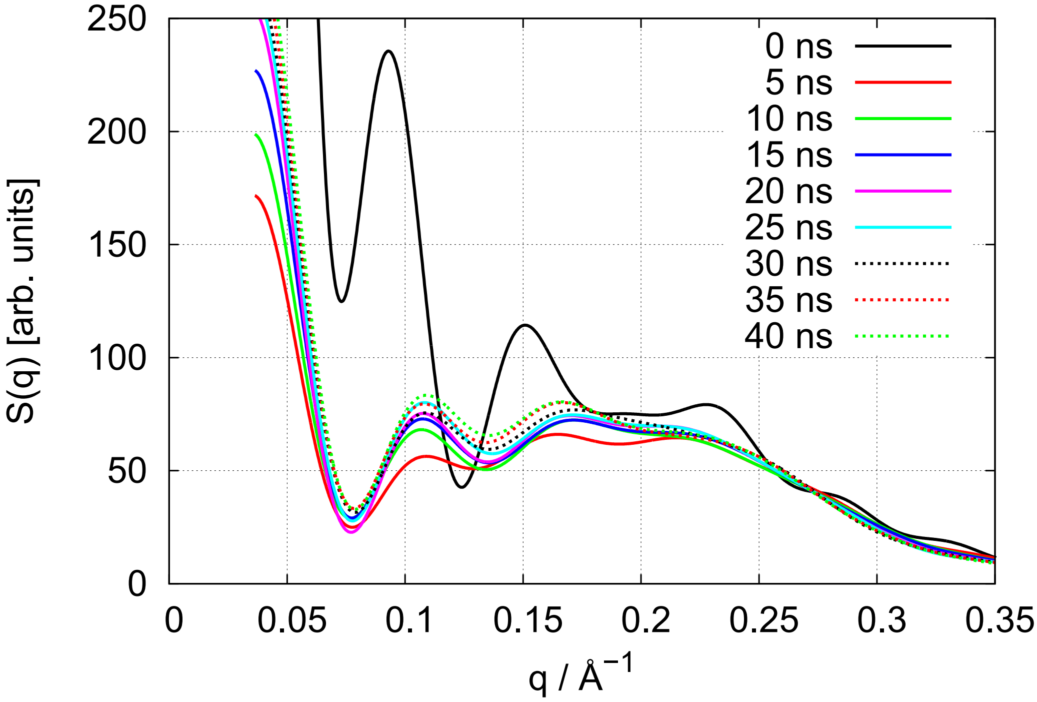


Figure 57: S(q) (X-ray), computed every 5 ns of the NR-10-1 system.

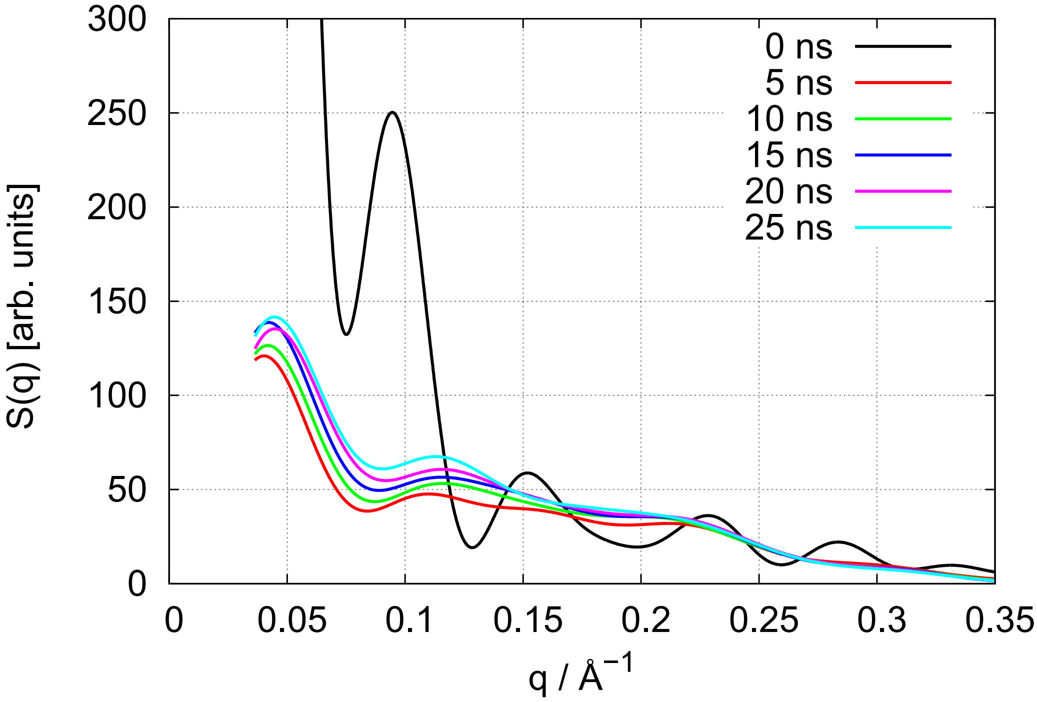


Figure 58: S(q) (neutron), computed every 5 ns of the NR-10-2 system.

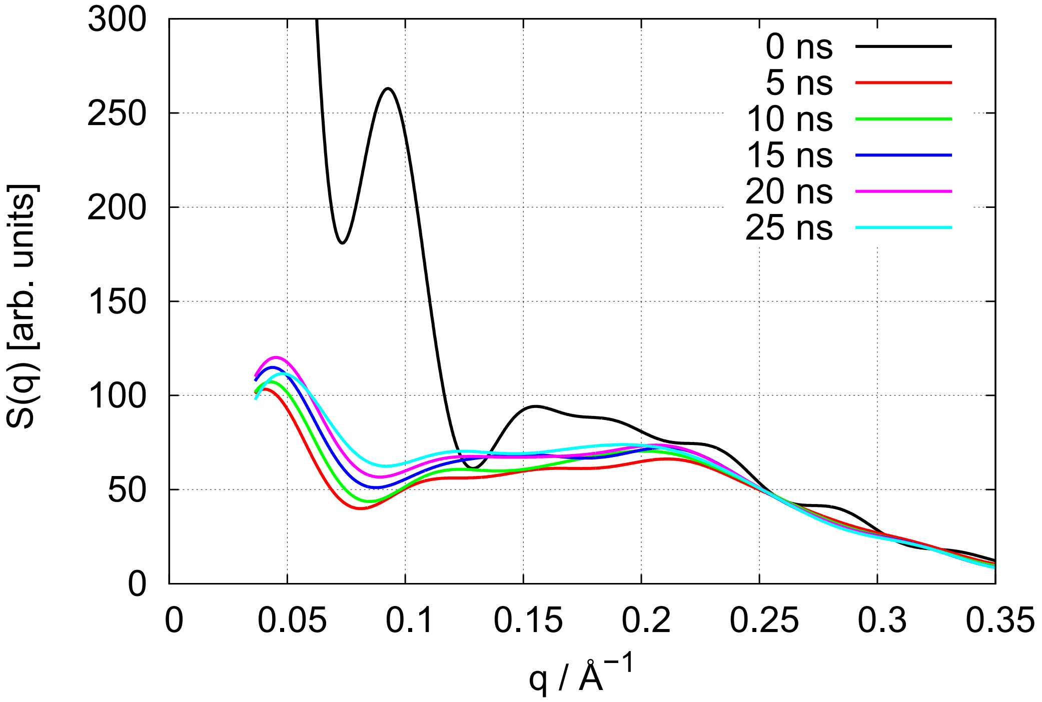


Figure 59: S(q) (X-ray), computed every 5 ns of the NR-10-2 system.

6.2.3 Mean Square Displacements

The self-diffusion of the water molecules and cations in Nafion/Hyflon systems in equilibrium is of particular interest with respect to the process of charge transport. In simulation studies the self-diffusion coefficients, see section 4.5, are routinely investigated by computing the mean square displacement function (msd). Such a self-diffusion is not easily defined for the polymer; one can, however, investigate the msds of various atoms in the polymer. In any case, a much slower dynamics is expected here compared to the liquid. Up to now no global but only local dynamical changes have been observed with respect to the polymer in the literature on simulation studies.

Water and cations

The msds of the water molecules and the K^+-ions are shown in figs. 60 and 61, for various systems. We do not show the msd-curves for all systems since many of them are similar (see Appendix 6.F, figs. 100 and 101). Also, the self-diffusion coefficients (fitted from the msd-curves over [1-9] ns using the Nernst-Einstein equation, see section 4.5) are given in table 5, for all systems.

It is seen that the water molecules and K^+-ions diffuse similarly in all systems hydrated at λ=10, independently of the polymer side chain (Nafion/Hyflon) and of the morphology (random/cylindrical): $D_{\mathrm{Ow}} \approx 0.3\ 10^{-5}\ \mathrm{cm}^2 \cdot \mathrm{s}^{-1}$ and $D_{\mathrm{K+}} \approx 0.07\ 10^{-5}\ \mathrm{cm}^2 \cdot \mathrm{s}^{-1}$. This is in contrast with the slightly higher self-diffusion coefficients of the water molecules and the hydronium ions in Hyflon than in Nafion suggested by Brandell et al. in their simulation study [58] (see above). We note in passing that the self-diffusion coefficient of water in bulk water at room temperature is roughly ten times higher than in Nafion/Hyflon [184, 185].

The self-diffusion coefficients of both the water molecules and the K^+-ions decrease with decreasing water content (see figs. 60, 61 and table 5), as expected [1, 18, 97, 186]. Fig. 62 shows the diffusion coefficients of both species as a function of λ, averaged for all side chains and morphologies (see error bars) at a given λ. One can see that the coefficients are linearly proportional to λ. Such a behavior has already been observed in other simulations and in experimental studies [18, 27, 180, 186] for similar λ-values. We recall that the water network has been shown to be percolated (with a distance criterion $d > 3.0$ Å, see cluster analysis) for all the present systems, i.e. with $\lambda \geq 5$ (see section 6.1.5). This linear behavior of the self-diffusion coefficients vs. λ is a-priori not unexpected (in our time window!). It indicates that the connectivity of the aqueous network must increase with increasing λ. This is in keeping with our results from the cluster analyses.

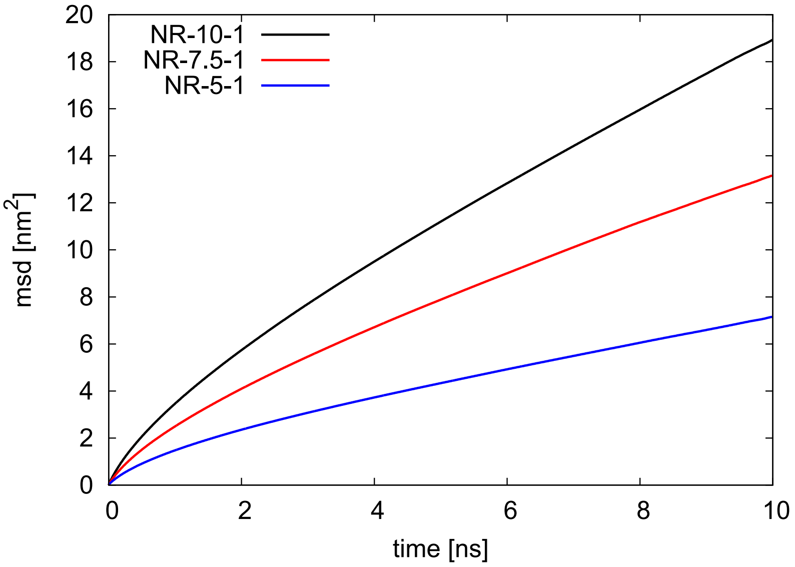


Figure 60: Mean square displacement of the water molecule oxygens (Ow) over the last 10 ns of the simulation (computed every 10 ps, hence averaged over 1000 configurations), for various systems. The curves for the systems not shown here are shown in Appendix 6.F.

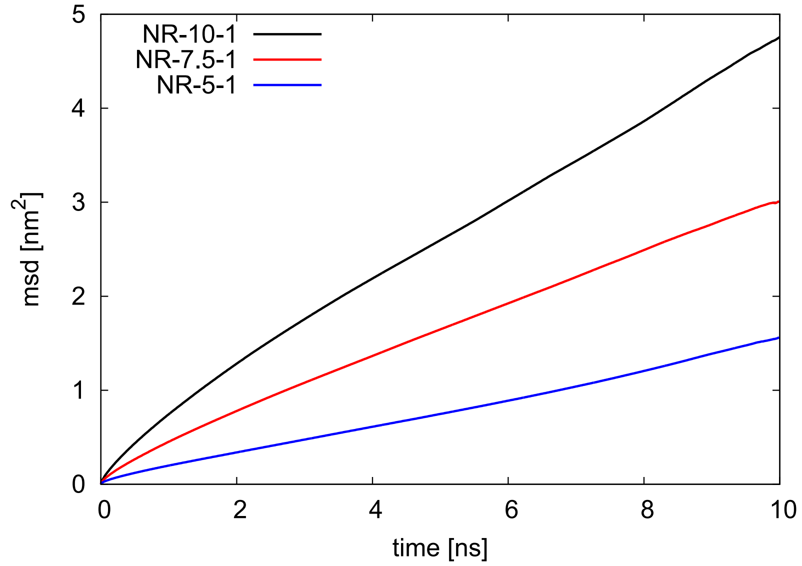


Figure 61: Mean square displacement of the K$^+$-ions over the last 10 ns of the simulation (computed every 10 ps, hence averaged over 1000 configurations), for various systems. The curves for the systems not shown here are shown in Appendix 6.F.

System	D_{Ow} $(10^{-5}\ \mathrm{cm}^2 \cdot \mathrm{s}^{-1})$	D_{K^+} $(10^{-5}\ \mathrm{cm}^2 \cdot \mathrm{s}^{-1})$	D_{S} $(10^{-5}\ \mathrm{cm}^2 \cdot \mathrm{s}^{-1})$	D_{C_9} $(10^{-5}\ \mathrm{cm}^2 \cdot \mathrm{s}^{-1})$
NR-10-1	0.29 (± 0.06)	0.072 (± 0.005)	< 0.01	< 0.01
NR-10-2	0.31 (± 0.05)	0.075 (± 0.009)	< 0.01	< 0.01
NR-10-3	0.32 (± 0.05)	0.071 (± 0.007)	< 0.01	< 0.01
NR-10-4	0.28 (± 0.05)	0.071 (± 0.005)	< 0.01	< 0.01
NR-10-5	0.28 (± 0.06)	0.072 (± 0.004)	< 0.01	< 0.01
NR-10-6	0.28 (± 0.03)	0.074 (± 0.009)	< 0.01	< 0.01
NR-7.5-1	0.20 (± 0.04)	0.048 (± 0.002)	< 0.01	< 0.01
NR-7.5-2	0.21 (± 0.04)	0.048 (± 0.004)	< 0.01	< 0.01
NR-5-1	0.10 (± 0.02)	0.024 (± 0.004)	< 0.01	< 0.01
NR-5-2	0.11 (± 0.02)	0.023 (± 0.004)	< 0.01	< 0.01
NC-10-1	0.31 (± 0.04)	0.068 (± 0.009)	< 0.01	< 0.01
HR-10-1	0.32 (± 0.04)	0.074 (± 0.009)	≈ 0.01	< 0.01
HR-10-2	0.30 (± 0.04)	0.073 (± 0.007)	≈ 0.01	< 0.01

Table 5: Self-diffusion coefficients, for all systems.

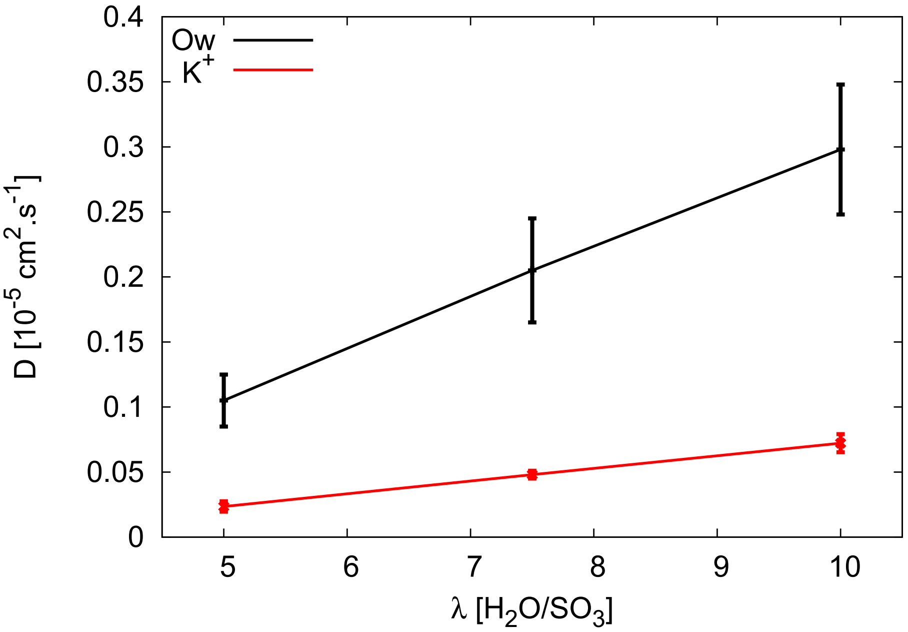


Figure 62: Self-diffusion coefficient of the Ow-atoms and K^+-ions vs. λ.

We now compare the present D-values with those obtained experimentally or from other simulation work. The present D_{Ow}-values are in good agreement with e.g. the ones obtained by Zawodzinski et al. [186] from conductivity measurements, and by Yeo and Eisenberg [187] from water-immersion experiments. Unfortunately, no experimental reference value was found for D_{K^+}. Nonetheless, the present D_{K^+}-values are almost one order of magnitude larger than those obtained with the MD simulations by Vishnyakov and Neimark [32] for equivalent water contents. Note that the present D_{Ow}-values are also one order of magnitude larger than those computed by these authors [32]. We suspect that the simulation systems in [32] were too small (only 150 SO_3^--groups and K^+-ions) [188]. Furthermore, percolation was not observed in [32], even at a water content $\lambda > 10$, which is unexpected [1, 18]. The authors thus concluded that only intracluster motions of water molecules and K^+-ions occur in their systems.

Atoms of the Polymer

Fig. 63 shows the msds of the sulfur atoms, for all systems. It is seen that the msd-curves are similar for all systems. There is, in this time window, i.e. 8 to 10 ns from the reference point, no linear regime at the end of these msd-curves. This is not what is expected for a normal diffusional process, for which the quantity msd/t should reach a constant value (proportional to the self-diffusion coefficient). The limits given in table 5 have been estimated by fitting the entire curves except for their very beginning and end. The uncertainties in D resulting from fitting in various time domains amount to roughly 50% of the very small values reported there.

In addition to the S-atoms, we computed the msds for one of the backbone atoms (specifically the 9th C-atom starting from the beginning of the backbone chain). This is seen in fig. 64. Like for the S-atoms, no clearly linear regime is observed for these msd-curves. The msds, and thus the self-diffusion coefficient, D_{C_9}, are of the same order of magnitude as for the S-atoms, i.e. in both cases at the limit of what can be detected in molecular simulations.

D_{S} and D_{C_9} are thus one or two orders of magnitude smaller than D_{Ow} and D_{K^+}, which are already smaller than the equivalent values in pure water or dilute aqueous solutions under the same thermodynamic conditions. This means that only local dynamical changes are observed with respect to the polymer. Hence, considering the huge amount of CPU-time required to disentangle such large chains, it is still today close to impossible to predict the global dynamics of the polymer.

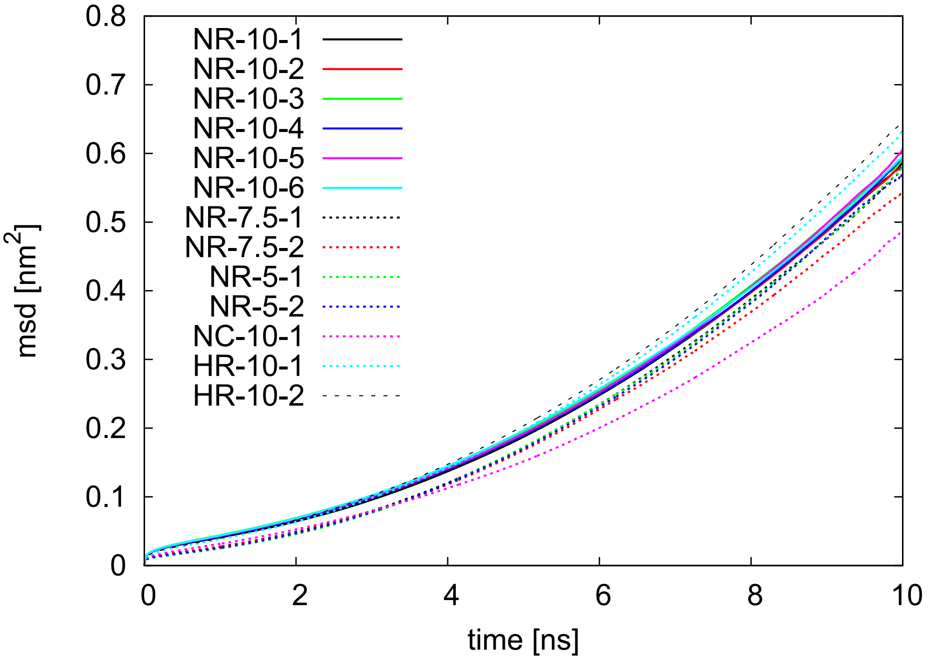


Figure 63: Mean square displacement of the S-atoms over the last 10 ns of the simulation (computed every 10 ps, hence averaged over 1000 configurations), for all systems.

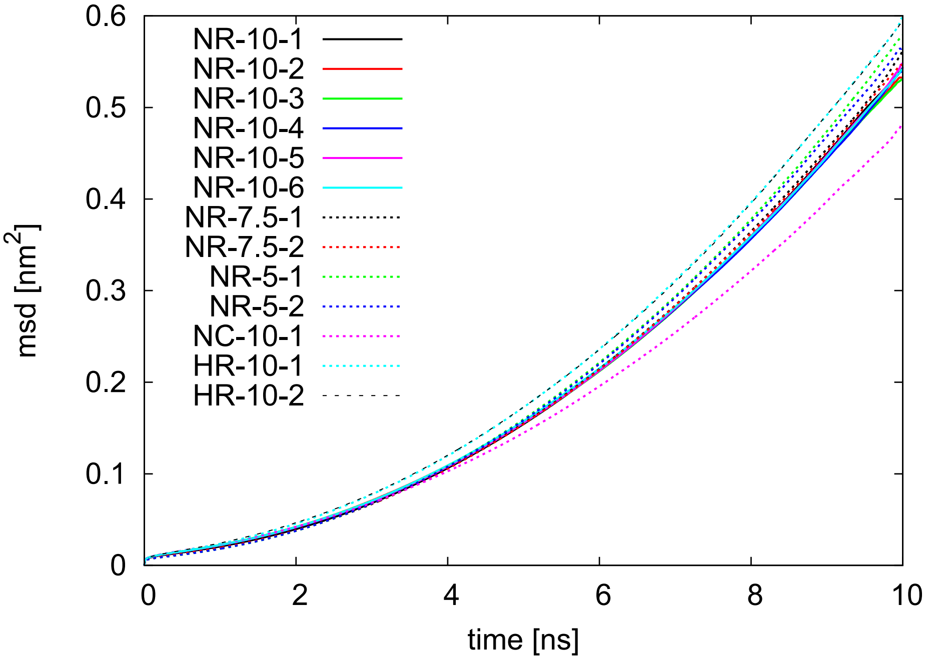


Figure 64: Mean square displacement of the C_9-atoms over the last 10 ns of the simulation (computed every 10 ps, hence averaged over 1000 configurations). for all systems.

Also, one may expect the sulfur atoms to be the more diffusive, at least at 'short' to 'intermediate' times, compared to atoms in the polymer backbone [58]. However, we find here similar absolute values and error bars for D_{S} and D_{C_9} (see table 5). We can thus not conclude, unlike Brandell et al. [58], that the absolute D-values of the atoms in the side chains are significantly higher in Hyflon than in Nafion and that these D-values increase outward along the side chain. This can be explained by the fact that Brandell et al. [58], in their systems, considered only ten side chains for each polymer, compared to twenty for us. The degree of entanglement of the chains, which can be seen as a barrier for the diffusion, must thus be larger for our polymers than the ones in [58]. Furthermore, there are only 32 polymers in [58] versus (at least) 200 in our systems. This must also increase the degree of entanglement of our chains compared to the ones in [58].

6.2.4 Cluster Analysis

The population of Ow-atoms in the largest cluster (see above) oscillates at the nanosecond time scale. This is seen in figs. 65 and 66: there are roughly 22500 Ow-atoms (with a distance criterion d=3.0 Å, see above) in the largest cluster of system NR-10-1 at t=19.5 ns, while only roughly 21000 at t=19.8 ns, more than 25000 at t=20 ns, and roughly 22000 at t=20.5 ns. Hence the formation and breaking of bridges between the largest cluster and the little clusters must occur. A similar result was obtained in [32], however, the number of water molecules in [32] does not exceed 1500, versus 40000 in the NR-10-1 system. The fact that the distribution of the water molecules into the largest and the smaller clusters oscillate can also be explained with a dynamical equilibrium between the largest cluster and the neighboring smaller ones.

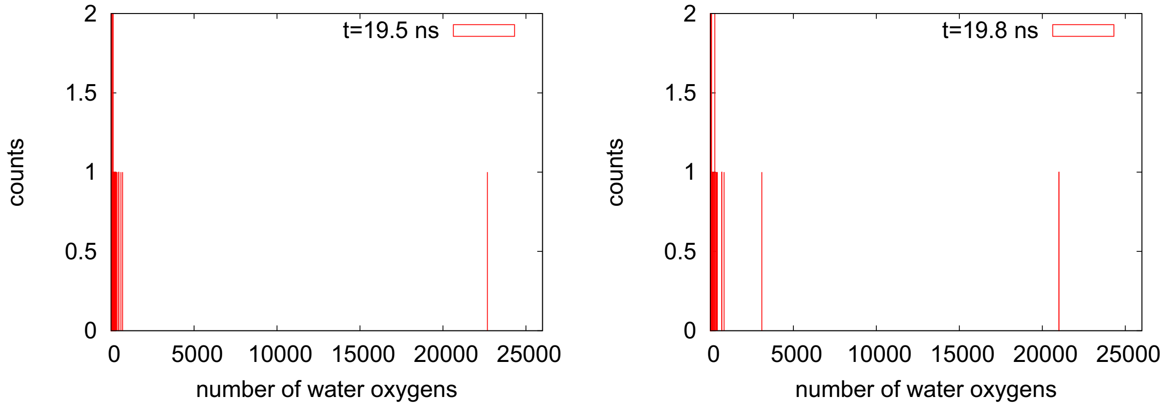


Figure 65: Histogram of the number of Ow-atoms per cluster in the NR-10-1 system at t=19.5 ns (left), and t=19.8 ns (right).

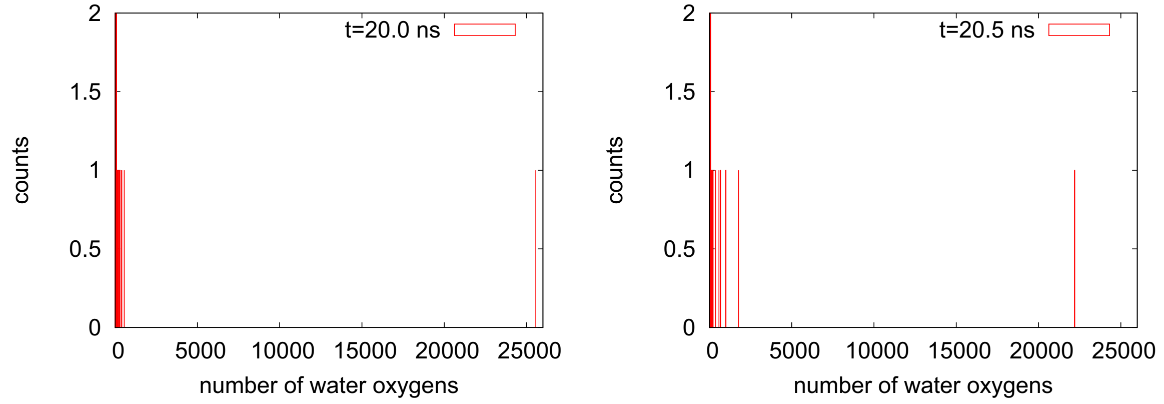


Figure 66: Histogram of the number of Ow-atoms per cluster in the NR-10-1 system at t=20.0 ns (left), and t=20.5 ns (right).

6.3 Vibrational Analysis

The frequencies of the vibrational modes in a molecule, and also of the hindered translational and rotational (librational) motions, can be compared with the ones from experiment, obtained e.g. from infra-red spectroscopy (see section 4.3). This is a way i) to test the force field and the force field parameters ii) to confirm that an appropriate value for the integration time step (see section 3.2.5) has been chosen. There are essentially two ways to compute these vibrational frequencies (see section 4.3): by Fourier transformation of the velocity auto correlation function c_{vv} (see section 3.1.3), and by computation of the normal modes (e.g. with Wilson's **GF** method [154]). We remind the reader that the Fourier transformation of the velocity autocorrelation function is called the spectral density of motions or the power spectrum.

6.3.1 Fourier Transformation of $\mathbf{c_{vv}}$

The velocity autocorrelation functions c_{vv}s were computed from the trajectory of a small hydrated Nafion system that contains one Nafion ionomer, one K^+ counterion, and 441 water molecules, and which was simulated for 5 ps in the (NpT) ensemble at room temperature and pressure using a 1 fs time step. This simulation has no other purpose than to serve as the material for the computation of the vibrational frequencies. We compared a flexible model of the Nafion molecule (i.e. without constraints on bonds and angles) with a rigid one that has fixed bond lengths. During the simulation, the bond lengths were kept constant using the "SHAKE" constraint algorithm [140]. The correlation length was chosen as half the simulation length, i.e. 2.5 ps. Fig. 67 shows $c_{\mathrm{vv}}(t)$ for the Nafion ionomer in both its flexible and rigid form. There is no unit for c_{vv} since it is normalized to 1 at t=0:

$$c_{\mathrm{vv}}(t) = \frac{\langle v(0) \cdot v(t) \rangle}{\langle v(0) \cdot v(0) \rangle} \tag{30}$$

where $\langle \quad \rangle$ means the average over the Nafion particles and over the number of time origins.

Fig. 67 shows that in both cases, the decorrelation occurs within a few 0.1 ps. Higher frequency oscillations are present for the flexible model, as expected. The Fourier transformations of the $c_{\mathrm{vv}}(t)$-curves in fig. 67 are shown in fig. 68. With the flexible Nafion molecule, the maximum wavenumber is about 1525 cm^{-1} while it is about 650 cm^{-1} for the rigidified Nafion. The bands between these two values are thus due to the stretching modes, and the broad bands below 650 cm^{-1} must thus be attributed to the bending and torsional motions of the polymer. To compare with experimental data, the spectra obtained from FT-IR (Fourier transform infra-red) spectroscopy [189] of dry Nafion shows a broad band centered at 1200 cm^{-1}. This is the highest frequency mode, it is attributed to the asymmetric CF stretch. Although our band is upshifted by about 100 cm^{-1} with respect to this experiment, we can say that our force field and parameters reproduce satisfactorily the frequency range of the vibrational modes in a Nafion (or Hyflon) molecule.

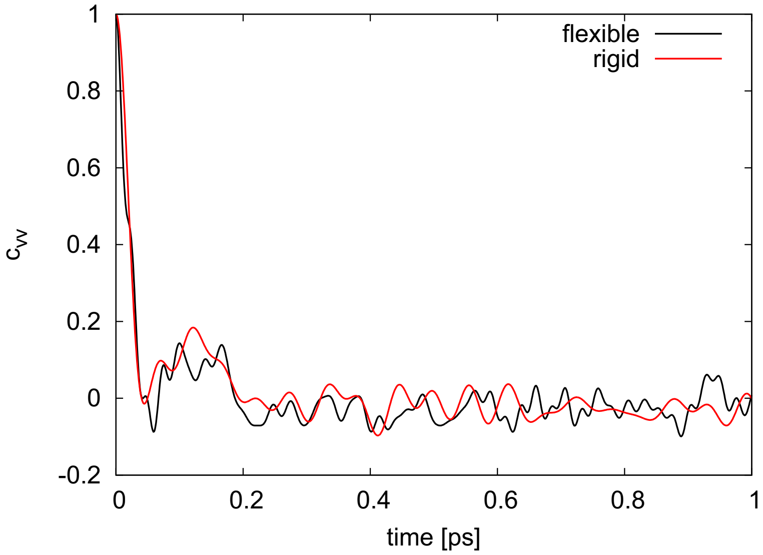


Figure 67: Velocity autocorrelation function c_{vv} for the flexible and rigid Nafion molecule in a small Nafion/Water system.

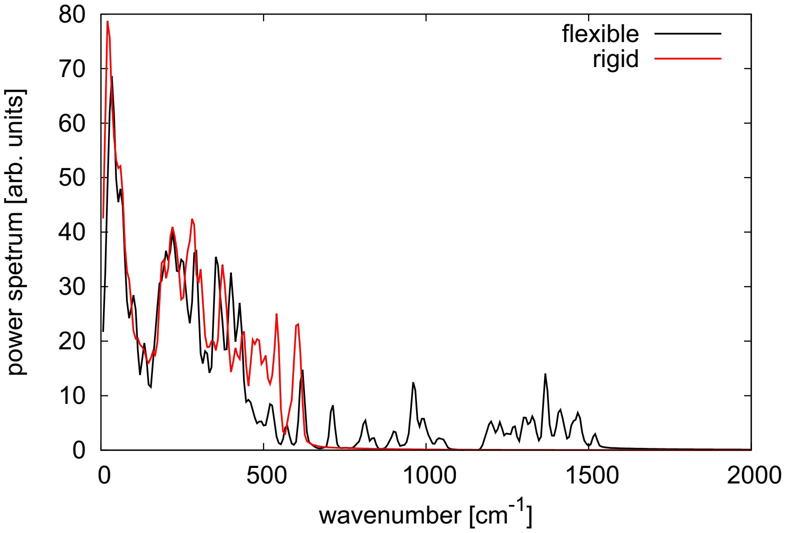


Figure 68: Power spectrum (in arbitrary units) for the flexible and rigid Nafion molecule in a small Nafion/Water system.

Fig. 69 shows the power spectra of the water oxygen and hydrogen atoms. The power spectrum of the K^+-ion is shown in Appendix 6.G (fig. 102), it displays a highest wavenumber of about 400 cm^{-1}. In agreement with the literature [190], the water oxygens have two peaks, around 50 and 200 cm^{-1}, the former one being attributed to hindered translations and the latter one, more a shoulder of the former peak, to the hindered rotational motions of the water molecules. The water hydrogens also exhibit a small peak around 50 cm^{-1}. In addition, a large band until about 1000 cm^{-1} is observed for the hydrogen atoms. This is again in agreement with the literature [190]: it corresponds to the rotational motions. Using a rigid model for the Nafion polymer, this would be the fastest motion in the system (see above), with a corresponding period of about 3.3×10^{-14} s.

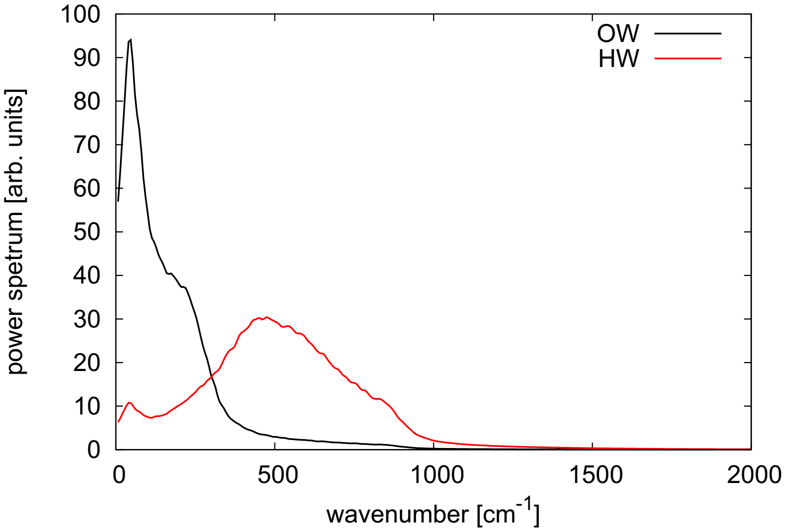


Figure 69: Power spectrum (in arbitrary units) for the Ow (water oxygens) and Hw (water hydrogens) atoms in a small Nafion/Water system.

Considerations about the integration time step:

A highest wavenumber of 1525 cm^{-1} (flexible Nafion) corresponds to a shortest period of motion of 2.19×10^{-14} s. Assuming that the integration time step δt should be smaller than one tenth [119,129] of the shortest period of motion, a 2 fs δt-value would be acceptable. For the rigid Nafion molecule, the maximum wavenumber is of about 650 cm^{-1} only, thus a time step of 5 fs would be sufficient. We obtain, however, a band at higher frequency when considering the intermolecular modes of the water molecules (since the water molecules are rigid, there are no intramolecular, or vibrational modes). A 3 fs time step would be reasonable to integrate 'correctly' these rotations of the water molecules. This is 1.5 times larger than the δt-value for the flexible model and limits the gain in CPU-time obtainable with rigid Nafion to this value.

However, constraining a bond is more CPU-time consuming than computing a simple harmonic potential since it requires an iterative process in order for the bond length to converge to its constrained value. The 1.5 gain is thus an optimistic prediction. Furthermore, a large δt usually leads to more iterations to converge to the constraints. As a result, we found that a simulation run using a 3 fs time step and the bonds constrained on the polymer was only slightly less than 1.3 times faster than an equivalent run using a 2 fs time step value without constraints. We did not find great advantages in adopting a rigid model, and therefore we choose not to constrain the bonds of our polymer and to use a 2 fs time step.

A 2 fs time step thus seems a reasonable choice. There is, nonetheless, an additional criterion on δt: the system should not violate the conservation law of energy [143] (see section 3.1.2) for isolated Hamiltonian systems, i.e. under (NVE) conditions. For this purpose, we submitted the last configuration of the simulation of the small Nafion/Water system (see above) to (NVE) runs only different by the δt-value and by the number of simulation steps (in order to have equivalent simulation lengths). Fig. 70 shows that the total system energy is maintained sufficiently constant with this time step.

It may be of interest to study the deviation of the trajectory from a 'reference' one as a function of δt. The best-possible trajectory should be theoretically (in reality this may not be the case for numerical reasons) the one obtained with the smallest δt. Fig 71 shows, as an example, the time-evolution of the x-coordinate of a given water oxygen atom within the first 0.5 ps of a simulation, for different δt-values. We also computed (see fig. 72) for the same simulations the distances between equivalent atoms in the reference (δt=0.1 fs) and in the given trajectory, for each δt-value:

$$\frac{1}{N}\sum_{i=1}^{N} r_i(t)$$

where N is the number of atoms in the system, r_i is the distance between the positions of the atom of index i in the given and in the reference trajectory, and t is the time. It shows that the larger δt, the larger is the deviation from the correct trajectory, as expected.

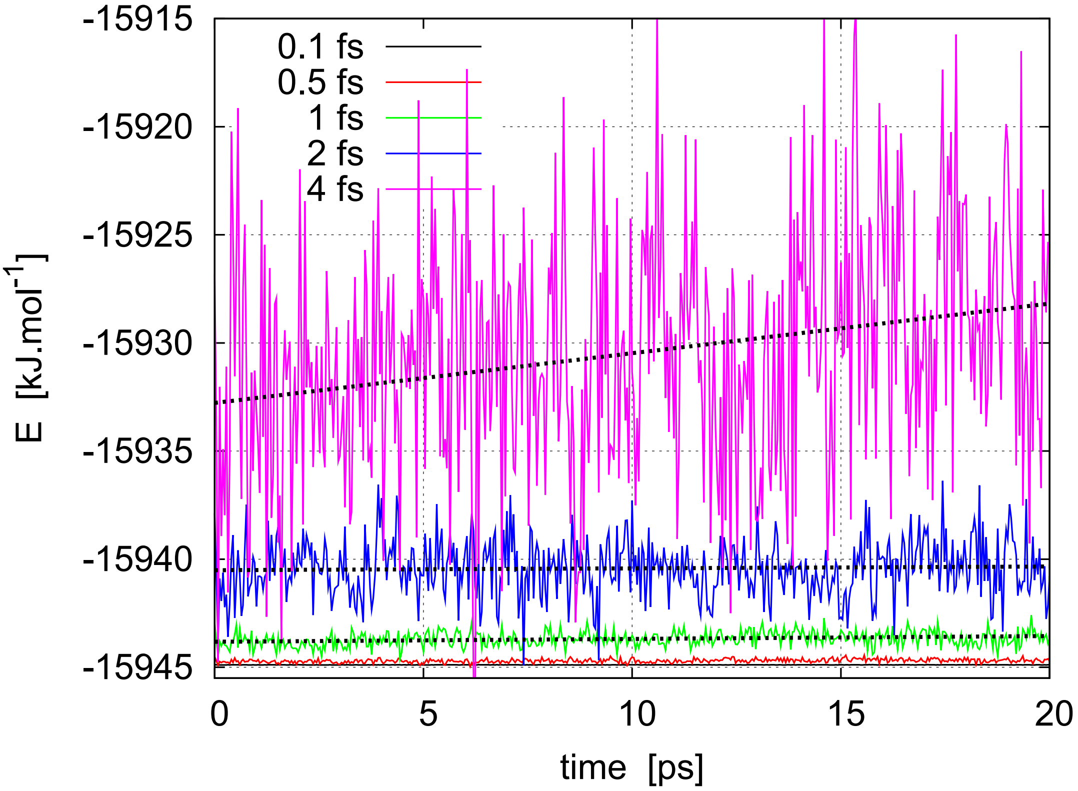


Figure 70: Time evolution of the total energy of a small Nafion/Water system, using different δt-values. The black dashed curves are fitted linear functions.

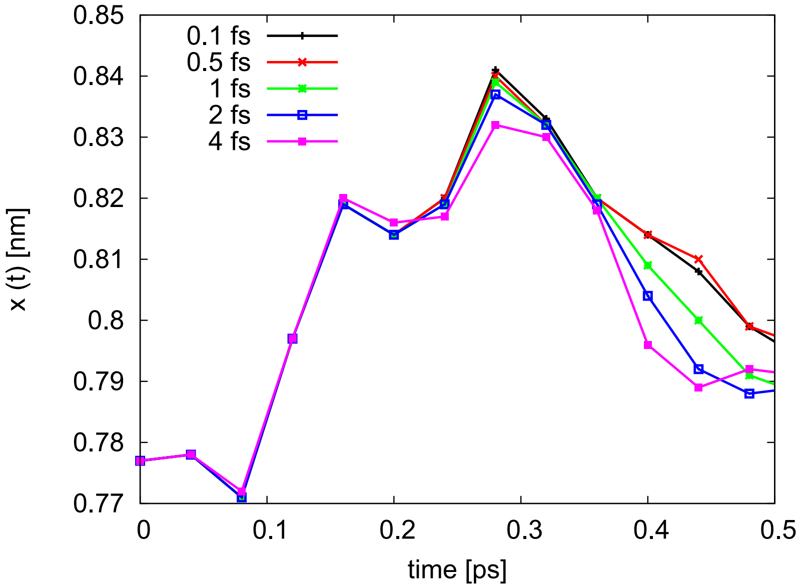


Figure 71: Time evolution of the x-coordinate of a given Ow-atom in a small Nafion/Water system, for different δt-values.

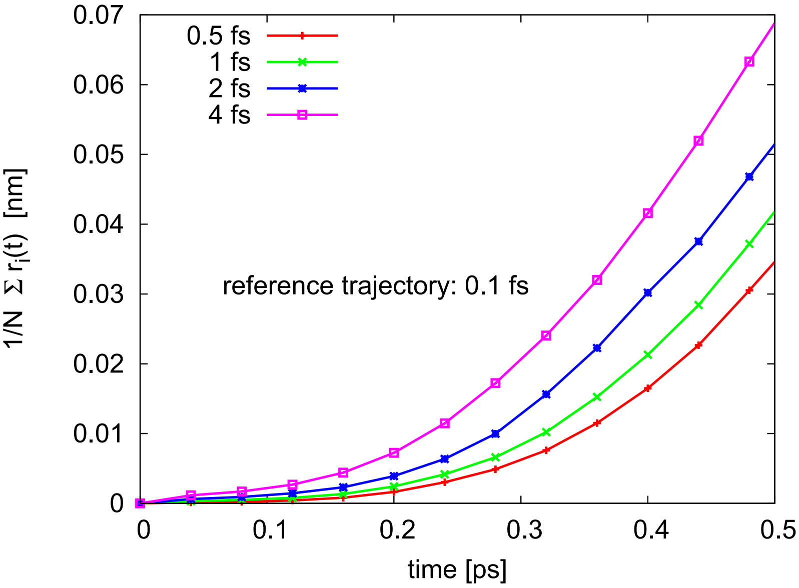


Figure 72: Deviation from the 'correct' trajectory of a small Nafion/Water system.

In summary: The a-coefficients from the fitted $ax + b$ functions in fig. 70 (we call them drift coefficients) shown in fig. 73 are negligible for $\delta t \leq 2$ fs. The deviations from the reference trajectory also seem acceptable [143]. Finally we thus took, see section 5.2.3, $\delta t = 2$ fs.

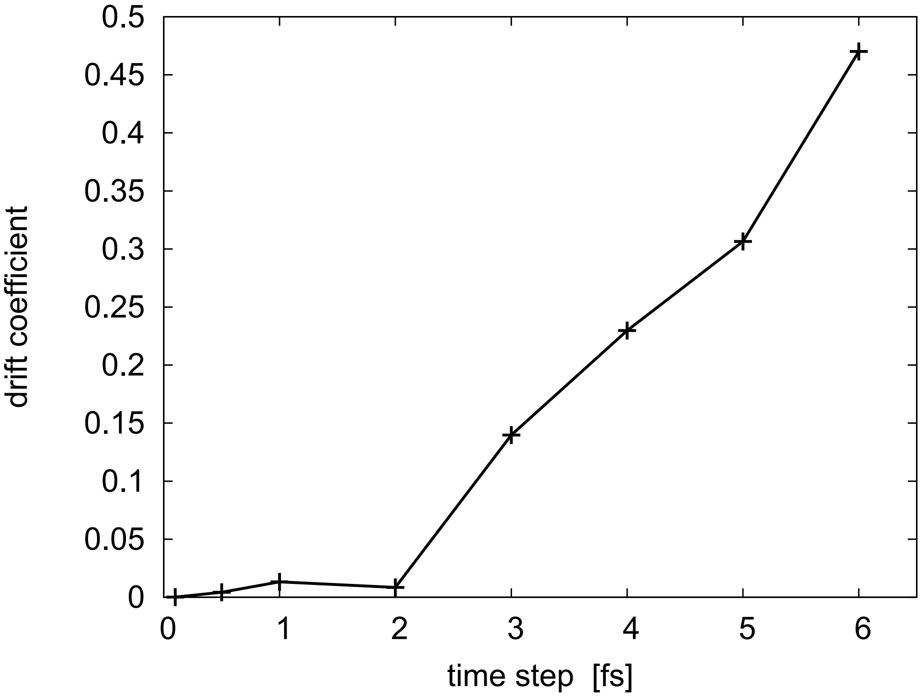


Figure 73: Drift coefficients of the total energy vs. δt.

6.3.2 Extraction of the Normal Modes

The vibrational frequencies can also be computed by extraction of the normal modes (see section 4.3). This involves, in the present implementation [129], not to use constraints. Furthermore, for a usual normal mode calculation, it is necessary to minimize the energy prior to the computation and diagonalization of the mass-weighted Hessian matrix. What tolerance for the energy minimization is required depends on the type of the system, but a rough indication is 0.001 $\mathrm{kJ \cdot mol^{-1}}$ [129]. This can be done e.g. with the L-BFGS (Limited-memory Broyden-Fletcher-Goldfarb-Shanno) [191–193] energy minimization algorithm.

We thus optimized the structure of a small (x =3 in eq. 1) Nafion molecule using the L-BFGS algorithm in double-precision. After 1660 optimization steps, the change in energy became smaller than the tolerance. Thereafter, we used the software Gromacs [129] for the computation and diagonalization of the Hessian matrix as well as the sorting of the normal modes according to their frequencies. Fig. 74 shows the eigenfrequencies (in wavenumber) thus obtained. According to this analysis, the fastest motion has a wavenumber of about 1519 cm^{-1}. This is very close to the value obtained from the power spectrum (1525 cm^{-1}, see above). Furthermore, the density of the modes is high between 1200 and 1500 cm^{-1} and even higher below about 500 cm^{-1}, in very good agreement with the intensities in fig. 68. We therefore conclude also here that the force field and the parameters used to reproduce Nafion (and subsequently Hyflon) polymers are adequate.

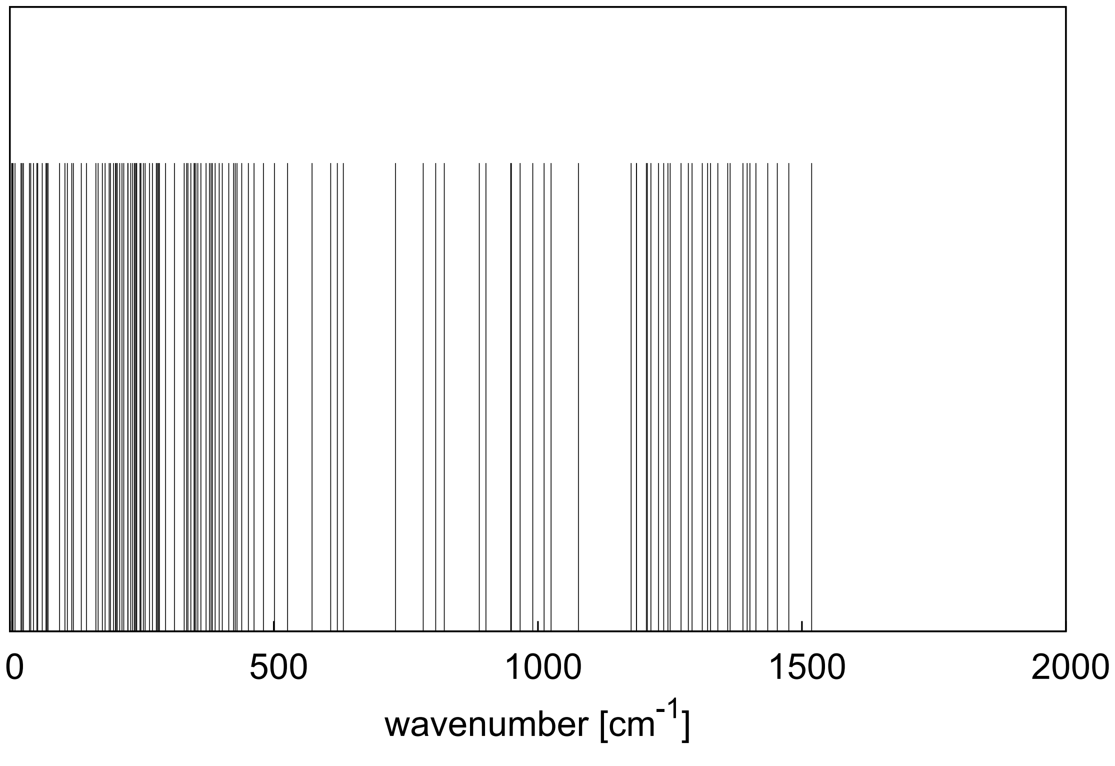


Figure 74: Eigenfrequencies obtained from the extraction of the normal modes of a Nafion molecule.

Although it is not possible to perform such a normal mode analysis on a system with constrained bonds, one can, nonetheless, 'separate' the stretching modes from the bending modes using arbitrary force constants. This is illustrated in fig. 75 with force constants for the bonds 10 times larger than in fig. 74. No frequency remains in the range [655 - 1635] cm^{-1}, the modes shifted to higher wavenumbers are thus stretching modes, as already seen above in the power spectrum for the rigid Nafion. Fig. 75 also shows that the eigenfrequencies below approximately 655 cm^{-1} differ slightly from the ones in fig. 74. This can be explained by the fact that each normal mode frequency depends (in principle, except for symmetry) on all moving masses and force constants in the system.

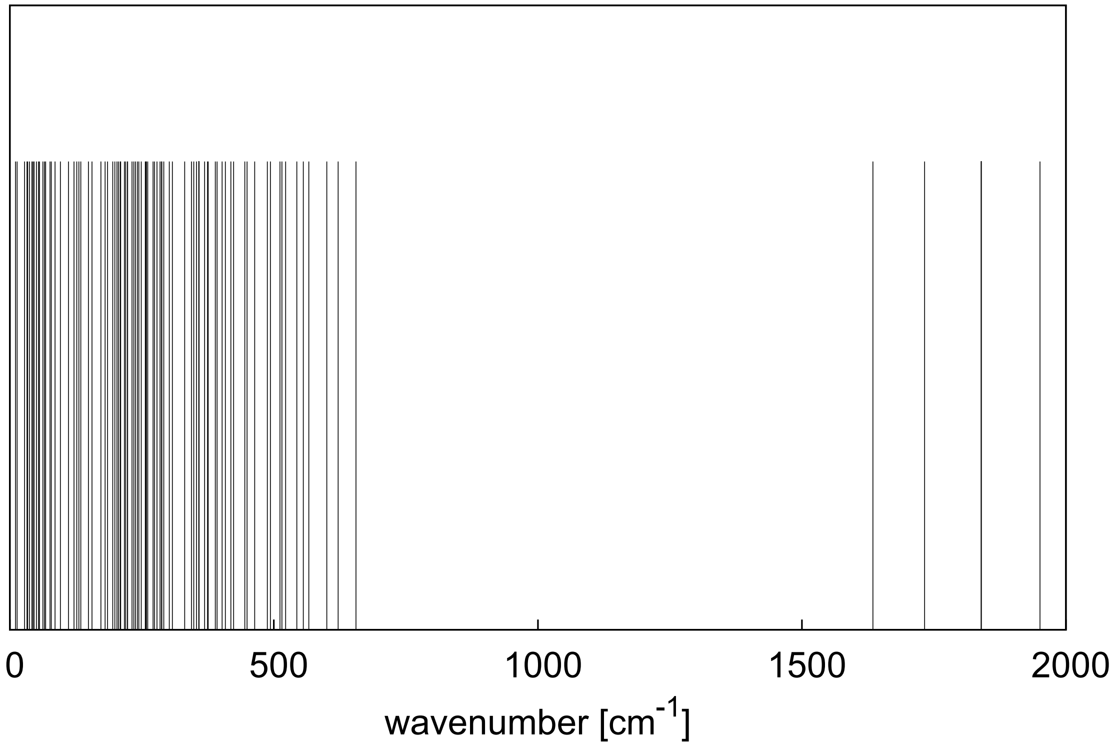


Figure 75: Eigenfrequencies obtained from the extraction of the normal modes of a Nafion molecule, using constant forces for the bonds 10 times larger than in fig. 74.

6.A Appendix: VMD Snapshots

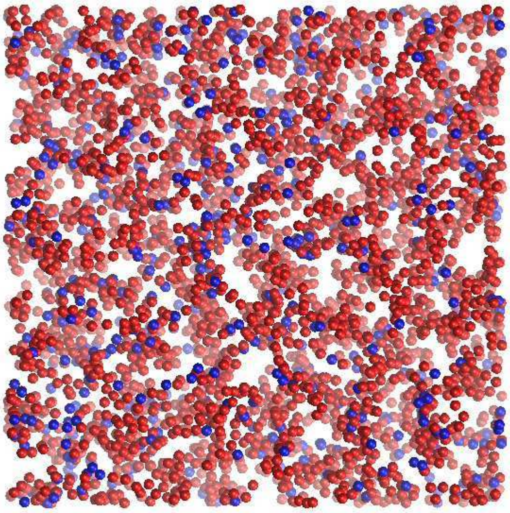


Figure 76: NR-10-1 system at t=20 ns. Only K^+-ions are shown. In red: 4.5 Å closer of at least one S-atom; in blue: not 4.5 Å closer than a S-atom.

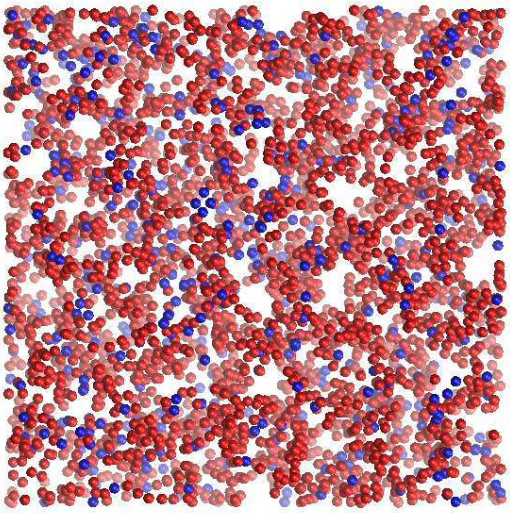


Figure 77: NR-10-1 system at t=30 ns. Only K^+-ions are shown. In red: 4.5 Å closer of at least one S-atom; in blue: not 4.5 Å closer than a S-atom.

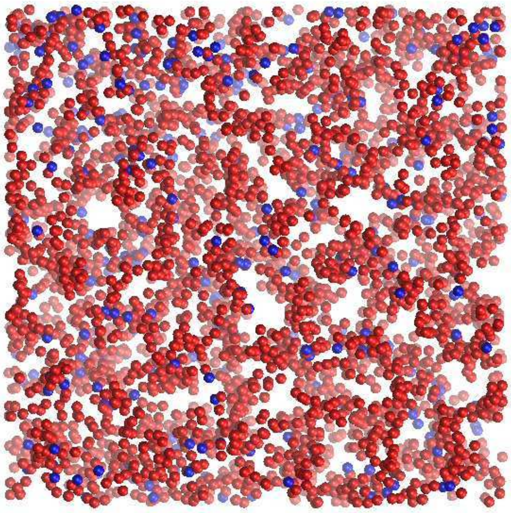


Figure 78: NR-7.5-1 system at t=15 ns. Only K^+-ions are shown. In red: 4.5 Å closer of at least one S-atom; in blue: not 4.5 Å closer than a S-atom.

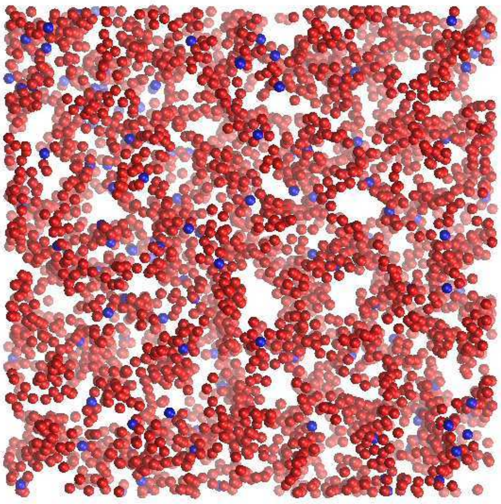


Figure 79: NR-5-1 system at t=15 ns. Only K^+-ions are shown. In red: 4.5 Å closer of at least one S-atom; in blue: not 4.5 Å closer than a S-atom.

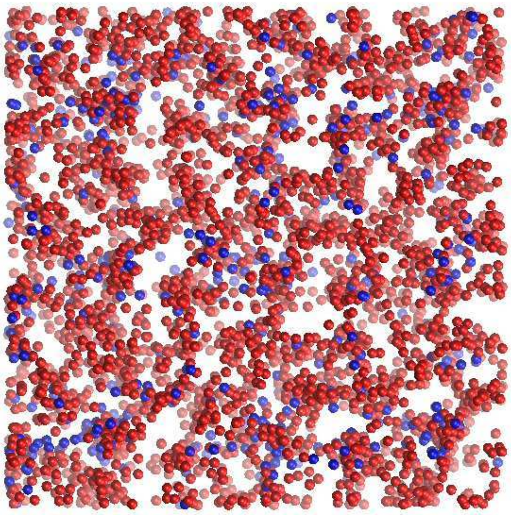


Figure 80: NC-10-1 system at t=15 ns. Only K^+-ions are shown. In red: 4.5 Å closer of at least one S-atom; in blue: not 4.5 Å closer than a S-atom.

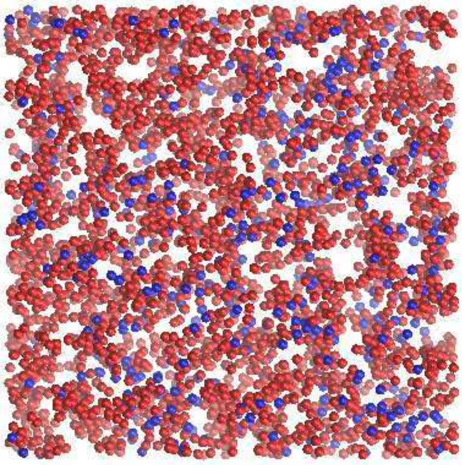


Figure 81: HR-10-1 system at t=20 ns. Only K^+-ions are shown. In red: 4.5 Å closer of at least one S-atom; in blue: not 4.5 Å closer than a S-atom.

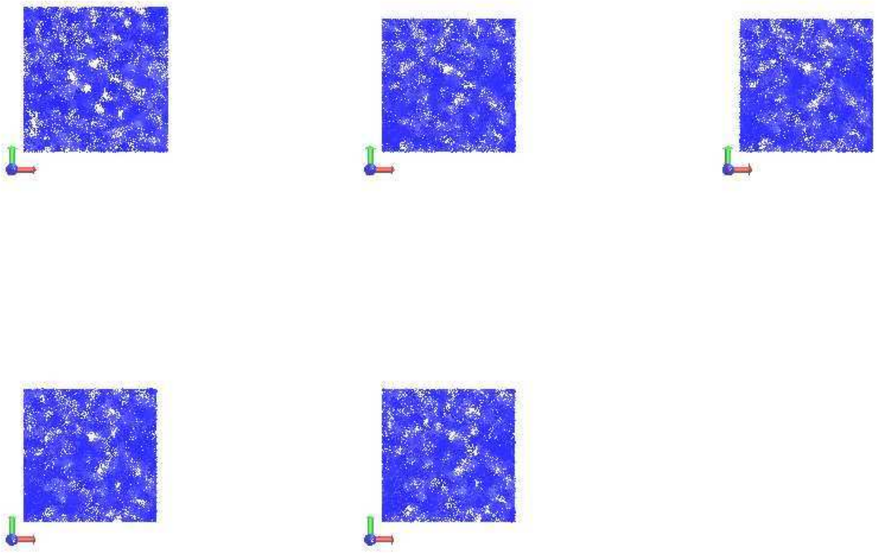


Figure 82: Pictures of the simulation of the NR-10-3 system, taken every 5 ns. Only the Ow-atoms are shown. top left: t=0; top middle: t=5 ns; top right: t=10 ns; middle left: t=15 ns; middle middle: t=20 ns.

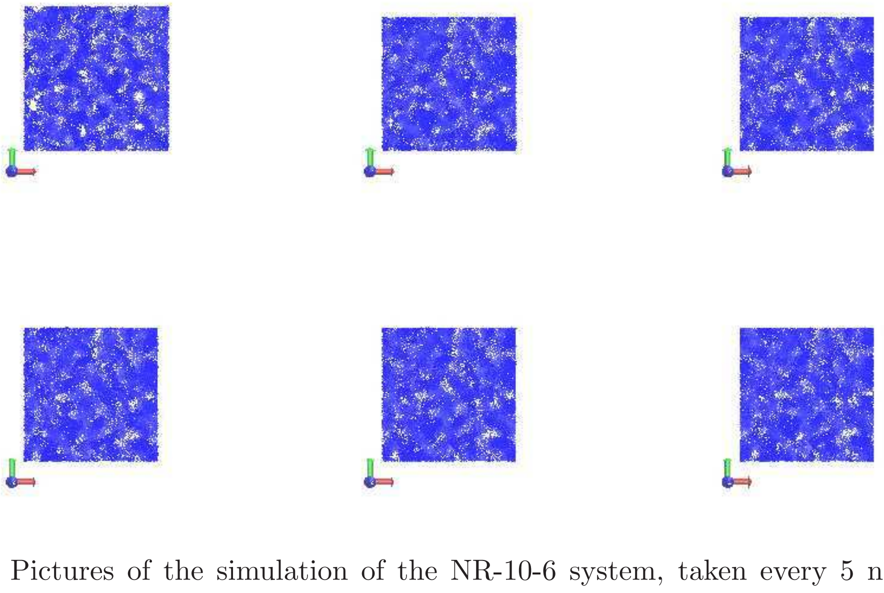


Figure 83: Pictures of the simulation of the NR-10-6 system, taken every 5 ns. Only the Ow-atoms are shown. top left: t=0; top middle: t=5 ns; top right: t=10 ns; middle left: t=15 ns; middle middle: t=20 ns; right middle: t=25 ns.

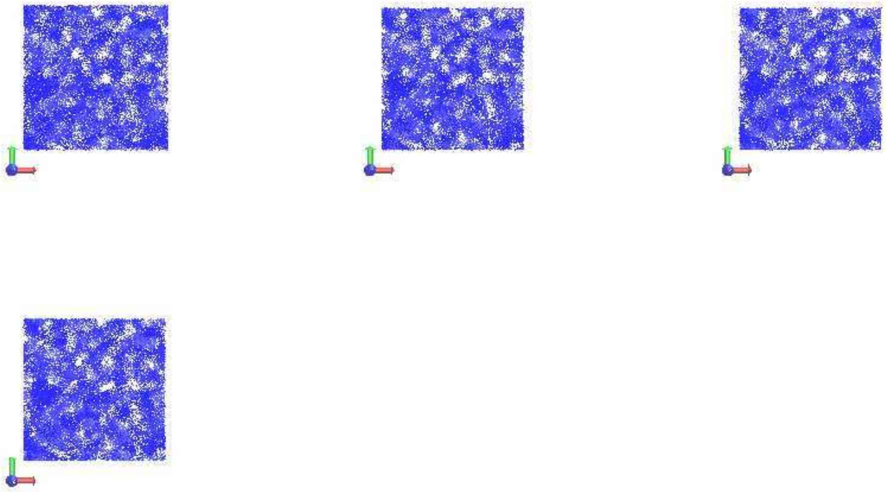


Figure 84: Pictures of the simulation of the NR-7.5-1 system, taken every 5 ns. Only the Ow-atoms are shown. top left: t=0; top middle: t=5 ns; top right: t=10 ns; middle left: t=15 ns.

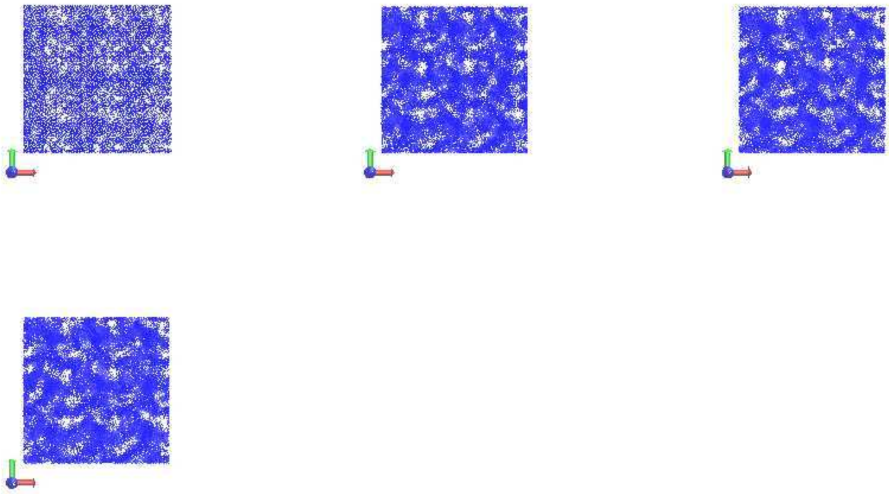


Figure 85: Pictures of the simulation of the NC-10-1 system, taken every 5 ns. Only the Ow-atoms are shown. top left: t=0; top middle: t=5 ns; top right: t=10 ns; middle left: t=15 ns;

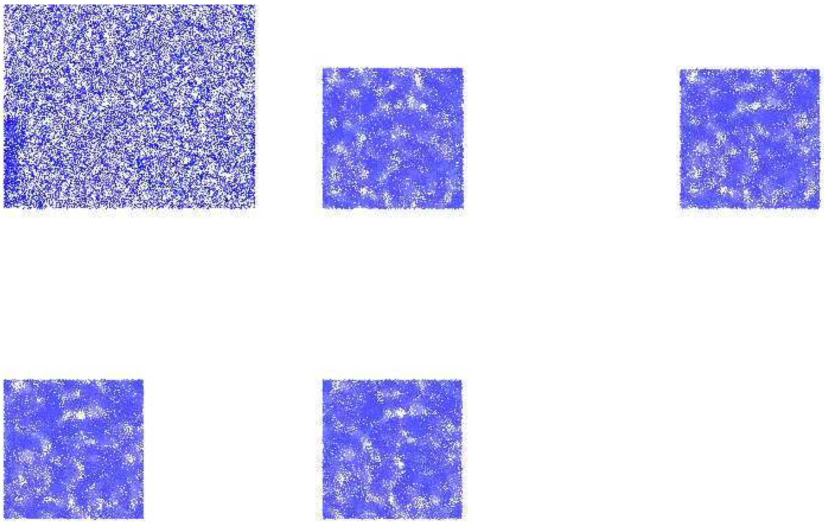


Figure 86: Pictures of the simulation of the HR-10-1 system, taken every 5 ns. Only the Ow-atoms are shown. top left: t=0; top middle: t=5 ns; top right: t=10 ns; middle left: t=15 ns; middle middle: t=20 ns.

6.B Appendix: Radial Distribution Functions

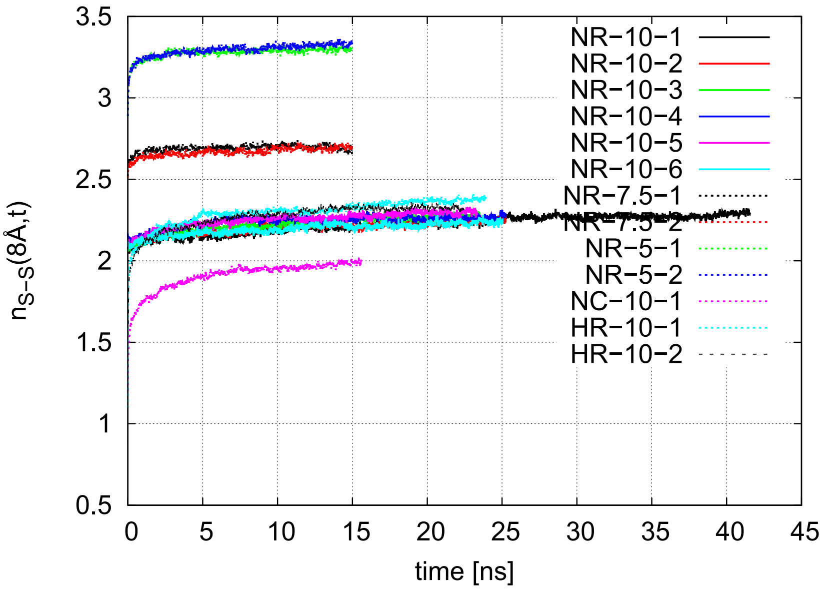


Figure 87: Average number of S-atoms closer than 8 Å of each S-atom vs. the simulation time, for all systems.

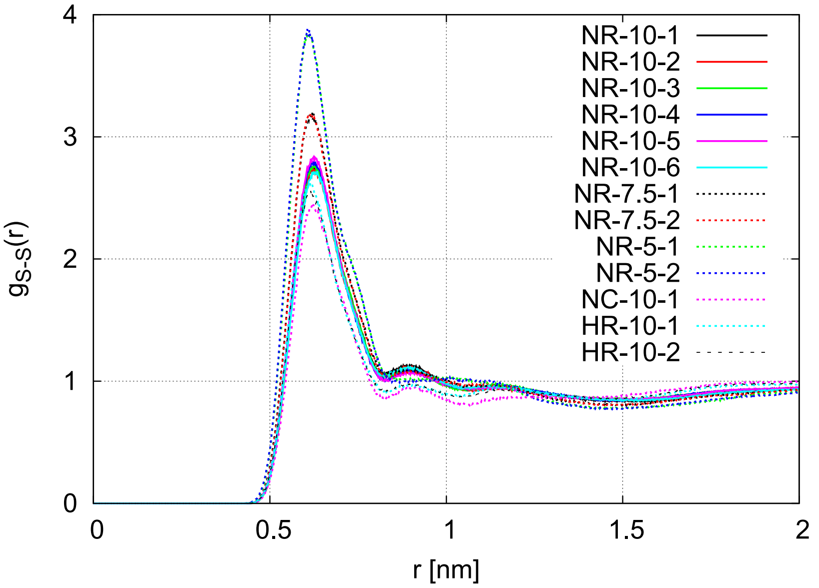


Figure 88: $g_{\mathrm{S-S}}(r)$ over the last 10 ns of the simulation (computed every 100 ps, hence averaged over 100 configurations), for all systems.

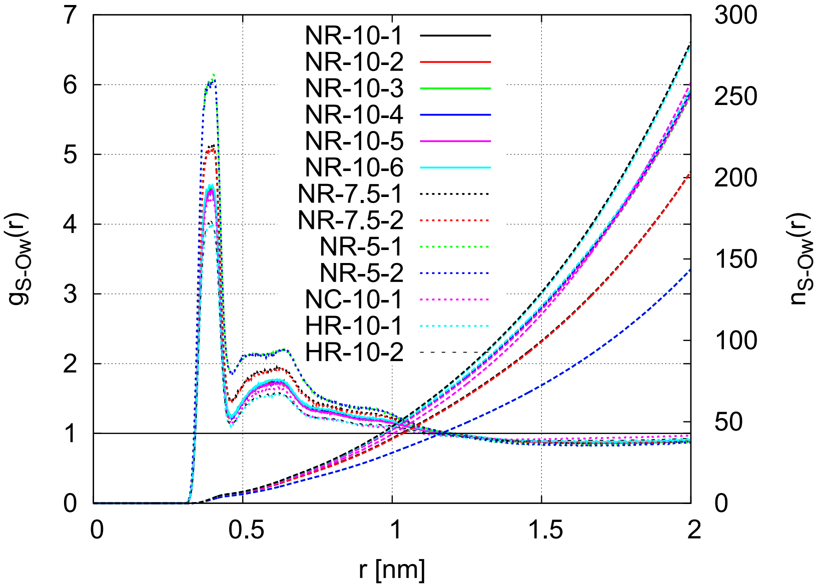


Figure 89: $g_{\mathrm{S-Ow}}(r)$ over the last 10 ns of the simulation (computed every 100 ps, hence averaged over 100 configurations), for all systems.

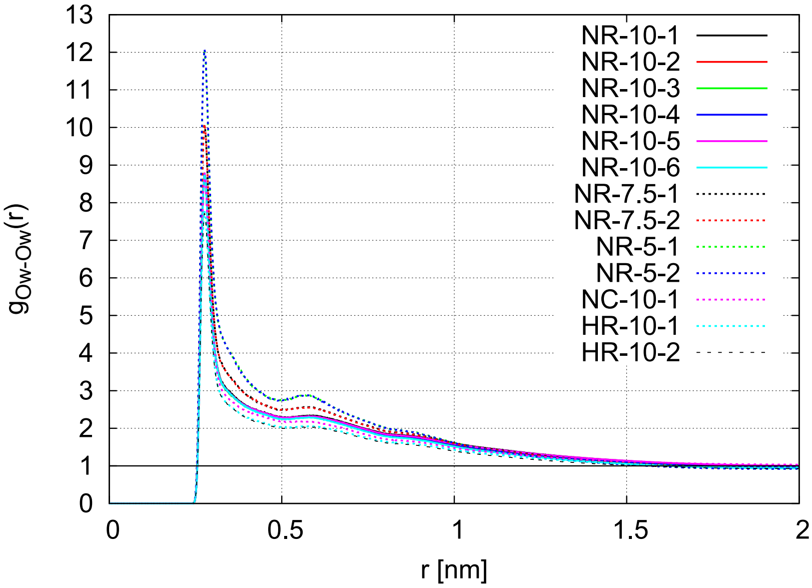


Figure 90: $g_{\mathrm{Ow-Ow}}(r)$ over the last 10 ns of the simulation (computed every 100 ps, hence averaged over 100 configurations), for all systems.

6.C Appendix: Structure Factors

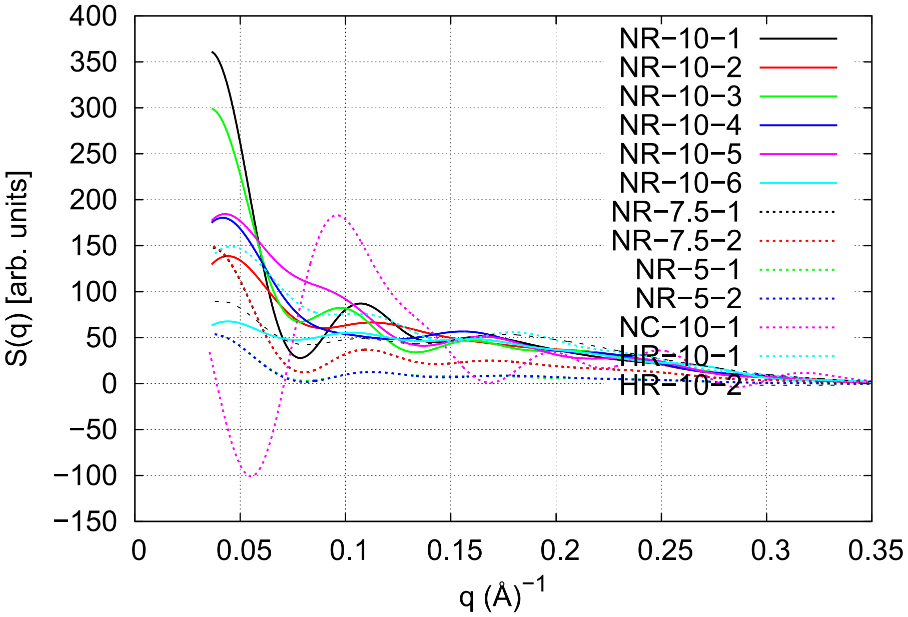


Figure 91: S(q) (neutron), computed for the last configuration, for all systems.

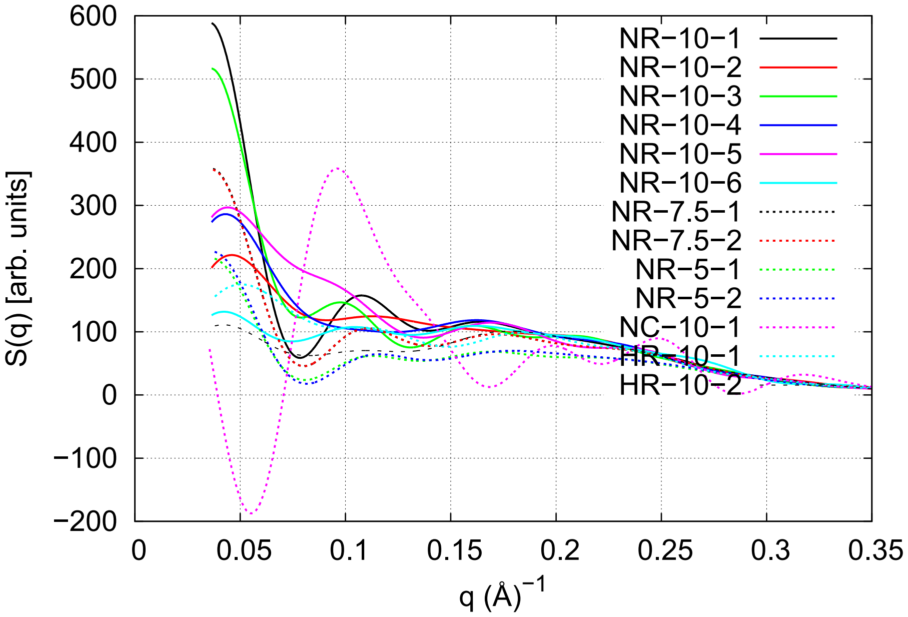


Figure 92: S(q) (X-ray), computed for the last configuration, for all systems.

6.D Appendix: End-to-End Distances

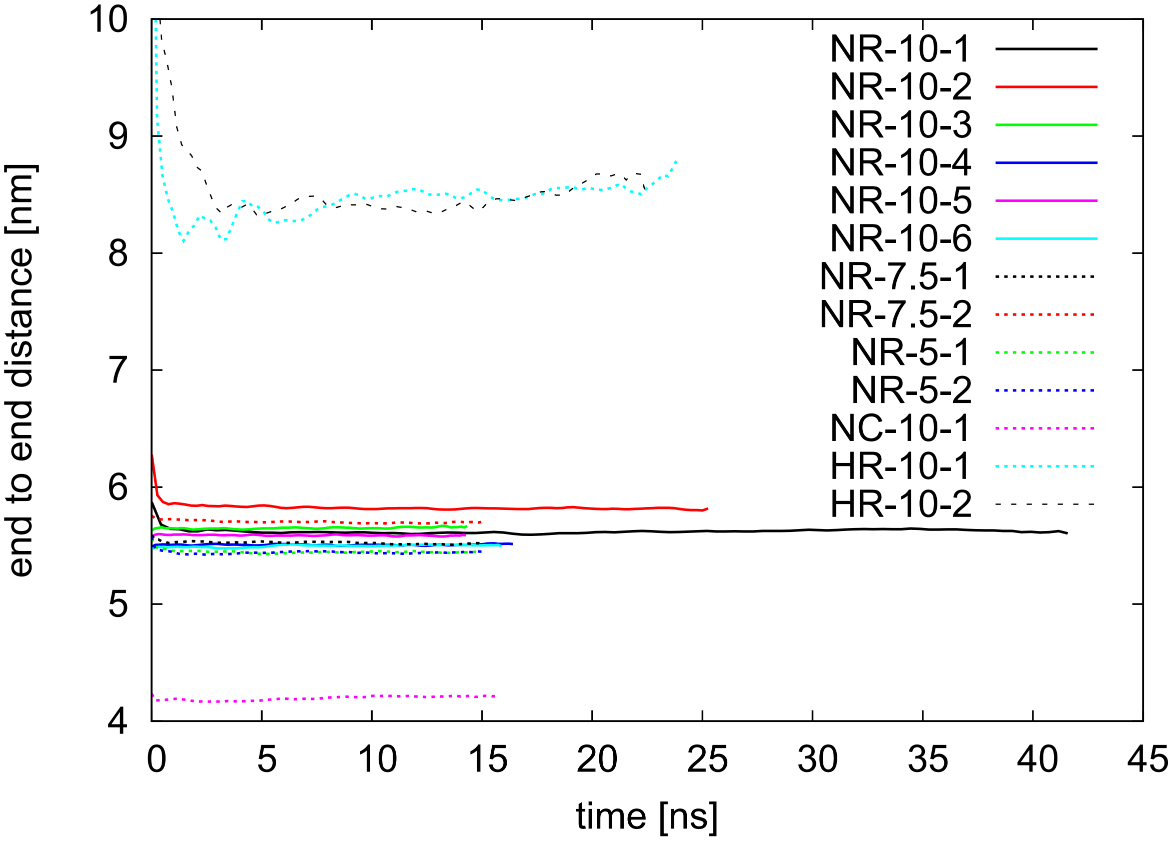


Figure 93: End-to-end distance vs. the simulation time, for all systems. The curves have been smoothed with a Bezier function.

6.E Appendix: Cluster Analysis

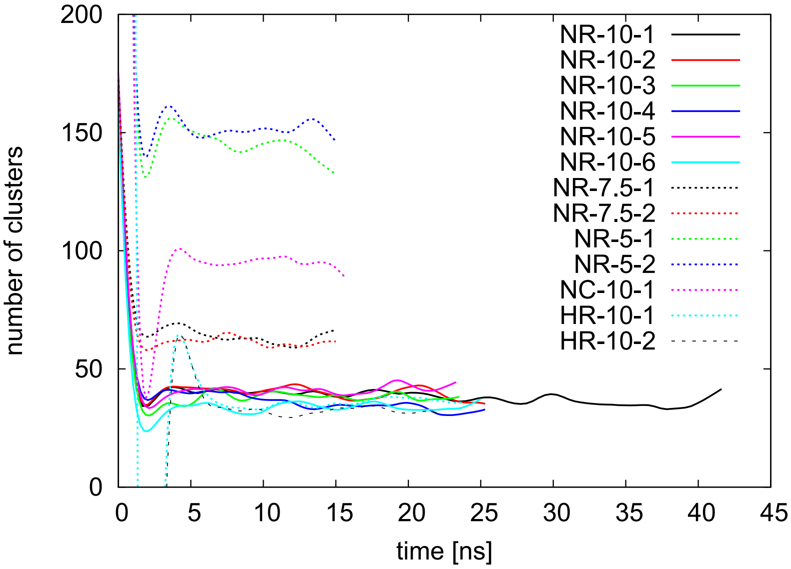


Figure 94: Number of aqueous clusters vs. the simulation time, for all systems. Ow-atoms and K^+-ions are considered only. The distance criterion is 3.5 Å. The curves have been smoothed with a spline function.

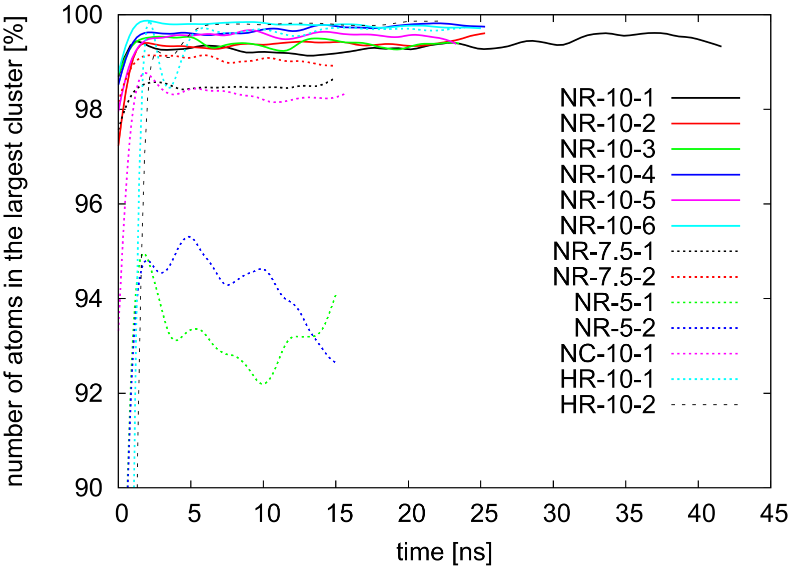


Figure 95: Number of atoms (in % of total) in the largest aqueous cluster vs. the simulation time, for all systems. Ow-atoms and K^+-ions are considered only. The distance criterion is 3.5 Å. The curves have been smoothed with a spline function.

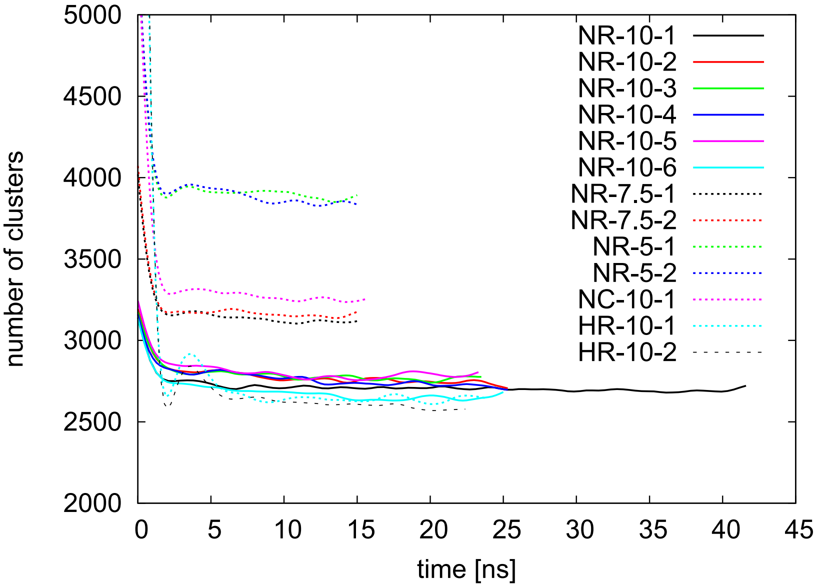


Figure 96: Number of aqueous clusters vs. the simulation time, for all systems. Ow-atoms and K^+-ions are considered only. The distance criterion is 3.0 Å. The curves have been smoothed with a spline function.

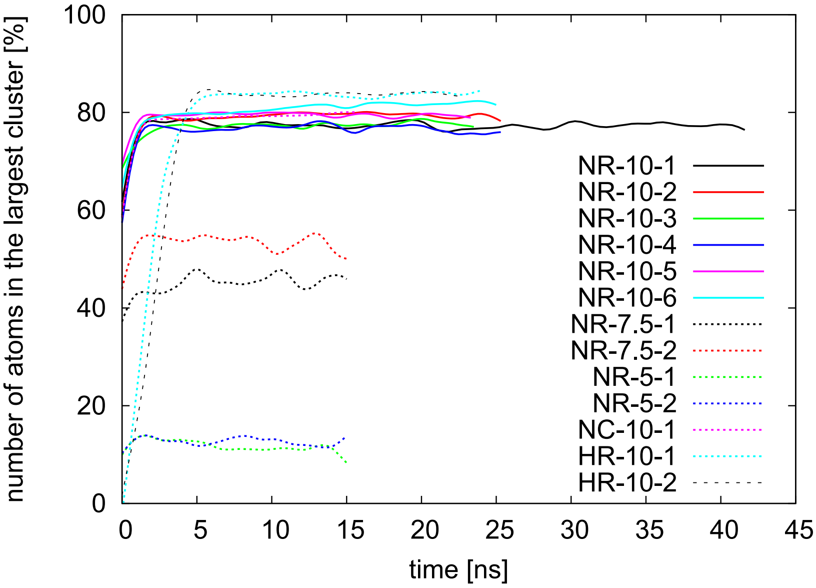


Figure 97: Number of atoms (in % of total) in the largest aqueous cluster vs. the simulation time, for all systems. Ow-atoms and K^+-ions are considered only. The distance criterion is 3.0 Å. The curves have been smoothed with a spline function.

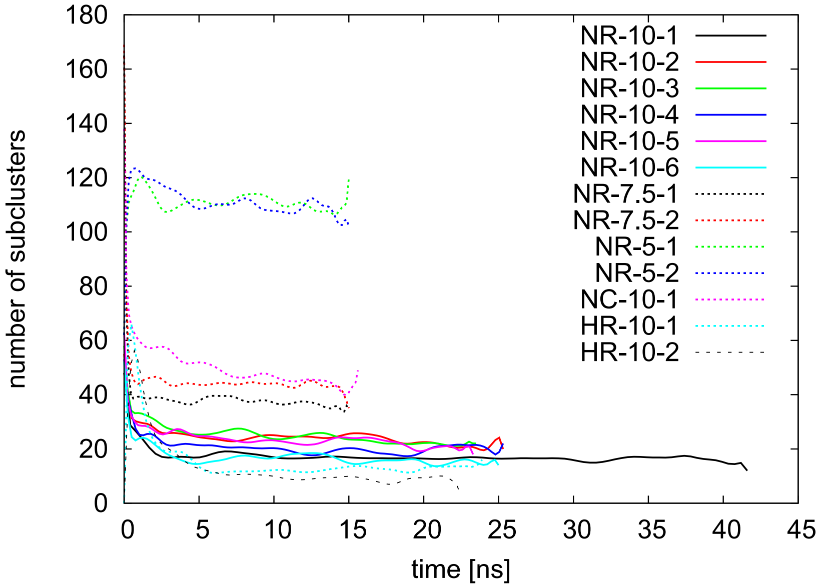


Figure 98: Number of subclusters, after iterative removal of atoms with up to two neighbors, vs. the simulation time, for all systems. Only Ow-atoms and K$^+$-ions are considered. The distance criterion is 3.5 Å. The curves have been smoothed with a Bezier function.

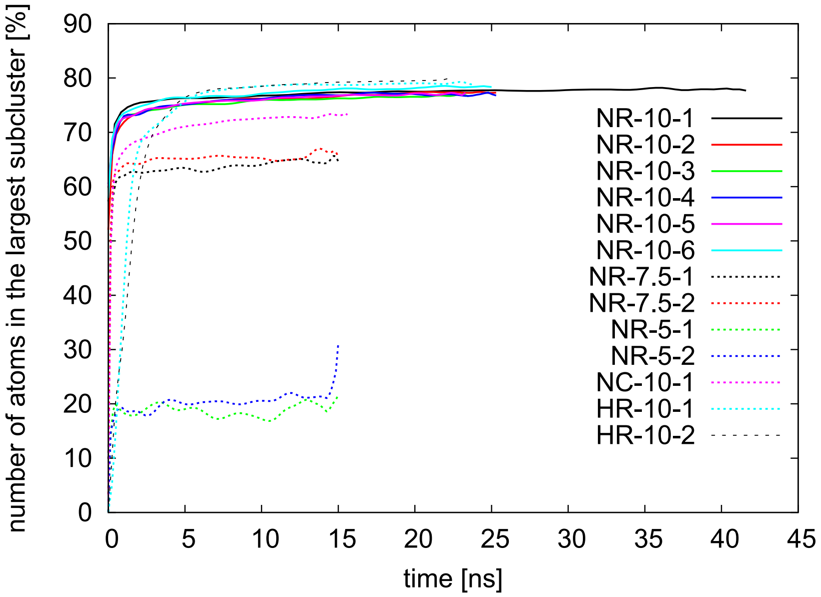


Figure 99: Number of atoms (in % of total) in the largest subcluster, after iterative removal of atoms with up to two neighbors, vs. the simulation time, for all systems. Only Ow-atoms and K$^+$-ions are considered. The distance criterion is 3.5 Å. The curves have been smoothed with a Bezier function.

6.F Appendix: Mean Square Displacements

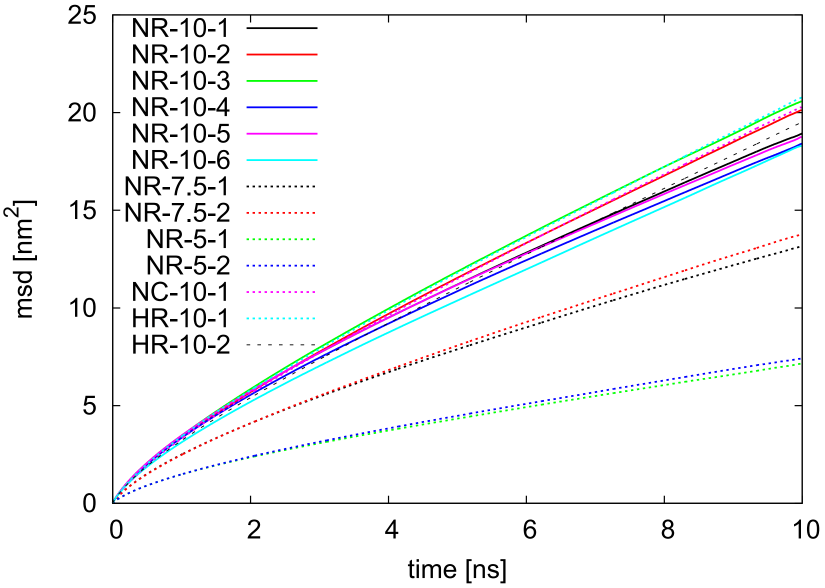


Figure 100: Mean square displacement of the water molecule oxygens (Ow) over the last 10 ns of the simulation (computed every 10 ps, hence averaged over 1000 configurations), for all systems.

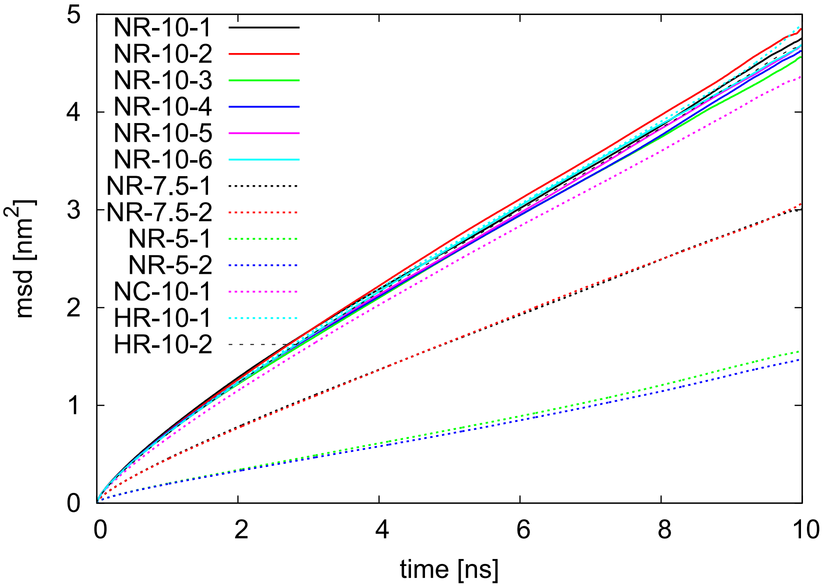


Figure 101: Mean square displacement of the K$^+$-ions over the last 10 ns of the simulation (computed every 10 ps, hence averaged over 1000 configurations), for all systems.

6.G Appendix: Vibrational Analysis

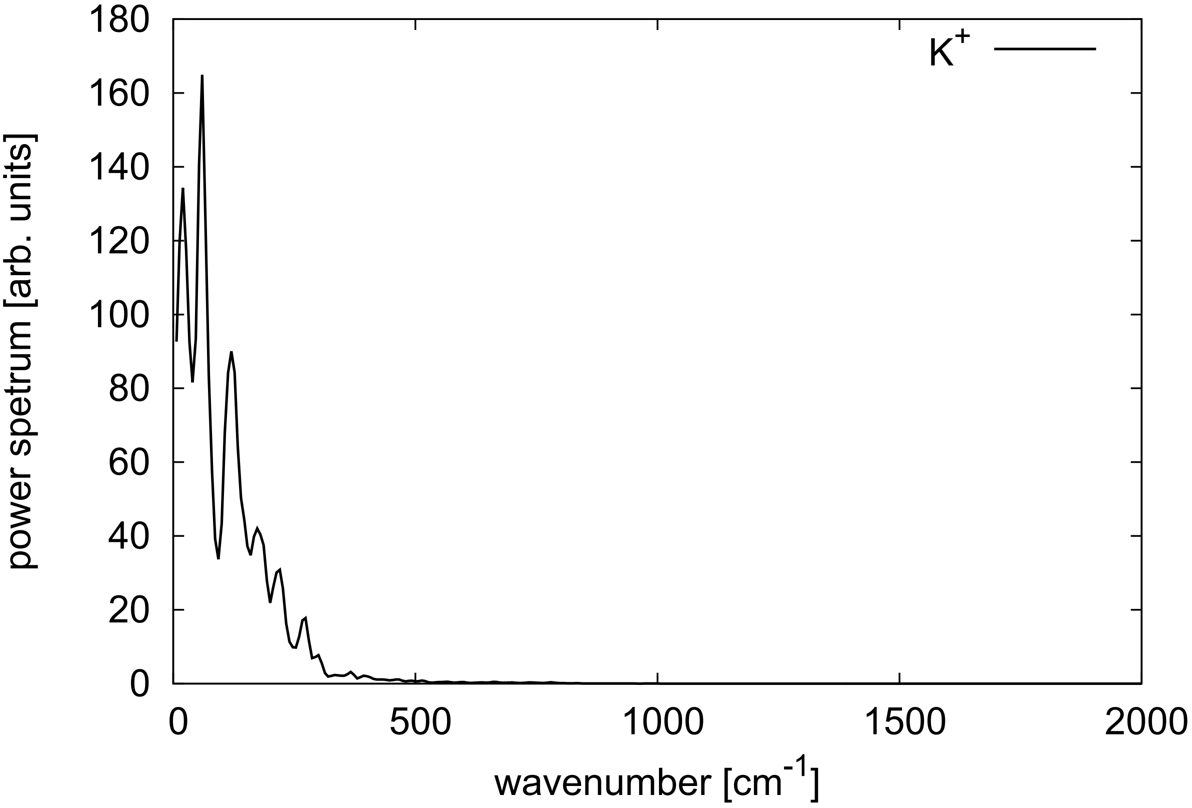


Figure 102: Power spectrum (in arbitrary units) of the K$^+$-ions in a small Nafion/Water system.

7 Conclusions and Outlook

This thesis aims to study Nafion / Water systems at the molecular level. The tool is Molecular Dynamics (MD) computer simulations with a subsequent analysis with established and new statistical methods. The main difficulties encountered are rooted in the very different time scales involved in the dynamics of the systems. We have attempted to overcome them by a particular, yet unbiased, choice of initial conditions. Among the main conclusions are the following assertions:

- MD simulations are presently not able to fully sample the configurational space of large scale Nafion systems. With the present version of the algorithm designed to generate random Nafion chains, several sets of configurations that are compatible with available scattering data can be generated and equilibrated.
- The force field and force field parameters used here for the Nafion systems reproduce satisfactorily the frequency range of the vibrational and librational modes expected for a system composed of Nafion (or Hyflon), water molecules, and K^+-ions.
- In contrast to Voth and Knox [2], large scale MD simulations of random morphological models of Nafion soaked in water are shown to yield, after suitable equilibration, structures in agreement with available experimental data. Several examples are shown.
- Both the cylindrical and the random morphological models of Nafion reproduce the experimental scattering data satisfactorily, confirming here Voth and Knox [2]'s opinion that scattering data alone are not sufficient to elucidate the structure of a system as complex as Nafion in water. Hence the ongoing debate over the morphology of Nafion.
- Based on the radial distribution functions, we find that water in Nafion is not predominantly bulk-like, even at high hydration levels, in contrast to Lee et al. [76].
- The aqueous domains show characteristic differences compared with pure liquid water at the same temperature. This is in keeping with an inhomogeneity in the spatial distribution of the water molecules at the nanometer scale.
- The size of and the connectivity between the aqueous domains, as well as the mean S-S distance increase when increasing the water content. This is the well-known swelling behavior of the membrane.
- There is a dynamical equilibrium for the distribution of the water molecules into a very large dominant cluster and smaller neighboring ones. This in keeping with

the formation and breaking of temporary bridges between aqueous clusters, as predicted by the small-scale Nafion simulation by Vishnyakov and Neimark [32].

- The so-called ionomer peak experimentally observed from scattering experiments is due to a periodicity involving mainly the water molecules in the aqueous domains and the matrix of CF_2-backbones.
- All (or almost all) K^+-ions are located in the aqueous domains and the S-atoms are located either near the polymer/water interface or in the aqueous domains.
- The S-atoms and the K^+-ions form some sort of 'intraclusters' inside the aqueous clusters.
- Like Voth and Knox [2] we find that the sulfonate groups play a bridging role between the aqueous domains.
- All (or almost all) the K^+-ions in the aqueous domains are associated with a SO_3^--group.
- In contrast to Dupuis et al. [18] we do not find a clear percolation threshold, but rather a continuous change when varying the water content λ.
- In agreement with Brandell et al. [58] it is found that Hyflon, due to its shorter side chain, promotes the formation of slightly larger S-clusters than Nafion.
- Except for the S-clusters, very similar results for the structure and dynamics are obtained for Hyflon and Nafion materials. Experimentally, the major difference between Hyflon and Nafion remains the higher glass-transition temperature in Hyflon.
- In contrast to Brandell et al. [58], the self-diffusion coefficients of the water molecules and cations are found to be independent of the length of the polymer side chains (Nafion/Hyflon).

This work has shown the power and limitations of molecular simulations of complex systems like the ionomer membranes used in industrial applications. A wealth of data related to local and some global properties with relaxation times of the order of nanoseconds or less can be obtained. However, phenomena with longer characteristic times remain mostly elusive. With some ingeniousness, these shortcomings can be alleviated: we propose here to sample the configurational space by averaging over several simulations stemming from different, but equivalent, starting conditions.

Future theoretical work on such systems must thus use combinations of methods. The difficulty here is to find consistent coarse-graining schemes. The restricted sizes of the systems that can be treated by simulations will remain a problem for the foreseeable future. More accurate experimental data, in particular on the morphology, is also very desirable. Much work remains thus to be done until the structure (and the dynamics) of hydrated polyelectrolyte membranes is fully understood at all length- and time-scales of interest.

8 Bibliography

References

[1] K. A. Mauritz and R. B. Moore. State of Understanding of Nafion. *Chem. Rev.*, 104:4535–4585, 2004.

[2] C. K. Knox and G. A. Voth. Probing Selected Morphological Models of Hydrated Nafion Using Large-Scale Molecular Dynamics Simulations. *J. Phys. Chem. B*, 114:3205–3218, 2010.

[3] K. Schmidt-Rohr and Q. Chen. Parallel cylindrical water nanochannels in Nafion fuel-cell membranes. *Nature Materials*, 7:75–83, 2008.

[4] T. D. Gierke, G. E. Munn, and F. C. Wilson. The Morphology in Nafion Perfluorinated Membrane Products, as Determined by Wide- and Small-Angle X-ray Studies. *Journal of Polymer Science: Polymer Physics Edition*, 19:1687–1704, 1981.

[5] L. Rubatat, A.-L. Rollet, G. Gebel, and O. Diat. Evidence of Elongated Polymeric Aggregate in Nafion. *Macromolecules*, 35:4050–4055, 2002.

[6] C. F. Schoenbein. On the voltaic polarization of certain solid and fluid substances. *The London and Edinburgh Philosophical Magazine and Journal of Science*, 14:43–45, 1839.

[7] W. R. Grove. On voltaic series and the combination of gases by platinum. *The London and Edinburgh Philosophical Magazine and Journal of Science*, 14:127–130, 1839.

[8] W. R. Grove. On a gaseous voltaic battery. *The London and Edinburgh Philosophical Magazine and Journal of Science*, 21:417–420, 1839.

[9] `http://www.ingsman.com/dmfc.htm`.

[10] Gregor Hooger. *Fuel Cell Technology Handbook*. 2003.

[11] `http://en.wikipedia.org/wiki/Fuel_cell_vehicle`.

[12] `http://www.gminsidenews.com/forums/f12/who-says-gm-late-hybrid-game-26366/index3.html`.

[13] K. D. Kreuer, S. J. Paddison, E. Spohr, and M. Schuster. Transport in Proton Conductors for Fuel-Cell Applications: Simulations, Elementary Reactions, and Phenomenology. *Chem. Rev.*, 104:4637–4678, 2004.

[14] L. Carrette, K. A. Friedrich, and U. Stimming. Fuel cells – fundamentals and applications. *Fuel Cells*, 1:5–39, 2001.

[15] `http://www1.eere.energy.gov/hydrogenandfuelcells/fuelcells/`.

[16] H. A. Gasteiger W. Vielstich, A. Lamm. *Handbook of Fuel Cells: Fundamentals, Technology and Applications*. J. Wiley & sons, New York, 2003.

[17] `http://www.strath.ac.uk/`.

[18] R. Devanathan, A. Venkatnathan, R. Rousseau, M. Dupuis, T. Frigato, W. Gu, and V. Helms. Atomistic Simulation of Water Percolation and Proton Hopping in Nafion Fuel Cell Membrane. *J. Phys. Chem. B*, 114:13681–13690, 2010.

[19] D. Seeliger, C. Hartnig, and E. Spohr. Aqueous pore structure and proton dynamics in solvated nafion membranes. *Electrochim. Acta*, 50:4234–4240, 2005.

[20] M. K. Petersen and G. A. Voth. Characterization of the solvation and transport of the hydrated proton in the perfluorosulfonic acid membrane nafion. *J. Phys. Chem. B*, 110:18594–18600, 2006.

[21] M. A. Ilhan and E. Spohr. Ab *initio* molecular dynamics of proton networks in narrow polymer electrolyte pores. *J. Phys.: Condens. Matter*, 23:234104–234112, 2011.

[22] M. A. Ilhan and E. Spohr. Hydrogen Bonding in Narrow Protonated Polymer Electrolyte Pores. *J. Electroanal. Chem.*, 660:347–351, 2011.

[23] C. J. T. de Grotthuss. Sur la décomposition de l'eau et des corps qu'elle tient en dissolution à l'aide de l'électricité galvanique. *Ann. Chim. (Paris)*, 58:54–74, 1806.

[24] N. Agmon. The Grotthuss Mechanism. *Chem. Phys. Lett.*, 244:456–462, 1995.

[25] H. Lapid, N. Agmon, M. K. Petersen, and G. A. Voth. A bond-order analysis of the mechanism for hydrated proton mobility in liquid water. *J. Chem. Phys.*, 122:14506–14516, 2005.

[26] `http://www.ifc.uni-osnabrueck.de/organische-chemie/organic-materials-chemistry-and-bio-organic-chemistry/organische-materialchemie-prof-dr-uwe-beginn/prof-dr-uwe-beginn-research/functionale-nanoporose-membranen`.

[27] R. Devanathan, A. Venkatnathan, and M. Dupuis. Atomistic Simulation of Nafion Membrane: 1. Effect of Hydration on Membrane Nanostructure. *J. Phys. Chem. B*, 111:8069–8079, 2007.

[28] R. Devanathan, A. Venkatnathan, and M. Dupuis. Atomistic Simulation of Nafion Membrane: 2. Dynamics of Water Molecules and Hydronium Ions. *J. Phys. Chem. B*, 111:13006–13013, 2007.

[29] N. P. Blake, M. K. Petersen, G. A. Voth, and H. Metiu. Structure of Hydrated Na-Nafion Polymer Membranes. *J. Phys. Chem. B*, 109:24244–24253, 2005.

[30] M. K. Petersen, F. Wang, N. P. Blake, H. Metiu, and G. A. Voth. Excess Proton Solvation and Delocalization in a Hydrophilic Pocket of the Proton Conducting Polymer Membrane Nafion. *J. Phys. Chem. B*, 109:3727–3730, 2005.

[31] M. K. Petersen, A. J. Hatt, and G. A. Voth. Orientational Dynamics of Water in the Nafion Polymer Electrolyte Membrane and its Relationship to Proton Transport. *J. Phys. Chem. B*, 112:7754–7761, 2008.

[32] A. Vishynakov and A. V. Neimark. Molecular Dynamics Simulation of Microstructure and Molecular Mobilities in Swollen Nafion Membranes. *J. Phys. Chem. B*, 105:9586–9594, 2001.

[33] A. Vishnyakov and A. V. Neimark. Molecular Simulation Study of Nafion Membrane Solvation in Water and Methanol. *J. Phys. Chem. B*, 104:4471–4478, 2000.

[34] A. Vishynakov and A. V. Neimark. Molecular Dynamics Simulation of Nafion Oligomer Solvation in Equimolar Methanol-Water Mixture. *J. Phys. Chem. B*, 105:7830–7834, 2001.

[35] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *J. Chem. Phys.*, 21:1087–1092, 1953.

[36] B. Alder and T. E. Wainwright. Studies in molecular dynamics. I. general method. *J. Chem. Phys.*, 31:459–466, 1959.

[37] W. W. Wood and F. R. Parker. Monte Carlo Equation of State of Molecules Interacting with the Lennard-Jones Potential. I. A Supercritical Isotherm at about Twice the Critical Temperature. *J. Chem. Phys.*, 27:720–733, 1957.

[38] A. Rahman. Correlations in the motion of atoms in liquid-argon. *Phys. Rev.*, 136:405–411, 1964.

[39] F. H. Stillinger and A. Rahman. Improved simulation of liquid water by molecular dynamics. *J. Chem. Phys.*, 60:1545–1557, 1974.

[40] J. A. McCammon, B. R. Gelin, and M. Karplus. Dynamics of folded proteins. *Nature*, 267:585–590, 1977.

[41] P. A. Bopp. *Molecular Dynamics Computer Simulations of the Single Molecule Dynamics in Hydrogen Bonded Liquids*. Darmstadt University, 1987. Habilitation Thesis.

[42] D. E. Curtin, R. D. Lousenberg, T. J. Henry, P. Tangeman, and M. Tisack. Advanced materials for improved PEMFC performance and life. *J. Power Sources*, 131:41–48, 2004.

[43] C. Heitner-Wirguin. Recent advances in perfluorinated ionomer membranes: structure, properties and applications. *J. Membrane Sci.*, 120:1–33, 1996.

[44] D. Wu, S. J. Paddison, J. A. Elliott, and S. J. Hamrock. Mesoscale Modeling of Hydrated Morphologies of 3M Perfluorosulfonic Acid-Based Fuel Cell Electrolytes. *Langmuir*, 26:14308–14315, 2010.

[45] M. Saito, N. Arimura, K. Hayamizu, and T. Okada. Mechanisms of Ion and Water Transport in Perfluorosulfonated Ionomer Membranes for Fuel Cells. *J. Phys. Chem. B*, 108:16064–16070, 2004.

[46] B. R. Ezzell, W. P. Carl, and W. A. Mod. Novel polymers having acid functionality. *US Patent*, page 4330654, 1982.

[47] B. R. Ezzell, W. P. Carl, and W. A. Mod. Sulfonic acid electrolytic cell having flourinated polymer membrane with hydration product less than 22000. *US Patent*, page 4358545, 1982.

[48] B. R. Ezzell, W. P. Carl, and W. A. Mod. Sulfonic acid electrolytic cell membranes. *US Patent*, page 4417969, 1983.

[49] B. R. Ezzell, W. P. Carl, and W. A. Mod. Electrolytic cell having an improved ion exchange membrane and process for operating. *US Patent*, page 4470889, 1984.

[50] B. R. Ezzell, W. P. Carl, and W. A. Mod. Sulfonic acid electrolytic cell membranes and use thereof in the electrolysis of sodium chloride. *US Patent*, page 4478695, 1984.

[51] B. R. Ezzell, W. P. Carl, and W. A. Mod. Process to produce novel fluorocarbon vinyl ethers and resulting polymers. *US Patent*, page 4834922, 1989.

[52] B. R. Ezzell, W. P. Carl, and W. A. Mod. Preparation of vinyl ethers. *US Patent*, page 4358412, 1982.

[53] V. Arcella, A. Ghielmi, and G. Tommasi. High performance perfluoropolymer films and membranes. *Ann. N. Y. Acad. Sci.*, 984:226–244, 2003.

[54] A. Ghielmi, P. Vaccarono, C. Troglia, and V. Arcella. Proton exchange membranes based on the short-side-chain perfluorinated ionomer. *J. Power Sources*, 145:108–115, 2005.

[55] V. Arcella and C. Troglia A. Ghielmi. Hyflon Ion Membranes for Fuel Cells. *Ind. Eng. Chem. Res.*, 44:7646–7651, 2005.

[56] A. S. Arico, V. Baglio, A. Di Blasi, V. Antonucci, L. Cirillo, A. Ghielmi, and V. Arcella. Proton exchange membranes based on the short-side-chain perfluorinated ionomer for high temperature direct methanol fuel cells. *Desalination*, 199:271–273, 2006.

[57] K. D. Kreuer, M. Schuster, B. Obliers, O. Diat, U. Traub, A. Fuchs, U. Klock, S. J. Paddison, and J. Maier. Short-side-chain proton conducting perfluorosulfonic acid ionomers: Why they perform better in PEM fuel cells. *J. Power Sources*, 178:499–509, 2008.

[58] J. Karo, A. Aabloo, J. O. Thomas, and D. Brandell. Molecular Dynamics Modeling of Proton Transport in Nafion and Hyflon Nanostructures. *J. Phys. Chem. B*, 114:6056–6064, 2010.

[59] S. J. Hamrock and M. A. Yandrasits. Proton Exchange Membranes for Fuel Cell Applications. *Journal of Macromolecular Science, Part C: Polymer Reviews*, 46:219–244, 2006.

[60] G. M. Haugen, F. Meng, N. V. Aieta, J. L. Horan, M. Kuo, M. H. Frey, S. J. Hamrock, and A. M. Herring. The Effect of Heteropoly Acids on Stability of PFSA PEMs under Fuel Cell Operation. *Electrochemical and Solid State Letters*, 10:51–55, 2007.

[61] N. V. Aieta, R. J. Stanis, J. L. Horan, M. A. Yandrasits, D. J. Cookson, B. Ingham, M. F. Toney, S. J. Hamrock, and A. M. Herring. Clipped Random Wave Morphologies and the Analysis of the SAXS of an Ionomer Formed by Copolymerization of Tetrafluoroethylene and $CF_2 = CFO(CF_2)_4SO_3H$. *Macromolecules*, 42:5774–5780, 2009.

[62] P. Térech. *Molecular Gels: Materials with Self-Assembled Fibrillar Networks*, chapter 8. Springer, 2005.

[63] E. J. Roche, M. Pineri, R. Duplessix, and A. M. Levelut. Small-Angle Scattering Studies of Nafion Membranes. *Journal of Polymer Science: Polymer Physics Edition*, 19:1–11, 1981.

[64] H. L. Yeager and A. Steck. Cation and Water Diffusion in Nafion Ion Exchange Membranes: Influence of Polymer Structure. *J. Electrochem. Soc.*, 128:1880–1884, 1981.

[65] E. J. Roche, M. Pineri, and R. Duplessix. Phase Separation in Perfluorosulfonate Ionomer Membranes. *Journal of Polymer Science: Polymer Physics Edition*, 20:107–116, 1982.

[66] M. Fujimura, T. Hashimoto, and H. Kawai. Small-Angle X-ray Scattering Study of Perfluorinated Ionomer Membranes. 1. Origin of Two Scattering Maxima. *Macromolecules*, 14:1309–1315, 1981.

[67] M. Fujimura, T. Hashimoto, and H. Kawai. Small-Angle X-ray Scattering Study of Perfluorinated Ionomer Membranes. 2. Models for Ionic Scattering Maximum. *Macromolecules*, 15:136–144, 1982.

[68] B. Dreyfus, G. Gebel, P. Aldebert, M. Pineri, M. Escoubes, and M. Thomas. Distribution of the "micelles" in hydrated perfluorinated ionomer membranes from SANS experiments. *J. Phys. France*, 51:1341–1354, 1990.

[69] G. Gebel and J. Lambard. Small-Angle Scattering Study of Water-Swollen Perfluorinated Ionomer Membranes. *Macromolecules*, 30:7914–7920, 1997.

[70] B. Loppinet and G. Gebel. Small-Angle Scattering Study of Perfluorosulfonated Ionomer Solutions. *J. Phys. Chem. B*, 101:1884–1892, 1997.

[71] B. Loppinet and G. Gebel. Rodlike Colloidal Structure of Short Pendant Chain Perfluorinated Ionomer Solutions. *Langmuir*, 14:1977–1983, 1998.

[72] G. Gebel and R. B. Moore. Small-Angle Scattering Study of Short Pendant Chain Perfluorosulfonated Ionomer Membranes. *Macromolecules*, 33:4850–4855, 2000.

[73] M. H. Litt. A reevaluation of Nafion morphology. *Polym. Prepr.*, 38:80, 1997.

[74] S. Kumar and M. Pineri. Interpretation of small-angle x-ray and neutron scattering data for perfluorosulfonated ionomer membranes. *Journal of Polymer Science Part B: Polymer Physics*, 24:1762–1782, 1986.

[75] R. B. Moore III and C. R. Martin. Morphology and Chemical Properties of the Dow Perfluorosulfonate Ionomers. *Macromolecules*, 22:3594–3599, 1989.

[76] E. M. Lee, R. K. Thomas, A. N. Burgess, D. J. Barnes, A. K. Soper, and A. R. Rennie. Local and Long-Range Structure of Water in a Perfluorinated Ionomer Membrane. *Macromolecules*, 25:3106–3109, 1992.

[77] D. S. Manley, D. L. Williamson, R. D. Noble, and C. A. Koval. Morphological Changes and Facilitated Transport Characteristics for Nafion Membranes of Various Equivalent Weights. *Chem. Mater.*, 8:2595–2600, 1996.

[78] S. K. Young, S. F. Trevino, and N. C. Beck Tan. Small-Angle Neutron Scattering Investigation of Structural Changes in Nafion Membranes Induced by Swelling with Various Solvents. *Journal of Polymer Science: Part B: Polymer Physics*, 40:387–400, 2002.

[79] J. A. Elliott and S. Hanna. Interpretation of the Small-Angle X-ray Scattering from Swollen and Oriented Perfluorinated Ionomer Membranes. *Macromolecules*, 33:4161–4171, 2000.

[80] R. Paul and S. J. Paddison. Effects of dielectric saturation and ionic screening on the proton self-diffusion coefficients in perfluorosulfonic acid membranes. *J. Chem. Phys.*, 123:224704–224717, 2005.

[81] S. Ochi, O. Kamishima, J. Mizusaki, and J. Kawamura. Investigation of proton diffusion in Nafion 117 membrane by electrical conductivity and NMR. *Solid State Ionics*, 180:580–584, 2009.

[82] I. Webman, J. Jortner, and M. H. Cohen. Numerical simulation of continuous percolation conductivity. *Phys. Rev. B*, 14:4737–4740, 1976.

[83] H. W. Starkweather Jr. Cristallinity in perfluorosulfonic acid ionomers and related polymers. *Macromolecules*, 15:320–323, 1982.

[84] D. Wu, S. J. Paddison, and J. A. Elliott. A comparative study of the hydrated morphologies of perfluorosulfonic acid fuel cell membranes with mesoscopic simulations. *Energy Environ. Sci.*, 1:284–293, 2008.

[85] K. Prater (from Ballard Power Systems Inc.). The Renaissance of the Solid Polymer Fuel Cell. *J. Power Sources*, 29:239–250, 1990.

[86] G. A. Eisman. *Proceedings of the 168th Electrochemical Society Meeting*, 83-13:156–171, 1986.

[87] M. R. Tant, K. P. Darst, K. D. Lee, and C. W. Martin. *Multiphase polymers: Blends and Ionomers*, chapter 15, pages 370–400. ACS Symposium Series, Vol. 395, 1989.

[88] G. A. Eisman. The application of Dow Chemical's perfluorinated membranes in proton-exchange membrane fuel cells. *J. Power Sources*, 29:389–398, 1990.

[89] J. Halim, F. N. Büchi, O. Haas, M. Stamm, and G. G. Scherer. Characterization of perfluorosulfonic acid membranes by conductivity measurements and small-angle x-ray scattering. *Electrochim. Acta*, 39:1303–1307, 1994.

[90] S. S. Jang, V. Molinero, T. Çağin, and W. A. Goddard III. Nanophase-Segregation and Transport in Nafion 117 from Molecular Dynamics Simulations: Effect of Monomeric Sequence. *J. Phys. Chem. B*, 108:3149–3157, 2004.

[91] A. Venakatnathan, R. Devanathan, and M. Dupuis. Atomistic Simulations of Hydrated Nafion and Temperature Effects on Hydronium Ion Mobility. *J. Phys. Chem. B*, 111:7234–7244, 2007.

[92] N. Idupulapati, R. Devanathan, and M. Dupuis. Ab initio study of hydration and proton dissociation in ionomer membranes. *J. Phys. Chem. A*, 114:6904–6912, 2010.

[93] E. Spohr, P. Commer, and A. A. Kornyshev. Enhancing Proton Mobility in Polymer Electrolyte Membranes: Lessons from Molecular Dynamics Simulations. *J. Phys. Chem. B*, 106:10560–10569, 2002.

[94] S. J. Paddison. The modeling of molecular structure and ion transport in sulfonic acid based ionomer membranes. *J. New. Mat. Electr. Sys.*, 4:197–207, 2001.

[95] S. J. Paddison and R. Paul. The Nature of Proton Transport in Fully Hydrated Nafion. *J. New. Mat. Electr. Sys.*, 4:1158–1163, 2002.

[96] F. Shulu and G. A. Voth. Proton Solvation and Transport in Hydrated Nafion. *J. Phys. Chem. B*, 115:5903–5912, 2011.

[97] D. R. Morris and X. Sun. Water-Sorption and Transport Properties of Nafion 117 H. *Journal of Applied Polymer Science*, 50:1445–1452, 1993.

[98] S. Gottesfeld and T. A. Zawodzinski. *Advances in Electrochemical Science and Engineering*, chapter 4, pages 195–297. J. Wiley & sons, New York, 1997.

[99] H. J. C. Berendsen, J. R. Grigera, and T. P. Straatsma. The Missing Term in Effective Potentials. *J. Phys. Chem.*, 91:6269–6271, 1987.

[100] I. Kusaka, Z.-G. Wang, and J. H. Seinfeld. Binary nucleation of sulfuric acid-water: Monte Carlo Simulation. *J. Chem. Phys.*, 108:6829–6848, 1998.

[101] J. M. J. Swanson, C. M. Maupin, H. Chen, M. Pettersen, J. Xu, Y. Wu, and G. A. Voth. Proton Solvation and Transport in Aqueous and Biomolecular Systems: Insights from Computer Simulations. *J. Phys. Chem. B*, 111:4300–4314, 2007.

[102] G. A. Voth. Computer Simulation of Proton Solvation and Transport in Aqueous and Biomolecular Systems. *Acc. Chem. Res.*, 39:143–150, 2006.

[103] M. Čuma, U. W. Schmitt, and G. A. Voth. A multi-state empirical valence bond model for acid-based chemistry in aqueous solution. *Chemical Physics*, 258:187–199, 2000.

[104] M. Čuma, U. W. Schmitt, and G. A. Voth. A Multi-State Empirical Valence Bond Model for Weak Acid Dissociation in Aqueous Solution. *J. Phys. Chem. A*, 105:2814–2823, 2001.

[105] T. J. F. Day, A. V. Soudackov, M. Čuma, U. W. Schmitt, and G. A. Voth. A second generation multistate empirical valence bond model for proton transport in aqueous systems. *J. Chem. Phys.*, 117:5839–5849, 2002.

[106] U. W. Schmitt and G. A. Voth. Multistate Empirical Valence Bond Model for Proton Transport in Water. *J. Phys. Chem. B*, 102:5547–5551, 1998.

[107] S. S. Jang, M. Blanco, W. A. Goddard III, G. Caldwell, and R. B. Ross. The Source of Helicity in Perfluorinated N-Alkanes. *Macromolecules*, 36:5331–5341, 2003.

[108] S. L. Mayo, B. D. Olafson, and W. A. Goddard III. DREIDING: a Generic Force Field for Molecular Simulations. *J. Phys. Chem. B*, 94:8897–8909, 1990.

[109] M. Levitt. Calibrating and Testing of a Water Model for Simulation of the Molecular Dynamics of Proteins and Nucleic Acids in Solution. *J. Phys. Chem. B*, 101:5051–5061, 1997.

[110] P. G. Khalatur, S. K. Talitskikh, and A. R. Khokhlov. Structural Organization of Water-Containing Nafion: The Integral Equation Theory. *Macromolecular Theory and Simulations*, 11:566–586, 2002.

[111] D. A. Mologin, P. G. Khalatur, and A. R. Khokhlov. Structural Organization of Water-Containing Nafion: A Cellular-Automaton-Based Simulation. *Macromolecular Theory and Simulations*, 11:587–607, 2002.

[112] S. Yamamoto and S. Hyodo. A Computer Simulation Study of the Mesoscopic Structure of the Polyectrolyte Membrane Nafion. *Polymer Journal*, 35:519–527, 2003.

[113] J. T. Wescott, Y. Qi, L. Subramanian, and T. W. Capehart. Mesoscale simulation of morphology in hydrated perfluorosulfonic acid membranes. *J. Phys. Chem.*, 124:134702–134714, 2006.

[114] S. Urata, J. Irisawa, A. Takada, W. Shinoda, S. Tsuzuki, and A. Takada. Molecular Dynamics Simulation of Swollen Membrane of Perfluorinated Ionomer. *J. Phys. Chem. B*, 109:4269–4278, 2005.

[115] R. W. Impey, P. A. Madden, and I. R. McDonald. Hydration and Mobility of Ions in Solution. *The Journal of Physical Chemistry*, 87:5071–5083, 1983.

[116] D. A. McQuarrie. *Statistical Mechanics.* Harper & Row, New York, 1976.

[117] D. Chandler. *Introduction to Modern Statistical Mechanics.* Oxford University Press, 1976.

[118] J. W. Gibbs. *Elementary principles in statistical mechanics, developed with especial reference to the rational foundation of thermodynamics*. Charles Scribner's sons, New York, 1902.

[119] D. Frenkel and B. Smit. *Understanding Molecular Simulation From Algorithms to Applications*. Academic Press, second edition, 2002.

[120] A. R. Leach. *Molecular Modelling Principles and Applications*. Pearson Education, Edinburgh, second edition, 2001.

[121] E.E. Polymeropoulos, P. A. Bopp, J. Brickmann, L. Jansen, and R. Block. Molecular-dynamics simulations in systems of rare gases using Axilrod-Teller and exchange three-atom interactions. *Phys. Rev. A*, 31:3565–3569, 1985.

[122] M. Wójcik and E. Clementi. Molecular dynamics simulation of liquid water with three-body forces included. *J. Chem. Phys.*, 84:5970–5971, 1986.

[123] M. Wójcik and E. Clementi. Single molecule dynamics of three body water. *J. Chem. Phys.*, 85:3544–3549, 1986.

[124] M. Wójcik and E. Clementi. Collective dynamics in three body water and sound dispersion. *J. Chem. Phys.*, 85:6085–6092, 1986.

[125] M. M. Probst, E. Spohr, K. Heinzinger, and P. A. Bopp. A Molecular Dynamics Simulation of an Aqueous Beryllium Chloride Solution. *Mol. Simulation*, 7:5970–5971, 1991.

[126] A. Lauenstein, K. Hermansson, J. Lindgren, M. M. Probst, and P. A. Bopp. Molecular Dynamics Simulation of an Aqueous Aluminium(III) Chloride Solution with Three-body Interactions. *Int. J. Quantum Chem*, 80:5970–5971, 2000.

[127] G. C. Maitland. *Intermolecular Forces: Their Origin and Determination*. Oxford University Press, 1987.

[128] D. A. Case, T. A. Darden, T. E. Cheatham III, C. L. Simmerling, J. Wang, R. E. Duke, R. Luo, R. C. Walker, W. Zhang, K. M. Merz, B. Roberts, S. Hayik, A. Roitberg, G. Seabra, J. Swails, A. W. Goetz, I. Kolossvai, K. F. Wong, F. Paesani, J. Vanicek, R. M. Wolf, J. Liu, X. Wu, S. R. Brozell, T. Steinbrecher, H. Gohlke, Q. Cai, X. Ye, J. Wang, M. J. Hsieh, G. Cui, D. R. Roe, D. H. Mathews, M. G. Seetin, R. Salomon-Ferrer, C. Sagui, V. Babin, T. Luchko, S. Gusarov, A. Kovalenko, and P. A. Kollman. *AMBER 12*. University of California, San Francisco, 2012.

[129] D. van der Spoel, E. Lindahl, B. Hess, G. Groenhof, A. E. Mark, and H. J. C. Berendsen. GROMACS: Fast, Flexible and Free. *Journal of Computational Chemistry*, 26:1701–1718, 2005.

[130] B. R. Brooks, R. E. Bruccoleri, B. D. Olafson, D. J. States, S. Swaminathan, and M. Karplus. A Program for Macromolecular Energy, Minimization, and Dynamics Calculations. *Journal of Computational Chemistry*, 4:187–217, 1983.

[131] W. D. Cornell, P. Cieplak, C. I. Bayly, I. R. Gould, K. M. Merz, D. M. Ferguson, D. C. Spellmeyer, T. Fox, J. W. Caldwell, and P. A. Kollman. A Second Generation Force Field for the Simulation of Proteins, Nucleic Acids, and Organic Molecules. *J. Am. Chem. Soc.*, 117:5179–5197, 1995.

[132] A. D. MacKerell Jr., D. Bashford, M. Bellott, R. L. Dunbrack Jr., J. D. Evanseck, M. J. Field, S. Fischer, J. Gao, H. Guo, S. Ha, D. Joseph-McCarthy, L. Kuchnir, K. Kuczera, F. T. K. Lau, C. Mattos, S. Michnick, T. Ngo, D. T. Nguyen, B. Prodhom, W. E. Reiher III, B. Roux, M. Schlenkrich, J. C. Smith, R. Stote, J. Straub, M. Watanabe, J. Wirkiewicz-Kuczera, D. Yin, and M. Karplus. All-Atom Empirical Potential for Molecular Modeling and Dynamics Studies of Proteins. *J. Phys. Chem. B*, 102:3586–3616, 1998.

[133] J. Chen, W. Im, and C. L. Brooks III. Balancing Solvation and Intramolecular Interactions: Toward a Consistent Generalized Born Force Field. *J. Am. Chem. Soc.*, 128:3728–3736, 2006.

[134] W. L. Jorgensen and J. Tirado-Rives. The opls [optimized potentials for liquid simulations] potential functions for proteins, energy minimizations for crystals of cyclic peptides and crambin. *J. Am. Chem. Soc.*, 110:1657–1666, 1988.

[135] W. L. Jorgensen, D. S. Maxwell, and J. Tirado-Rives. Development and Testing of the OPLS All-Atom Force Field on Conformational Energetics and Properties of Organic Liquids. *J. Am. Chem. Soc.*, 118:11225–11236, 1996.

[136] B. Guillot. A Reappraisal of What we have Learnt During Three Decades of Computer Simulations on Water. *J. Mol. Liquids*, 101:219–260, 2002.

[137] J. C. Soetens, M. T. C. Martins Costa, and C. Millot. Static dielectric constant of the polarizable NCC water model. *Mol. Phys.*, 94:577–579, 1998.

[138] J. Richardi, P. H. Fries, and J. C. Soetens. A generalized self-consistent mean-field theory for fluids of molecules with distributed polarizabilities: Comparisons with computer simulations. *J. Mol. Liquids*, 88:209–228, 2000.

[139] S. Flügge, P. Walger, and A. Weiguny. A generalization of the morse potential for diatomic molecules. *Journal of Molecular Spectroscopy*, 23:243–257, 1967.

[140] 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*, 23:327–341, 1977.

[141] 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*, 18:1463–1472, 1997.

[142] R. Lustig. Microcanonical Monte Carlo simulation of thermodynamic properties. *J. Chem. Phys.*, 109:8816–8828, 1998.

[143] M. P. Allen and D. J. Tildesley. *Computer Simulation of Liquids.* Oxford University Press, 1987.

[144] P. P. Ewald. Die Berechnung optischer und elektrostatischer Gitterpotentiale. *Annalen der Physik*, 369:253–287, 1921.

[145] C. Kittel. *Introduction to Solid State Physics.* J. Wiley & sons, New York, 5th edition, 1976.

[146] D. J. Adams. Computer simulation of ionic systems: The distorting effects of the boundary conditions. *Chem. Phys. Lett.*, 62:329–332, 1979.

[147] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak. Molecular dynamics with coupling to an external bath. *J. Chem. Phys.*, 81:3684–3690, 1984.

[148] S. Nosé. A molecular dynamics method for simulations in the canonical ensemble. *Mol. Phys.*, 52:255–268, 1984.

[149] W. G. Hoover. Canonical dynamics: Equilibration phase-space distributions. *Phys. Rev. A*, 31:1695–1697, 1985.

[150] G. Bussi, D. Donadio, and M. Parrinello. Canonical sampling through velocity rescaling. *J. Chem. Phys.*, 126:014101–014108, 2007.

[151] M. Parrinello and A. Rahman. Polymorphic transitions in single crystals: A new molecular dynamics method. *Journal of Applied Physics*, 52:7182–7190, 1981.

[152] S. Nosé and M. L. Klein. Constant pressure molecular dynamics for molecular systems. *Mol. Phys.*, 50:1055–1076, 1983.

[153] S. Hayward and N. Go. Collective Variable Description of Native Protein Dynamics. *Annual Review of Physical Chemistry*, 46:223–250, 1995.

[154] E. B. Wilson, J. C. Decius, and P. C. Cross. *Molecular Vibrations.* Mc Graw Hill 1955, republished by Dover, New York, 1980.

[155] F. Lado. Numerical fourier transforms in one, two, and three dimensions for liquid state calculations. *Journal of Computational Physics*, 8:417–433, 1971.

[156] `http://www.ccp5.ac.uk/ftpfiles/ccp5/allen_tildesley.tar.gz`.

[157] E. Spohr, G. Pálinkás, K. Heinzinger, P. A. Bopp, and M. M. Probst. A Molecular Dynamics Study of an Aqueous $SrCl_2$ Solution. *J. Phys. Chem.*, 92:6754–6761, 1988.

[158] S. Le Roux and P. Jund. Ring statistics analysis of topological networks: New approach and application to amourphous GeS_2 and SiO_2 systems. *Computational Materials Science*, 49:70–83, 2010.

[159] W. Humphrey, A. Dalke, and K. Schulten. VMD – Visual Molecular Dynamics. *Journal of Molecular Graphics*, 14:33–38, 1996.

[160] K. D. Kreuer. On the development of proton conducting polymer membranes for hydrogen and methanol fuel cells. *J. Membrane Sci.*, 185:29–39, 2001.

[161] S. Goldman and P. Backx. On water's ability to dissociate ion pairs. *J. Chem. Phys.*, 84:2761–2765, 1986.

[162] T. Darden, D. York, and L. Pedersen. Particle mesh Ewald: An $N \cdot \log(N)$ method for Ewald sums in large systems. *J. Chem. Phys.*, 98:10089–10092, 1993.

[163] U. Essmann, L. Perera, M. Berkowitz, T. Darden, and H. Lee et al. A smooth particle mesh Ewald method. *J. Chem. Phys.*, 103:8577–8593, 1995.

[164] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, and J. Hermann. INTERACTION MODELS FOR WATER IN RELATION TO PROTEIN HYDRATION. *Intermolecular Forces*, pages 331–342, 1981.

[165] G. A. Kaminski, R. A. Friesner, J. Tirado-Rives, and W. L. Jorgensen. Evaluation and Reparametrization of the OPLS-AA Force Field for Proteins via Comparison with Accurate Quantum Chemical Calculations on Peptides. *J. Phys. Chem. B*, 105:6474–6487, 2001.

[166] D. Brandell, J. Karo, A. Liivat, and J. O. Thomas. Molecular dynamics studies of the nafion, dow and aciplex fuel-cell polymer membrane. *J. Mol. Model.*, 13:1039–1046, 2007.

[167] I. T. Todorov, W. Smith, K. Trachenko, and M. T. Dove. DL_POLY_3: new dimensions in molecular dynamics simulations *via* massive parallelism. *Journal of Materials Chemistry*, 16:1911–1918, 2006.

[168] J. Wang, P. Cieplak, and P. A. Kollman. How well does a restrained electrostatic potential (RESP) model perform in calculating conformational energies of organic and biological molecules? *Journal of Computational Chemistry*, 21:1049–1074, 2000.

[169] J. Wang, R. M. Wolf, J. W. Caldwell, P. A. Kollman, and D. A. Case. Development and testing of a general amber force field. *Journal of Computational Chemistry*, 25:1157–1174, 2004.

[170] R. A. Kendall, E. Aprà, D. E. Bernholdt, E. J. Bylaska, M. Dupuis, G. I. Fann, R. J. Harrison, J. Ju, J. A. Nichols, J. Nieplocha, T. P. Straatsma, T. L. Windus, and A. T. Wong. High performance computational chemistry: An overview of NWChem a distributed parallel application. *Comput. Phys. Commun.*, 128:260–283, 2000.

[171] E. Aprà, T. L. Windus, T. P. Straatsma, E. J. Bylaska, W. de Jong, Hirata W., M. Valiev, M. Hackler, L. Pollack, K. Kowalski, R. Harrison, M. Dupuis, D. M. A Smith, J. Nieplocha, V. Tipparaju, M. Krishnan, A. A. Auer, E. Brown, G. Cisneros, G. Fann, H. Fruchtl, J. Garza, K. Hirao, R. Kendall, J. Nichols, K. Tsemekhman, K. Wolinski, J. Anchell, D. Bernholdt, P. Borowski, T. Clark, D. Clerc, H. Dachsel, M. Deegan, K. Dyall, D. Elwood, E. Glendening, M. Gutowski, A. Hess, J. Jaffe, B. Johnson, J. Ju, R. Kobayashi, R. Kutteh, Z. Lin, R. Littlefield, X. Long, B. Meng, T. Nakajima, S. Niu, M. Rosing, G. Sandrone, M. Stave, H. Taylor, G. Thomas, J. van Lenthe, A. Wong, and Z. Zhang. *NWChem, A Computational Chemistry Package for Parallel Computers, Version 4.7.* Pacific Northwest National Laboratory: Richland, WA, 2006.

[172] J. Anderson, J. J. Ullo, and S. Yip. Molecular dynamics simulations of dielectric properties of water. *J. Chem. Phys.*, 87:1726–1732, 1987.

[173] H. E. Alper and R. M. Levy. Computer simulations of the dielectric properties of water: Studies of the simple point charge and transferrable intermolecular potential models. *J. Chem. Phys.*, 91:1242–1251, 1989.

[174] M. Eikerling, A. A. Kornyshev, A. M. Kuznetsov, J. Ulstrup, and S. Walbran. Mechanisms of Proton Conductance in Polymer Electrolytes Membranes. *J. Phys. Chem. B*, 105:3646–3662, 2001.

[175] S. P. Narasimachary, A. Roudgar, and M. H. Eikerling. *Ab initio* study of interfacial correlations in polymer electrolyte membranes for fuel cells at low hydration. *Electrochim. Acta*, 53:6920–6927, 2008.

[176] A. Roudgar, S. P. Narasimachary, and M. Eikerling. *Ab initio* study of surface-mediated proton transfer in polymer electrolyte membranes. *Chem. Phys. Lett.*, 457:337–341, 2008.

[177] A. Roudgar, S. P. Narasimachary, and M. Eikerling. Hydrated Arrays of Acidic Surface Groups as Model Systems for Interfacial Structure and Mechanisms in PEMs. *J. Phys. Chem. B*, 110:20469–20477, 2006.

[178] A. Luzar. Resolving the hydrogen bond dynamics conundrum. *J. Phys. Chem.*, 113:10663–10675, 2000.

[179] A.-L. Rollet, G. Gebel, J.-P. Simonin, and P. Turq. A SANS Determination of the Influence of External Conditions on the Nanostructure of Nafion Membrane. *Journal of Polymer Science Part B: Polymer Physics*, 39:548–558, 2001.

[180] K. Malek, M. Eikerling, Q. Wang, Z. Liu, S. Otsuka, K. Akizuki, and M. Abe. Nanophase segregation and water dynamics in hydrated Nafion: Molecular modeling and experimental validation. *J. Phys. Chem.*, 129:204702–204711, 2008.

[181] S. Cui, J. Liu, M. E. Selvan, S. J. Paddison, D. J. Keffer, and B. J. Edwards. Comparison of the hydration and diffusion of protons in perfluorosulfonic acid membranes with molecular dynamics simulations. *J. Phys. Chem. B*, 112:13273–13284, 2008.

[182] S. Cui, J. Liu, M. E. Selvan, D. J. Keffer, and W. V. Steele. A Molecular Dynamics Study of a Nafion Polyectrolyte Membrane and the Aqueous Phase Structure for Proton Transport. *J. Phys. Chem. B*, 111:2208–2218, 2007.

[183] J. A. Elliott, D. Wu, S. J. Paddison, and R. B. Moore. A unified morphological description of Nafion membranes from SAXS and mesoscale simulations. *Soft Matter*, 7:6729–7124, 2011.

[184] R. Mills. Self-Diffusion in Normal and Heavy Water in the Range 1-45°. *J. Phys. Chem.*, 77:685–688, 1973.

[185] M. Holz, S. R. Heil, and A. Sacco. Temperature-dependent self-diffusion coefficients of water and six selected molecular liquids for calibration in accurate ^{1}H NMR PFG measurements. *Phys. Chem. Chem. Phys.*, 2:4740–4742, 2000.

[186] T. A. Zawodzinski, J. Davey, J. Valerio, and S. Gottesfeld. THE WATER CONTENT DEPENDENCE OF ELECTRO-OSMOTIC DRAG IN PROTON-CONDUCTING POLYMER ELECTROLYTES. *Electrochim. Acta*, 40:297–302, 1995.

[187] S. C. Yeo and A. Eisenberg. Physical properties and supermolecular structure of perfluorinated ion-containing (nafion) polymers. *Journal of Applied Polymer Science*, 21:875–898, 1977.

[188] A. Gotte, D. Spångberg, K. Hermansson, and M. Baudin. Molecular Dynamics study of oxygen self-diffusion in reduced CeO_2. *Solid State Ionics*, 178:1421–1427, 2007.

[189] Z. Liang, W. Chen, J. Liu, S. Wang, Z. Zhou, W. Li, G. Sun, and Q. Xin. FT-IR study of the microstructure of Nafion membrane. *J. Membrane Sci.*, 233:39–44, 2003.

[190] G. E. Walrafen, M. R. Fisher, M.S. Hokmabadi, and W.-H. Yang. Temperature dependence of the low- and high-frequency Raman scattering from liquid water. *J. Chem. Phys.*, 85:6970–6982, 1986.

[191] R. H. Byrd, P. Lu, and J. Nocedal. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific and Statistical Computing*, 16:1190–1208, 1995.

[192] C. Zhu, R. H. Byrd, and J. Nocedal. L-BFGS-B: Algorithm 778: L-BFGS-B, FORTRAN routines for large scale bound constrained optimization. *ACM Transactions on Mathematical Software*, 23:550–560, 1997.

[193] D. Liotard. Lecture: 'Algorithmes de prospection d'une SEP'. *Master Chimie, Université Bordeaux 1, academic year 2008/2009*, 2008.

9 Curriculum Vitae

Name:	Gabriel Jacques Frédéric Marchand
Marital Status:	Single
Date of Birth:	17th May 1985
Citizenship:	French

EDUCATION

2008-2012	PhD in chemistry (in progress): **Advanced Computer Simulations of Nafion / Water Systems** Duisburg-Essen University, Germany; Bordeaux University, France Supervisors: Pr. Dr. E. Spohr, Pr. Dr. Ph. A. Bopp
2006-2008	Master's degree in chemistry, Bordeaux University, France: **The Dynamics of the Water Molecules in the Vicinity of the Ions in Aqueous LiCl Solutions up to Saturation Studied by Correlation Functions.** Supervisor: Pr. Dr. Ph. A. Bopp
2003-2006	Bachelor's degree in chemistry, Bordeaux University, France Supervisor: Dr. André del Guerzo
2003	Baccalauréat (equivalent to Abitur/A-levels), Lycée Montesquieu High School Bordeaux

WORK EXPERIENCE

2008-2012	Research and teaching assistantship in Duisburg-Essen University

LANGUAGES

French: mother tongue; English: fluent; German: intermediate

CONTACT

E-mail:	gabriel.marchand@uni-due.de , gabriel.marchand@etu.u-bordeaux1.fr
Adresses:	34 allee Davezac, F-33200 Bordeaux, France

10 'Erklärung'

Hiermit versichere ich, dass ich die vorliegende Arbeit mit dem Titel

"Advanced Computer Simulations of
Nafion / Water Systems"

selbst verfasst und keine außer den angegebenen Hilfsmittel und Quellen benutzt habe. Diese Arbeit wird außer bei der Universität Duisburg-Essen im Rahmen eines mit dieser Universität vertraglich vereinbarten gemeinsamen Promotionsverfahrens (thèse en co-tutelle) ebenfalls bei der Université Bordeaux 1 eingereicht.

Essen, im Mai 2012

11 Acknowledgments

First of all I would like to thank my supervisors professors Eckhard Spohr and Philippe Bopp for their in-depth knowledge, advise, encouragement, and also for giving me scientific freedom.

Financial support came through professor Spohr's grant from the Universität Dusiburg-Essen.

I would also like to thank all colleagues and for their help and scientific discussions, in particular professor Jean-Cristophe Soetens and Dr. Holger Somnitz. Special thanks go to Mehmet Ali Ilhan and Igor Shapiro.

À la famille, celle qui nous veut du bien.

à Bush, Benji, Flo, Mehmet, Boris, Bouit, pour qui j'éprouve une profonde amitié.

À Christelle.

À tous les gens que j'aime et qui m'aiment, que ce soit d'Eysines, de Bordeaux, d'Essen ou d'ailleurs. Ils se reconnaîtront.

Merci pour tout (sauf les colères, et encore, j'estime avoir été épargné) PAB!

À Evgenia!