Università degli Studi di Sassari
Dipartimento di Chimica e Farmacia

Scuola di Dottorato in Scienze e Tecnologie Chimiche
Indirizzo Scienze Chimiche
Ciclo XXVI

Hierarchical Multiscale Modeling of Materials: an Application to Microporous Systems

Tesi di Dottorato di
Andrea Gabrieli

Il Direttore
Prof. Stefano Enzo

Il Supervisore
Prof. Pierfranco Demontis

Il Co-Supervisore
Dr. Marco Sant

Anno Accademico 2012/2013

Alla mia famiglia

Contents

Chapter 1

Introduction

Computer modeling techniques have now established as a standard method for the study of chemical and physical phenomena in complex systems. The origin of their success is twofold: from one side, the continuous increase in the computational power, along with the spread of parallel architectures, has made accessible to a wide audience the most accurate methodologies, while, on the other side, the refinement of the techniques themselves has improved the reliability of the results. As a consequence, scientists now have access to a variety of tools useful to make predictions, to test theories, and to understand phenomena experimentally unaccessible. Nevertheless, when dealing with multiscale systems, i.e., systems whose behavior is controlled by phenomena that occur at different space and time scales, it is not possible to effectively use just one single simulation technique. This problem is largely due to inherent limitations of the methods, and it depends only slightly on the finite nature of computing resources. In Molecular Dynamics, for example, the highest vibrational frequency in the system imposes an upper bound to the length of the time step. A similar problem can be found in kinetic Monte Carlo, where the space and time scales are tied in such a way that the extension of one, causes the shrinking of the other. In general a highly detailed technique can be employed only to investigate very small scales. Unfortunately, no technique is free from these kinds of issues, thus creating a hierarchy among them.

The extension of space and time scales can be achieved through two different approaches: one consists in connecting different levels in the hierarchy by a coarse-graining of the information coming from highly detailed levels, and the other consists in overcoming the intrinsic limitations of the single techniques.

In this thesis, the possibility of using both approaches for large scale simulations has been investigated. The methods here developed have been applied to the study of microporous systems, being particularly well suited for a multiscale approach. Nonetheless, such methods are completely general, and can be employed for the simulation of a wide class of materials without too much effort.

The thesis is organized as follows. In chapter 2 the theory of the main simulation techniques employed is briefly reported. In chapter 3 the potential of using a parallel kinetic Monte Carlo algorithm for the study of large zeolitic systems is investigated. It is shown that it is possible to achieve very good efficiencies, thus obtaining an extension of space and time scales. In chapter 4, starting from a Partitioning Cellular Automaton, a simplified coarse-grained model of the hopping process of a tagged particle in a confined lattice system has been developed, providing an accurate reproduction of the memory effects in the self-diffusion at a minimum computational cost. In chapter 5 a new force field for molecular dynamics simulations in flexible aluminosilicates has been developed, choosing a functional form which can be used in a number of MD packages, so that massively parallel architectures can be exploited, thus extending the space and time scales accessible to classical MD simulations. Finally in chapter 6 a new fast implementation of the *force matching* technique is presented. Starting from detailed *ab-initio* data, it is shown that it is possible to obtain accurate classical molecular dynamics force fields, tailored to each specific structure.

Chapter 2

Theoretical background

2.1 *Ab-initio*

The quantum mechanical formalism, developed at the beginning of the 20th century, provides rigorous foundation for the prediction of observable physical properties from first principle. In this framework all physical systems are described by a fundamental (mathematical) object, the *wavefunction* $\mathbf{\Psi}$ from which it is possible to retrieve the values of physical *observables* e by means of quantum mechanical *operators* θ:

$$\theta\mathbf{\Psi} = e\mathbf{\Psi}. \tag{2.1}$$

In linear algebra representation $\mathbf{\Psi}$ is a N-element column vector, and is called an eigenfunction, e is a scalar, and is called an eigenvalue, and the operator θ is an $N \times N$ square matrix. In the Born interpretation [1,2] $|\Psi|^2$ is a probability density and the wavefunction is a probability amplitude. Consequently, the probability to find the system in some region of the multi-dimensional space is given by integrating the density over that region, and when the whole space is considered, the following relation

$$\int |\mathbf{\Psi}|^2 \, d\boldsymbol{\tau} < \infty, \tag{2.2}$$

where $d\boldsymbol{\tau}$ is the volume element, is true, in particular it is equal to 1 provided that the wavefunctions are normalized.

The time evolution of the system is entirely defined by the time dependent Schrödinger equation:

$$\mathrm{i}\hbar\frac{\partial\mathbf{\Psi}}{\partial t} = H\mathbf{\Psi}. \tag{2.3}$$

In this equation H is the operator for the total system energy, the Hamiltonian, given by the sum of the kinetic energy T and potential energy V operators:

$$H = T + V. \tag{2.4}$$

If the potential energy is independent of time, the Equation 2.3 can be rewritten as:

$$\boldsymbol{\Psi}(\mathbf{r}, t) = \boldsymbol{\psi}(\mathbf{r})\mathrm{e}^{(-\mathrm{i}Et/\hbar)}, \tag{2.5}$$

where $\boldsymbol{\psi}(\mathbf{r})$ is the time independent Schrödinger equation:

$$H\boldsymbol{\psi} = E\boldsymbol{\psi}. \tag{2.6}$$

The wavefunction, though, is a very complicated object and is in general unknown. For molecular systems it has to be determined by solving the Schrödinger equation 2.6 for a system consisting of interacting electrons and nuclei. Commonly H, contains the following terms:

$$H = -\sum_i \frac{\hbar^2}{2m_e}\nabla_i^2 - \sum_k \frac{\hbar^2}{2m_k}\nabla_k^2 - \sum_i\sum_k \frac{e^2 Z_k}{r_{ik}} + \sum_{i<j}\frac{e^2}{r_{ij}} + \sum_{k<l}\frac{e^2 Z_k Z_l}{r_{kl}}, \tag{2.7}$$

where indices i and j refer to electrons, k and l refer to nuclei, m_e is the electron mass and e is the electron charge, m_k is the k-th nucleus mass and Z_k its atomic number, and r_{ab} is the distance between particles a and b. Each term represents, in order, the kinetic energy of electrons, the kinetic energy of the nuclei, the electron-nuclei attraction, the electron-electron repulsion and the nuclei-nuclei repulsion. Note that, depending on the system of interest, other terms can be introduced for example to take into account the effect of external fields.

Unfortunately, an analytical solution for this equation exists only only for systems so simple as to be devoid of interest in real world applications. Also its numerical solution is a difficult task, and requires several approximations. The problems are caused by correlations in the motion of the particles.

Before discussing these approximations is appropriate to introduce the variation theorem:

$$\langle \boldsymbol{\phi} \,|\, H \,|\, \boldsymbol{\phi} \rangle \geq E_0, \tag{2.8}$$

given a system described by an Hamiltonian H with lowest eigenvalue E_0, and an arbitrary trial wavefunction $\boldsymbol{\phi}$, the eigenvalue of the energy will be equal to E_0 if and only if $\boldsymbol{\phi} \equiv \boldsymbol{\psi}$ the true ground state wavefunction of the system [1]. This theorem is of fundamental importance because, as will be seen below, provides a practical way to obtain the desired wavefunction.

The first approximation, which permits to overcome the difficulties related to the motion of the nuclei, is to express the wavefunction as a product of an electron wavefunction $\boldsymbol{\psi}_{el}$, which depends on the electronic coordinates $\mathbf{q}_i$ and only parametrically on the nuclei coordinates $\mathbf{q}_k$, and a nuclear wavefunction $\boldsymbol{\psi}_{nuc}$ which depends only on $\mathbf{q}_k$:

$$\boldsymbol{\psi}(\mathbf{q}_i, \mathbf{q}_k) = \boldsymbol{\psi}_{el}(\mathbf{q}_i; \mathbf{q}_k)\boldsymbol{\psi}_{nuc}(\mathbf{q}_k). \tag{2.9}$$

This is the Born-Oppenheimer approximation and it is based on the fact that the motion of the nuclei is orders of magnitude slower than that of the electrons, so it is safe to assume that they respond to a change in their position instantaneously [2]. This permits to treat the electrons as they are moving in the field generated by the fixed nuclei (the proton mass is about 1800 times larger than that of the electron, so the kinetic energy term of the nuclei in the Hamiltonian becomes negligible). Moreover the electron-nuclei correlation vanishes and the nuclei-nuclei potential energy term becomes a constant. It is then possible to solve separately two Schrödinger equations. For our purposes we are interested only in the electronic one:

$$(H_{el} + V_{nuc})\boldsymbol{\psi}_{el}(\mathbf{q}_i; \mathbf{q}_k) = E_{el}\boldsymbol{\psi}_{el}(\mathbf{q}_i; \mathbf{q}_k). \tag{2.10}$$

The Born-Oppenheimer approximation is ubiquitous in quantum chemistry because it holds in the vast majority of cases and it allows a great simplification of the calculations.

The problems caused by the electron correlation are far more difficult to solve and, in fact, are still today object of active research. The way the fourth term of Equation 2.7 is approximated is crucial because the accuracy of the results will ultimately depends on it.

Hartree-Fock. The first step in approximating the electron correlation is to not consider it at all. One idea can then be to express the N-electrons wavefunction with a product of N one-electron wavefunctions:

$$\boldsymbol{\psi}^{\circ} = \psi_1\psi_2\cdots\psi_N, \tag{2.11}$$

which is called an “Hartree-product” wavefunction [2]. This approximation is possible because in the case of non interacting electrons, only the kinetic energy and the potential term due to the nuclei are present in the Hamiltonian, which is then separable:

$$H = \sum_{i=1}^{N} h_i, \tag{2.12}$$

where h_i is the one electron Hamiltonian:

$$h_i = -\frac{1}{2}\nabla_i^2 - \sum_{k=1}^{M} \frac{Z_k}{r_{ik}}, \tag{2.13}$$

where M is the number of nuclei. Each ψ_i is an eigenfunction of h_i (they are solutions of the one-electron Schrödinger equation) and the energy eigenvalue is simply the sum of the one-electron energies.

By virtue of the variational principle 2.8 the energy computed applying the correct Hamiltonian on the Hartree-product wavefunction will be higher than the true ground state energy. One then wish to find the set of ψ_i (called orbitals) which minimizes:

$$E = \langle \boldsymbol{\psi}^{\circ} \,|\, H \,|\, \boldsymbol{\psi}^{\circ} \rangle \,. \tag{2.14}$$

Relying again on the variational principle it is possible to show that each ψ_i is an eigenfunction of the following operator:

$$h_i = -\frac{1}{2}\nabla_i^2 - \sum_{k=1}^{M} \frac{Z_k}{r_{ik}} + \sum_{j \neq i} \int \frac{\rho_j}{r_{ij}}\, d\mathbf{r}, \tag{2.15}$$

where the third term represents the interaction of the electron i with the charge density ρ_j of the electron j. This means that the electrons interact with each other in an averaged way and not instantaneously.

The problem here is that, to determine each individual ψ_i, the knowledge of all the others is required, being $\rho_j = |\psi_j|^2$. The solution to this problem has been given by Hartree [1,2] which introduced a procedure called Self Consistent Field (SCF). This procedure consists in guessing a wavefunction for each electron and then solve the corresponding one-electron Schrödinger equation. The thus obtained new (hopefully improved) wavefunctions are used to compute ρ and are then used as a starting point for a new calculation. This way, iteratively solving the Schrödinger equation, it is possible to systematically improve the wavefunctions. When the change between two consecutive calculated wavefunctions is negligible the iterations will stop and the last set of ψ_i are accepted as an approximation of the *true* wavefunction.

At this point, to be more correct, it is necessary to take into account the spin and the fact that the electrons must obey the Pauli principle. The wavefunction including the spin can be written as the product of a spatial part ψ with a spin part α or β and, for example, a two electrons Hartree-product wavefunction can be written as:

$$\boldsymbol{\psi}^{\circ} = \psi_a(1)\alpha(1)\psi_b(2)\alpha(2), \tag{2.16}$$

this wavefunction still does not satisfy the Pauli principle, which requires the function to be antisymmetric with respect to an interchange of two electrons coordinates. To construct such a wavefunction it is possible to use the Slater determinant:

$$\boldsymbol{\psi}^{\circ} = (N!)^{-1/2} \det \left| \chi_a(1)\chi_b(2)\cdots\chi_N(N) \right| , \tag{2.17}$$

where χ is a *spinorbital*, a joint spin-space state of the electron (i.e., the product of spatial and spin eigenfunction) [1]. Among the features of such a determinant one is extremely important: the quantum mechanical *exchange*. It consists in a reduction of the classical Coulomb repulsion between electrons. This is a consequence of the Pauli principle and represents the fact that electrons with same spin tend to avoid each other. There is a depletion in the probability of finding an electron in the proximity of another having the same spin. This depletion is called *Fermi hole*.

The use of wavefunctions obtained from Slater determinant in the Hartree SCF procedure, was first proposed by Fock and is at the basis of the so called Hartree-Fock method, which is a milestone in the field of *ab-initio* computations. In this method the interaction of each electron with the static field of all of the others includes exchange effects.

Post Hartree-Fock. Clearly the Hartree-Fock wavefunction is not *exact*. As stated before, the coulombic interaction between electrons is considered only on average, thus neglecting instantaneous and quantum mechanical effects [1]. In other words it does not take into account electron correlation. To include those effects and improve the quality of of the wavefunction, over the years, a great effort has been made, which led to the development of many methods. Some of them like the Configuration Interaction and the Multiconfiguration methods, rely on the use of linear combinations of Slater determinants, while others, like the Møller-Plesset method, rely on the Perturbation-Theory. In any case, even if they are extremely accurate, those methods are unsuitable for studying systems consisting in more than a few tens of atoms because of their computational cost. For this reason, in recent years, a new method, based on totally different assumptions, has emerged: the Density Functional Theory (DFT).

2.2 DFT

In Density Functional Theory as a central quantity, the wavefunction is replaced by the electron density. The electronic energy is said to be a *functional* of the electron density, meaning that at each given function $\rho(\mathbf{r})$ is associated only one value for the energy:

$$E[\rho(\mathbf{r})].$$

Early implementations date back to the late twenties of the last century with the work of Thomas [3] and Fermi [4]. Their model however was too approximated to be actually used, being not able to correctly bind atoms in molecules. In subsequent years other models were developed like the one of Bloch [5] and

Dirac [6] or the Xα method of Slater [7]. But was only in the mid-sixties with the work of Hohenberg and Kohn that a formal proof of the basic assumptions of DFT was given [8, 9].

As just mentioned, the fundamental quantity in DFT is the electronic density ρ. In a molecule the electrons interact one with another and with an *external potential* generated by the nuclei. Hohenberg and Kohn proved via *reductio ad absurdum* that "the ground state density determines the external potential". Let us start by assuming that given a nondegenerate ground state density ρ_0, can be consistent with two different external potential v_a and v_b. The corresponding Hamiltonian H_a and H_b have associated a ground state wave function ψ_0 and energy eigenvalue E_0. The variational principle (see equation 2.8) states that the expectation value of a given Hamiltonian over an arbitrary wavefunction is always higher then the ground state value. It is then possbile to write:

$$E_{0,a} < \langle \psi_{0,b} | H_a | \psi_{0,b} \rangle , \tag{2.18}$$

which, after some simple algebraic manipulation, leads to:

$$E_{0,a} < \int [v_a(\mathbf{r}) - v_b(\mathbf{r})] \rho_0(\mathbf{r}) \, d\mathbf{r} + E_{0,b}, \tag{2.19}$$

but, being a and b arbitrary also the following holds:

$$E_{0,b} < \int [v_b(\mathbf{r}) - v_a(\mathbf{r})] \rho_0(\mathbf{r}) \, d\mathbf{r} + E_{0,a}, \tag{2.20}$$

finally, summing this two expressions it is easy to get:

$$E_{0,a} + E_{0,b} < E_{0,b} + E_{0,a}. \tag{2.21}$$

This is clearly an impossible result which falsifies the initial assumption. The consequence is that the external potential and thus the Hamiltonian, are determined by the non-degenerate ground state density. This implies that "the ground state energy and all other ground state electronic properties are uniquely determined by the electron density" [1, 2, 10].

Having demonstrated the existence of a unique relation between ρ and the ground state energy, what is missing is a method to obtain the electron density.

Thanks again to Hohenberg and Kohn a step towards the realization of this method was made: they demonstrate the existence of a variational principle. Given a guess for the density which satisfies $N = \int \rho(\mathbf{r}) \, d\mathbf{r}$, with N number of electrons, the existence theorem asserts that the Hamiltonian H_g and the wavefunction ψ_g are uniquely determined. Relying again on the variational principle it is possible to write:

$$\langle \psi_g | H_g | \psi_g \rangle = E_g \geq E_0. \tag{2.22}$$

The energy E_g evaluated using a trial electron density is greater or equal to the ground state energy E_0. One possible way to get the desired electron density is, in analogy to the Hartree-Fock method, the consecutive refinement of the trial wavefunction until a certain accuracy is reached. This approach, though, is not useful for two reasons, one is that there is no practical way to compute an improved electron density, and the other is that, even if such a way existed, there would be no improvement in the computational efficiency or simplification with respect to the Hartree-Fock method, having still to solve the Schrödinger equation.

Was in 1965 that Kohn and Sham provided a method to solve the problem [9]. The key point of their formulation is to work with a fictitious system of N non-interacting electrons having the same ground state density of the real (interacting) one. The Hamiltonian for this system is then greatly simplified being the sum of one-electron operators. Moreover its eigenfunctions are Slater determinants of the one-electron eigenfunctions and the eigenvalues are simply the sum of the one-electron eigenvalues.

It is possible to rewrite the expression for the energy as a sum of several terms, namely: the kinetic energy of of the non-interacting electrons T_{ni}, the nucleus-electron interaction V_{ne}, the (classical) electron-electron interaction V_{ee}, the kinetic energy difference between the interacting and non-interacting electrons ΔT, the non classical corrections to the electron-electron interaction ΔV_{ee}:

$$E[\rho(\mathbf{r})] = T_{ni}[\rho(\mathbf{r})] + V_{ne}[\rho(\mathbf{r})] + V_{ee}[\rho(\mathbf{r})] + \Delta T[\rho(\mathbf{r})] + \Delta V_{ee}[\rho(\mathbf{r})]. \tag{2.23}$$

Describing the electrons in terms of orbitals (like in the Hartree-Fock theory) we get another expression for the ground state energy:

$$\begin{aligned} E[\rho(\mathbf{r})] = &\sum_{i}^{N} \left(\left\langle \chi_i \middle| -\frac{1}{2}\nabla_i^2 \middle| \chi_i \right\rangle - \left\langle \chi_i \middle| \sum_{k}^{nuclei} \frac{Z_k}{|\mathbf{r}_i - \mathbf{r}_k|} \middle| \chi_i \right\rangle \right) \\ &+ \sum_{i}^{N} \left\langle \chi_i \middle| \frac{1}{2} \int \frac{\rho(\mathbf{r}')}{|\mathbf{r}_i - \mathbf{r}'|} d\mathbf{r}' \middle| \chi_i \right\rangle + E_{xc}[\rho(\mathbf{r})], \end{aligned} \tag{2.24}$$

where in this equation χ_is are the so called *Kohn-Sham orbitals* and the ground state electron density is:

$$\rho = \sum_{i=1}^{N} \langle \chi_i | \chi_i \rangle \, . \tag{2.25}$$

The last term of Equation 2.24 E_{xc} is called *exchange-correlation energy*. This term, in addition to the effects of exchange and correlation, also includes corrections for the self-interaction energy [2], and for the kinetic energy difference between interacting and non-interacting electrons. This is the only term for which an exact analytical form is not known and for this reason must be approximated.

By solving the Kohn-Sham (KS) equations it is possible to compute the orbitals which minimize the energy. They have the following form:

$$\left\{ -\frac{1}{2}\nabla_i^2 - \sum_k^{nuclei} \frac{Z_k}{|\mathbf{r}_i - \mathbf{r}_k|} + \int \frac{\rho(\mathbf{r}')}{|\mathbf{r}_i - \mathbf{r}'|}\, d\mathbf{r}' + V_{xc} \right\} \chi_i = \epsilon_i \chi_i, \tag{2.26}$$

which is a pseudoeigenvalue equation. V_{xc}, the exchange-correlation potential, is the functional derivative of the exchange-correlation energy:

$$V_{xc} = \frac{\delta E_{xc}}{\delta \rho}. \tag{2.27}$$

The knowledge of the KS orbitals allows the computation of the electron density ρ. The solution of the KS equations is carried out with a procedure analogous to that used for the Hartree-Fock SCF. The first step is to compute (solving the Equation 2.26) a set of KS orbitals using an initial guess for the density and some fixed form for E_{xc}. The thus obtained orbitals are used in Equation 2.25 to compute a new improved density, which in turn becomes the starting point for a new cycle of calculations. The procedure is repeated until the change in some property (usually the density) between to consecutive steps falls below a given threshold. At the end it is possible to use this optimized set of orbitals to compute the electronic energy. It is important to note that the orbitals thus obtained provide the exact density, being the energy under minimization the exact one. The formulation here reported is then in principle *exact*. In real applications, however, it is approximate, being unknown the analytical form of E_{xc}. It is then crucial for the accuracy of the computation to have a good approximation for this term, for this reason this is still an active field of research [11].

Introducing the energy density ϵ_{xc}, which is dependent on the electron density, it is common to express E_{xc} as:

$$E_{xc}[\rho(\mathbf{r})] = \int \rho(\mathbf{r})\epsilon_{xc}[\rho(\mathbf{r})]\, d\mathbf{r}, \tag{2.28}$$

and it is also common to separate it in an exchange and in a correlation only part:

$$E_{xc} = E_x + E_c.$$

During the years a large number of such approximations have been developed. One of the first is the local density approximation (LDA) for which an extension to spin-polarized systems is straightforward (i.e., the local spin density LSD approximation [2]). Here the exchange correlation energy at a given position **r** depends only on the value of the electron density at that position. Although the definition is very general, only one formulation is used in practice and is derived from the analysis of the homogeneous electron gas with constant density, where $\epsilon_{xc}[\rho(\mathbf{r})]$ is the exchange correlation energy of an electron moving in a space of infinite volume containing a uniform and continuous distribution of positive charge [1, 12]. Despite of this simple depiction the LDA functionals are quite good in predicting several properties, in particular the structural ones, and have found applications in solid state [12]. This approximation fails when dealing with molecules where the electron density is far from being uniform. For example the binding energies are overestimated so it is not well suited for solving chemical problems.

To overcome the limitations of the LDA approach the Generalized Gradient Approximation (GGA) was developed. In this framework not only the exchange correlation energy depends on the value of the electron density at a given position but also on its gradient. The general formulation for a GGA functional is the following:

$$E_{xc}^{GGA}[\rho_\uparrow, \rho_\downarrow] = \int f(\rho_\uparrow(\mathbf{r}), \rho_\downarrow(\mathbf{r}), \nabla\rho_\uparrow, \nabla\rho_\downarrow)\, d\mathbf{r}, \tag{2.29}$$

where $\rho\uparrow$ and $\rho\downarrow$ are the spin density α and β respectively. The most common GGA functionals are built by simply adding the gradient correction to an LDA functional for both the exchange and correlation part (indicated by x/c):

$$\epsilon_{x/c}^{GGA}[\rho(\mathbf{r})] = \epsilon_{x/c}^{LSD}[\rho(\mathbf{r})] + \Delta\epsilon_{x/c}\left[\frac{|\nabla\rho(\mathbf{r})|}{\rho^{4/3}(\mathbf{r})}\right]. \tag{2.30}$$

The GGA formulation largely improves the results obtained from LSD calculations. Among the improved properties are worth mentioning [12, 13]: total energy, atomization energy (errors are reduced by a factor of about 5), energy barriers and in general the description of bonds (in some cases there are overcorrections [13]).

The number of proposed functions f (Equation 2.29) is really large, and there is no unique recipe to choose the best one. Two remarkable examples are the BLYP and PBE functionals. The first is composed by the exchange developed by Becke [14] (B) and the correlation developed by Lee, Yang, and Parr [15] (LYP) and is largely employed in molecular calculations, while the latter dominates the field of materials, in particular when systems are large, and was developed by Perdew, Burke, and Ernzerhof [13].

The subsequent step in the approximation of the exchange correlation energy is obtained replacing a variable portion of the exchange part with Hartree-Fock exchange:

$$E_{xc} = E_x^{HF} + z(E_{xc}^{DFT} - E_x^{HF}),$$

where z is a parameter to be optimized. The functionals of this class are called hybrid. In 1993 Becke proposed a three parameter scheme [16] which, coupled with the LYP correlation, leads to the famous B3LYP functional.

The drawback of this kind of functionals is that, having included a part from Hartree-Fock, the computational cost is increased with respect to GGA, and applications to large systems require supercomputers and *ad hoc* codes [17].

Despite the successes achieved by the DFT there are some aspects that are particularly difficult to deal with, as they are inherent in the derivation of the method. For the purposes of this thesis, the most relevant is the failure to describe the dispersion interactions. In recent years, however, much efforts have been made to solve this problem and we can expect further improvements in the near future [11, 18].

DFT computations are extensively employed for the work carried out in chapter 6.

2.3 Molecular Dynamics

Molecular Dynamics (MD) is probably the most employed simulation technique when the time evolution of a many-body system is object of interest. The origin of this technique dates back to the late fifties of the last century, with the work of Alder and Wainwright [19, 20]. From then on its adoption and its development have been very rapid, going hand in hand with the increase in computer performance, thus extending the space and time scales accessible. MD is suitable to investigate both equilibrium and transport properties for a wide range of systems, ranging from simple liquids to proteins, even chemical reactions can be taken into account [21–24].

The main assumption in MD is that the motion of the nuclei is governed by the Newton laws of classical mechanics. This is a very good approximation because quantum effects become important only when $h\nu > k_b T$, with ν the highest vibrational frequency in the system, and k_b the Boltzmann constant. In practice this is a concern only if one wants to investigate the motion of light species like H_2 [22].

The aim of an MD simulation is to compute macroscopic properties starting from microscopic informations obtained by solving the following equations:

$$\begin{cases} \dot{q}_i(t) = \dfrac{p_i(t)}{m_i} = v_i(t) \\ \dot{p}_i(t) = -\nabla_{q_i} V(\mathbf{q}(t)) \end{cases} \qquad \longleftrightarrow \qquad m_i \ddot{q}_i = -\nabla_{q_i} V(\mathbf{q}(t)) \tag{2.31}$$

where q_i is the position, p_i is the momentum, v_i the velocity, and m_i the mass of the i-th particle in the system. The vector $\mathbf{q}(t)$ is the configuration of the system at time t, and V is the potential ruling the particle interactions. Statistical mechanics provides the means to perform this task [25–27].

2.3.1 Statistical Mechanics

Given a macroscopic system, its thermodynamic state is completely defined, regardless how complex it is, by a small set of quantities, for example number of particles N, volume V, total energy E. From the microscopic point of view, instead, one needs to know in which of all possible quantum states is the system. Being the number of such states, for an N-body isolated system, of the order of 10^N this is an hopeless task. This problem can be overcome thanks to the ensemble method of Gibbs. An ensemble is a really large collection of $\mathfrak{N}$ mental replica of the system. All replicas are identical only from a thermodynamic point of view (e.g., same *N, V,* and *E* fixed) while they can be in any microstate compatible with the given conditions (which is an extremely large number). The value of a mechanical thermodynamic property[1], at a given time, will in general be different among each replica [27]. The average value of this instantaneous property, computed giving the same statistical weight to each replica is called *ensemble average*. It is then postulated that *the time average of a mechanical variable is equal to the ensemble average, in the limit as* $\mathfrak{N} \to \infty$ [27]. To actually compute the averages it is necessary to know the relative probability of occurrence of different quantum state in the ensemble systems. As there is no reason to believe otherwise, it is postulated that *in an isolated system (N, V, and E fixed) the ensemble replicas are distributed uniformly over the possible quantum states consistent with the values of N, V, and E* [27]. This is also known as *principle of equal a priori probabilities* and, together with the first postulate, implies that an isolated system, after a sufficiently long time, spends an equal amounts of time in each available quantum state. This is the quantum *ergodic hypotesis*.

Similar postulates exist even in the case of a classical system. In such case the state of the system is fully described by $3N$ positions q_i, and $3N$ conjugate momenta p_i. These are coordinates of a $6N$ dimensional *phase space* and the time evolution of the system is described by a point moving through this space according to the Equations 2.31.

Again it is possible to replace the time averages with the ensemble averages. For example one builds an ensemble of $\mathfrak{N}$ replicas of an isolated system consistent with the given values of *N, V,* and *E* fixed. Each element of this ensemble

[1] Quantity which can be defined in purely mechanical terms without appealing to the concept of temperature, e.g., pressure.

can be represented by a point of the *same* phase space evolving independently from the others. The whole ensemble, being $\mathfrak{N} \to \infty$, can be seen as a *cloud* of points with a continuous density. To compute the ensemble average, in analogy to the second postulate of the quantum case, it is postulated that *the density of phase points is constant throughout the region of phase space between the surface* E $=$ *constant and* E + $\delta E =$ *constant, with* δE *arbitrarily small* [27]. This means that all regions of phase space having points consistent with the thermodynamics of the system are equally important [25]. Finally the classical *ergodic hypotesis* states that *an isolated system, after a sufficiently long time, spends an equal amounts of time in equals volumes of phase space between the surface* E $=$ *constant and* E + $\delta E =$ *constant, with* δE *arbitrarily small* [27].

It is now possible to compute an observable A with the following integral over the phase space:

$$A = \langle A \rangle = \int A(\mathbf{q},\mathbf{p}) f(\mathbf{q},\mathbf{p})\, d\mathbf{q}\, d\mathbf{p}, \tag{2.32}$$

where $\langle A \rangle$ is the ensemble average, and $A(\mathbf{q},\mathbf{p})$ is the value of A at a given phase point. $f(\mathbf{q},\mathbf{p})$ is the probability to observe a certain configuration $(\mathbf{q},\mathbf{p})$ and, for example, in the case of N, V, and T fixed, is given by:

$$Z^{-1} e^{-\dfrac{H(\mathbf{q},\mathbf{p})}{k_b T}} = f(\mathbf{q},\mathbf{p}),$$

where Z is the canonical partition function, H the energy of the given configuration, k_b the Boltzmann constant, and T the temperature. The problem now is that the integral 2.32 is $6N$-dimensional and is impossible to compute: for example given a system of 10^6 particles in two dimensions, if one wants to use a numerical quadrature on a grid with 10 points per dimension, will end up to evaluate $10^{2 \cdot 10^6}$ times the function [28]. This problem can be overcome by using the fact that, if ergodicity holds:

$$\lim_{t \to \infty} \frac{1}{t} \int_0^t A(\mathbf{q}(\theta),\mathbf{p}(\theta))\, d\theta = \int A(\mathbf{q},\mathbf{p}) f(\mathbf{q},\mathbf{p})\, d\mathbf{q}\, d\mathbf{p}, \tag{2.33}$$

the time averages are equals to the ensemble averages. Being obviously not possible to follow the time evolution of the system forever, it is necessary to assume that the first integral of 2.33 can be approximated with:

$$\frac{1}{t_{end}} \int_0^{t_{end}} A(\mathbf{q}(\theta),\mathbf{p}(\theta))\, d\theta \simeq \langle A \rangle, \tag{2.34}$$

which is true for large t_{end}. Finally it is possible to write the discrete time version of the 2.34:

$$\frac{1}{n_{tot}}\sum_{i=1}^{n_{tot}} A(\mathbf{q}^i, \mathbf{p}^i) \simeq \langle A \rangle. \tag{2.35}$$

This summation can be easily computed by means of MD, where n_{tot} is the number of configuration snapshots and $\mathbf{q}^i \simeq \mathbf{q}(i\Delta t)$ and $\mathbf{p}^i \simeq \mathbf{p}(i\Delta t)$ are the numerical approximation of $\mathbf{q}(\theta)$ and $\mathbf{p}(\theta)$, thus demonstrating that it is possible to measure macroscopic properties from a MD simulation provided that a large enough number of system configurations is generated.

To better understand the link between statistical mechanics ensembles and MD trajectories it is more convenient to rewrite the problem in the Hamiltonian formalism:

$$\begin{cases} \dot{\mathbf{q}}_i(t) = \dfrac{\partial H}{\partial \mathbf{p}_i(t)} \\[2ex] \dot{\mathbf{p}}_i(t) = -\dfrac{\partial H}{\partial \mathbf{q}_i(t)} \end{cases} \tag{2.36}$$

These are the canonical equations of Hamilton, which are a set of $2N$ first order differential equations [29] and are equivalent to the Newton equations (second law). H is the Hamiltonian function and must be constructed based on the problem of interest, $\mathbf{q}_i$ are the N generalized coordinates and $\mathbf{p}_i$ are the N generalized (conjugate) momenta [29] and are also called canonical variables.

When only interparticle interactions are considered, the Hamiltonian takes the form $H(\mathbf{q}(t), \mathbf{p}(t)) = \sum \frac{\mathbf{p}_i^2}{2m_i} + V(\mathbf{q}(t))$. Differentiating the previous equation it is possible to show that H is constant:

$$\frac{d}{dt}\left[H(\mathbf{q}(t), \mathbf{p}(t))\right] = \frac{\partial H}{\partial \mathbf{q}}\frac{d\mathbf{q}(t)}{dt} + \frac{\partial H}{\partial \mathbf{p}}\frac{d\mathbf{p}(t)}{dt}. \tag{2.37}$$

Considering for simplicity the one-dimensional case, $H(\mathbf{q}(t), \mathbf{p}(t)) = \frac{p^2}{2m} + V(q(t))$ substituting in the 2.37 one gets:

$$V'(q(t))\frac{p(t)}{m} + \frac{p(t)}{m}(-V'(q(t))) = 0. \tag{2.38}$$

This is also true for the N-dimensional case:

$$H(\mathbf{q}(t), \mathbf{p}(t)) = H(\mathbf{q}(0), \mathbf{p}(0)) \qquad \forall t \geq 0, \tag{2.39}$$

which means that the Hamiltonian does not depend explicitly on the time. Moreover if V is a conservative potential then H is equal to the total energy of the system [29], and then the energy is a conserved quantity. It is now clear the connection between MD and statistical mechanics, in particular with the so called *microcanonical* ensemble where *N*, *V*, and *E* are fixed, which represents an isolated system. In any case, it is always possible through appropriate modifications of the formulation proposed here, to perform simulation consistent with other ensembles [30].

2.3.2 Integrator

Hamiltonian equations have in general no analytical solution, so a numerical method is needed. The integration methods currently adopted in MD are variants of the finite difference method which are used to find approximate solution for the problem:

$$\begin{cases} \dfrac{d}{dt}\mathbf{q} = \dfrac{\mathbf{p}}{m} \\ \\ \dfrac{d}{dt}\mathbf{p} = -\nabla V(\mathbf{q}) \end{cases} \Longrightarrow \dot{y}(t) = f(y), \qquad \text{with } y \in \mathbb{R}^2, \tag{2.40}$$

subject to the appropriate boundary conditions and with initial conditions $y(t_0)$. As an example it is possible to use the Euler method where the new configuration at discrete step $n+1$ is given by:

$$y^{n+1} = y^n + \Delta t f(y^n), \tag{2.41}$$

where y^n is the numerical approximation of $y(n\Delta t)$. A numerical method is said to be *convergent* if the difference between y^n and $y(n\Delta t)$ goes to zero when Δt goes to zero:

$$\lim_{\Delta t \to 0} \left(\max_{0 \le n \le N} ||y^n - y(n\Delta t)|| \right) = 0. \tag{2.42}$$

This limit, in practice, is difficult to verify as it is, but it becomes possible by means of 2 concepts: the consistency and the stability. A method which is stable and consistent is also convergent.

Consistency. The consistency is related to the error over one time step, in other words to the local (truncation) error of the numerical scheme starting from an exact value. For example expanding the solution at a time $t = 0 + \Delta t$ in Taylor series one gets:

$$y(\Delta t) = y(0) + \Delta t y'(0) + \frac{\Delta t^2}{2} y''(0) + O(\Delta t^3). \tag{2.43}$$

Recognizing that $y'(0)$ is $f(y(0))$ and dropping all the terms above the first order one obtains:

$$y(\Delta t) = y(0) + \Delta t f(y(0)) + O(\Delta t^2), \tag{2.44}$$

where the first two terms on the right hand side are simply Equation 2.41 with $y^0 \equiv y(0)$. Finally it is possible to define the truncation error as:

$$e(\Delta t) \equiv O(\Delta t^{p+1}) = y(\Delta t) - y(0) - \Delta t f(y(0)), \tag{2.45}$$

where p is called the order of consistency. A method is said to be consistent if the following holds:

$$\lim_{\Delta t \to 0} \frac{e(\Delta t)}{\Delta t} = 0, \tag{2.46}$$

which means that the error vanishes removing the discretization.

Stability. The stability is related to the sensibility of the procedure to perturbations. To estimate it one starts with two close different solutions y^0 and z^0 and then integrate them with the numerical algorithm, adding to one of the two, say z, at each step a small perturbation δ. A method is said to be stable if there exists a constant $S > 0$ such that the following relation is satisfied:

$$\max_{0<n<N} ||y^n - z^n|| \leq S \left(||y^0 - z^0|| + \sum_{n=0}^{N} ||\delta^n|| \right), \tag{2.47}$$

which means that the method is insensitive to small perturbations like numerical errors. It is important to note that it is not possible for a numerical algorithm to reproduce accurately the *correct trajectory* for a really long time. Two trajectories in fact are subject to exponential divergence in time. This, however, is not a problem, because as long as the energy is conserved, they are statistically equivalent [30]. The accuracy of the numerical scheme is ultimately related to the magnitude of the time step Δt, the smaller it is the smaller is the error. On the other hand, a small time step requires more steps to simulate the same total *real time*, consequently a compromise must be reached. In practice the time step is chosen to be a fraction of the fastest vibration period, thus defining an upper bound to the time scales accessible.

To reproduce the Hamiltonian dynamics, energy conservation (ensured by the convergence) is not enough. Two other requirements must be fulfilled, namely time reversibility and symplecticity. The first, means that integrating backward in time (i.e., reversing the velocities) the previous configurations with $\mathbf{p}$ of the opposite sign, have to be reproduced[2], while the second means that

[2] This is true only from a theoretical point of view due to the roundoff errors.

geometric properties of the phase space like the preservation of volume are not lost [30].

One integrator which possesses all of this features is the Verlet algorithm [31]. It is present in almost all MD programs and owes its success to its great stability. To integrate the equation of motion it relies on the positions $\mathbf{q}$ at time t and $t - \Delta t$ and on the acceleration $\mathbf{a}$ at time t. The positions at time $t + \Delta t$ are computed with the following equation:

$$\mathbf{q}(t + \Delta t) = 2\mathbf{q}(t) - \mathbf{q}(t - \Delta t) + \Delta t^2 \mathbf{a}(t). \tag{2.48}$$

This originates by summing the two following Taylor expansions:

$$\begin{aligned} \mathbf{q}(t + \Delta t) &= \mathbf{q}(t) + \Delta t \mathbf{v}(t) + \frac{1}{2}\Delta t^2 \mathbf{a}(t) + O(\Delta t^3), \\ \mathbf{q}(t - \Delta t) &= \mathbf{q}(t) - \Delta t \mathbf{v}(t) + \frac{1}{2}\Delta t^2 \mathbf{a}(t) - O(\Delta t^3). \end{aligned} \tag{2.49}$$

As it is clear from the previous equations velocities are not computed, so, being useful in a simulation, they can be evaluated via the following relation:

$$\mathbf{v}(t) = \frac{\mathbf{q}(t + \Delta t) - \mathbf{q}(t - \Delta t)}{2\Delta t}. \tag{2.50}$$

The only drawback of this approach is that the error on $\mathbf{v}$ is of order Δt^2 while the error on Equation 2.48 is of order Δt^4 [21].

Several variants of this algorithm exist but one of particular interest is the so called *velocity* Verlet [21]. The equations involved are the following:

$$\begin{aligned} \mathbf{q}(t + \Delta t) &= \mathbf{q}(t) + \Delta t \mathbf{v}(t) + \frac{1}{2}\Delta t^2 \mathbf{a}(t), \\ \mathbf{v}(t + \Delta t) &= \mathbf{v}(t) + \frac{1}{2}\Delta t [\mathbf{a}(t) + \mathbf{a}(t + \Delta t)]. \end{aligned} \tag{2.51}$$

Here positions, velocities, and accelerations are evaluated at the same time t and the round-off errors are minimized. With respect to the basic Verlet algorithm, the velocity version works in two stages and there is a computation of the forces in between the two. The first stage consists in the computation of the new positions, then the velocities are computed at mid-step with:

$$\mathbf{v}(t + \frac{1}{2}\Delta t) = \mathbf{v}(t) + \frac{1}{2}\Delta t \mathbf{a}(t). \tag{2.52}$$

Finally forces and accelerations at time $t + \Delta t$ are computed, thus permitting to complete the velocity move:

$$\mathbf{v}(t + \Delta t) = \mathbf{v}(t + \frac{1}{2}\Delta t) + \frac{1}{2}\Delta t \mathbf{a}(t + \Delta t). \tag{2.53}$$

This algorithm is stable, simple, and very convenient thus explaining its wide adoption.

2.3.3 Interparticle interactions

Up to this point nothing has been said about V. This is a central quantity in MD because it describes the way the system constituents interact. The accuracy of the entire simulation will ultimately depends from the quality of the forces computed by the algorithm, which derive from the potential. In classical molecular dynamics V is approximated with a functional form which has to be parametrized. The parametrization can be obtained by fit of experimental data, *ab-initio* potential energy surfaces or both. This topic will be covered in great detail in chapters 5 and 6.

Another possibility is to compute *on the fly* the potential by means of *ab-initio* calculations. This is the core of the *ab-initio* Molecular Dynamics (AIMD) technique. Also in this case there are a lot of variants [32], and for some of the calculations carried out in chapter 6 the Born-Oppenheimer Molecular Dynamics (BOMD) technique has been adopted [33]. In this variant atoms positions are propagated in time using the classical equations of motion, which take the following form:

$$m_i \ddot{\mathbf{q}}_\mathbf{i}(t) = -\nabla_i E(\{\mathbf{q}_i\}) = -\nabla_i \min_{\rho(r)} E(\{\mathbf{q}_i\}, \rho(r)). \tag{2.54}$$

The potential energy surface is given by the electronic ground state energy (last term of the equation 2.54) [33].

The main advantage of AIMD with respect to classical MD lies in the fact that at each MD step an full electronic structure calculation is performed on the given configuration of atoms. In principle, then, there is no limit on the range of phenomena that can be simulated, even if they are unexpected. In MD instead it is not possible to predict the accuracy of the results outside the conditions used for fitting the potential energy function. On the other hand, being the approximation of the method linked to the way the Schrödinger equation is approximated, it is possible that, to reach the desired accuracy, the computational cost will become prohibitive. This is in fact the reason why these kind of simulations became accessible only in recent years thanks to the combination of two factors: the increase of computational power and the rise of the DFT.

It is the main topic of chapter 6 investigating the possibility to build a bridge between this two techniques.

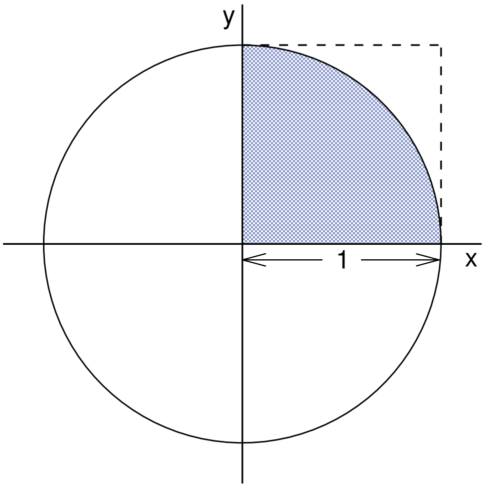


Figure 2.1: Graphical representation of a simple Monte Carlo integration scheme [21].

2.4 Stochastic methods

2.4.1 Monte Carlo

With molecular dynamics it is possible to compute macroscopic observables by replacing phase space integrals 2.32 which is an ensemble average, with a time average. Another way to solve multidimensional phase space integrals, which actually is prior to MD, is the so called Monte Carlo method. This is a stochastic method, and owes its name to the extensive use of random numbers [34]. Its origins date back to the late forties of the last century, not by chance coinciding with the birth of the first computer. In fact, stochastic methods have existed for many years, but being really long and tedious to perform, it was only with the advent of computers that they began to succeed [21, 35, 36].

The basic idea of the method can be explained by means of a simple example [21], the computation of π. Looking at the Figure 2.1 and moving the attention only to the first quadrant it is easy to see that, taking random points inside the dashed square (for example throwing darts towards it), one will hits the circle τ_{hit} times and the ratio τ_{hit}/τ_{shot} will approach the ratio between the shaded area and the area of the square itself as the number of shots increase. The desired value will be then:

$$\pi = \lim_{\tau_{shot} \to \infty} \frac{4\tau_{hit}}{\tau_{shot}}. \tag{2.55}$$

The actual computation is carried out by repeatedly extracting two random numbers, one for the x coordinate and one for the y coordinate. These numbers are produced by a so called random number generator (RNG) which typically is nothing more than a computer code capable to return a sequence of numbers which resembles a real random one. For this reason the numbers thus generated are also called *pseudo* random numbers. It is important to note that the overall accuracy of a Monte Carlo procedure is heavily influenced by the quality of the RNG.

Another example of Monte Carlo integration procedure which has found some use in the past [37] is the *sample mean* method. In this case the integral:

$$F = \int_a^b f(x)\, dx, \tag{2.56}$$

can be approximated by an average over a large number of trials. This approach is similar to the standard quadrature, the only difference is that, instead of evaluating the function at predetermined set of points, it is evaluated taking τ_{max} random points ζ in the interval (a, b) from a probability distribution $\rho(x)$. The problem then can be rewritten as:

$$F = \int_a^b \frac{f(x)}{\rho(x)} \rho(x)\, dx, \tag{2.57}$$

which is the average of the quantity $\dfrac{f(x)}{\rho(x)}$ over the number of trials. Finally if $\rho(x)$ is uniform in the given interval, by means of the mean value theorem it is possible to approximate F with:

$$F \simeq \frac{(b-a)}{\tau_{max}} \sum_{\tau=1}^{\tau_{max}} f(\zeta_\tau). \tag{2.58}$$

Although this method requires a number of function evaluations much smaller than a standard numerical method, this number is still too large. The solution to this problem consists in extracting the points on which the function will be evaluated from a non-uniform distribution [21].

Importance Sampling. A Monte Carlo procedure generates a random walk through configuration space. The aim is then to sample positions $\mathbf{q}$ according to the Boltzmann distribution $\rho_{NVT} = Z_{\mathbf{q}}^{-1} e^{-\beta V(\mathbf{q})}$. The need of a method capable of sampling only selected regions of configuration space becomes evident considering the integral:

$$\langle A \rangle_{NVT} = \int A(\mathbf{q}) \rho_{NVT}(\mathbf{q})\, d\mathbf{q}. \tag{2.59}$$

Comparing it with Equation 2.57 it is possible to write:

$$\langle A\rangle_{NVT} = \left\langle \frac{A(\boldsymbol{\zeta}_\tau)\rho_{NVT}(\boldsymbol{\zeta}_\tau)}{\rho(\boldsymbol{\zeta}_\tau)} \right\rangle_{trials}, \tag{2.60}$$

where $\boldsymbol{\zeta}_\tau$ is the random configuration space point chosen at step τ. In general ρ_{NVT} is significant in phase space region where $A(\boldsymbol{\zeta}_\tau)$ is close to its average value [38]. This implies that choosing $\rho = \rho_{NVT}$ will let the method explore only the important regions of configuration space.

Metropolis *et al.* in 1953 [39, 40] proposed a method able to generate points according to the equilibrium distribution of choice without ever calculating the partition function [21].

The idea is to construct an aperiodic symmetric Markov chain [41–43] such that it converges to the limiting distribution ρ_{NVT} [40]. A Markov chain is a stochastic process where the outcome of each step belongs to a finite set of states $\{\boldsymbol{\Gamma}_1, \boldsymbol{\Gamma}_2, \ldots, \boldsymbol{\Gamma}_m, \boldsymbol{\Gamma}_n\}$, the time is discrete, and the outcome at step τ depends only on the outcome at step $\tau - 1$ (this last condition is also known as Markov property). The transition probabilities p_{mn} to go from state m to n (to simplify the notation a state Γ_m will be indicated by its identifying number m) do not depend on n and can be used to define a transition matrix $\mathbf{P}$, having entries P_{mn} [41] which are all nonzero and which in columns adds up to one [42]. The transition matrix at time τ is simply given by:

$$\mathbf{P}(\tau) = (\mathbf{P}(1))^\tau, \tag{2.61}$$

and then probability distribution $\rho(\tau)$ at step τ can be computed with:

$$\rho(\tau) = \mathbf{P}^\tau \rho(0). \tag{2.62}$$

This kind of matrices are called *stochastic* and have always an eigenvalue 1 and a right eigenvector ρ^s such that $\mathbf{P}\rho^s = \rho^s$, which is the limiting distribution of the stationary process, and in general, does not depend on the initial $\rho(0)$ (Perron-Frobenius theorem) [42].

The difficult part is to find the correct $\mathbf{P}$ entries. To do so it is possible to impose the sufficient but not necessary [44] condition of *detailed balance*:

$$\frac{P_{mn}}{P_{nm}} = \frac{\rho_n}{\rho_m} = e^{-\beta\Delta V_{mn}}. \tag{2.63}$$

Finally in the Metropolis algorithm the P_{mn} are obtained from:

$$P_{mn} \propto \begin{cases} 1, & \rho_n \geq \rho_m \rightarrow \Delta V_{mn} \leq 0 \\ \\ e^{-\beta\Delta V_{mn}}, & \rho_n < \rho_m \rightarrow \Delta V_{mn} > 0. \end{cases} \tag{2.64}$$

As one can see from previous equations the computation of the partition function is avoided being P_{mn} dependent on the ratio $\dfrac{\rho_n}{\rho_m}$.

The basic steps for a Metropolis Monte Carlo simulation are the following:

- generate an initial configuration $\mathbf{q}_0$;
- propose a new configuration $\mathbf{q}_{test}$ and compute: $\Delta V_{\mathbf{q}_{current},\mathbf{q}_{test}}$ and then the probability P_{mn} following the prescriptions of Equation 2.64;
- accept the new configuration if $P_{mn} > \xi$ with ξ uniform random number in $(0, 1)$;
- repeat until the final step τ_{max} is reached.

2.4.2 Kinetic Monte Carlo

The Monte Carlo schemes previously mentioned completely lack of information on the dynamic evolution of the system. It was only in the sixties, in order to study the radiation damage, that began to appear the first algorithms that were capable to provide also this kind of information. Over the next 20 years they appeared in studies of adsorption on surfaces, diffusion and growth, studies of statistical physics and many others [45]. This technique is particularly useful when the long time evolution of the system of interest is dominated by *rare events*. An event e_i is a transition from a state $\mathbf{\Gamma}_m$ to $\mathbf{\Gamma}_n$, characterized by a transition rate r_i. Rare refers to the fact that the waiting time between events is much larger than the time required to perform a transition.

The underlying idea in kinetic Monte Carlo (kMC) is to build a connection between *real* time and Monte Carlo time (steps) by appealing to the theory of Poisson processes. In the case of rare event systems, this is possible by constructing the transition matrix $\mathbf{P}$ in such a way that, other than fulfill the conditions for sampling the desired limit distribution, a hierarchy among transition probabilities is established, and these probabilities are based on a realistic model of the system dynamics [46].

The master equation:

$$\frac{\partial \rho_n(t)}{\partial t} = \sum_m P_{mn}\rho_m(t) - \sum_n P_{nm}\rho_n(t), \tag{2.65}$$

gives a stochastic description of the system in terms of the time evolution of the probability density function ρ. Here P_{mn}, the entries of the transition matrix $\mathbf{P}$, are probability per unit time. In the long time limit, the solutions to the master equation will tend in general to a stationary one, and if the detailed balance is imposed (Equation 2.63), this stationary solution will be the equilibrium distribution consistent with the chosen thermodynamic conditions [42, 46]. As already seen in the case of the Metropolis algorithm the transition probabilities are not uniquely defined by the above prescription, it is then possible to choose them in such a way that both static (equilibrium) and dynamical properties are reproduced. The main assumption of the method is that the events can not occur simultaneously so that it is possible to represent the evolution of the system as a sequence of events separated by time interval Δt. Moreover it is safe to assume that such a sequence is a Markov chain, which implies that the probability that an event can occur at time t is the same as that occurs at time $t + \Delta t$. This probability depends on the rate but is independent of the previous history. The average rate for a given event is simply the ratio between the number of successful transitions and the observation time, and can be seen as a time density of events [46]. If the observation time is splitted in small equals interval δ, the average rate can be approximated by the ratio between the number of intervals containing events n_δ and the total number of intervals n per unit time δ. This approximation becomes exact in the limit $\delta \to 0$ and $n \to \infty$:

$$r = \lim_{\delta \to 0, t \to \infty} \frac{n_\delta}{t}. \tag{2.66}$$

Considering that each δ in this limit can contain no more than one event with probability $r\delta$ it is possible to compute the probability that n_e events occur in a time t by means of the binomial distribution:

$$P(N_{e,t} = n_e) = \binom{n}{n_e} (r\delta)^{n_e} (1 - r\delta)^{n - n_e}, \tag{2.67}$$

where $N_{e,t}$ is a stochastic variable containing the number of events occurred at time t. In the limit of $n \to \infty$ and $r\delta \to 0$ the 2.67 can be approximated by the Poisson distribution:

$$P(N_{e,t} = n_e) = \frac{(rt)^{n_e}}{n_e!} e^{-rt}. \tag{2.68}$$

This distribution describes a set of objects scattered randomly in a region, which, in this case, are events scattered over a time interval t [46].

In the framework of the Poisson processes theory it is straightforward to obtain the probability density of inter event times t_e:

$$f_{t_e}(t) = re^{-rt}, \tag{2.69}$$

and from this, the mean inter event time is:

$$\langle t_e \rangle = \frac{1}{r}. \tag{2.70}$$

It is then possible to generalize to the case of N independent Poisson processes exploiting the fact that they can be represented by a single large Poisson process whose statistical properties depend on the individual processes [46]:

$$P(N_{e0,t} = n_e) = \frac{(r_{tot}t)^{n_e}}{n_e!} e^{-r_{tot}t}, \tag{2.71}$$

where $N_{e0,t}$ is the stochastic variable counting the number of events occurring in the ensemble and r_{tot} is the sum of each individual rate:

$$r_{tot} = \sum_{i=1}^{N} r_i, \tag{2.72}$$

and then the mean inter event time:

$$\langle t_{e0} \rangle = \frac{1}{r_{tot}}. \tag{2.73}$$

Finally the correct way to assign a time to a kMC step is to draw it from the distribution 2.71:

$$\tau = \frac{-\ln(u)}{r_{\text{tot}}}, \qquad \text{with } u \text{ random number} \in (0,1), \tag{2.74}$$

so that it is independent of the specific rate of the event occurred [45, 46].

It should be now clear that, being the connection between real time and simulated time built through the rates constants r, it is extremely important for these to be chosen so as to accurately reproduce the microscopic dynamics of the system. Typically the rate constants are obtained from MD simulations (classical or *ab-initio*) [43, 45]. In principle, if the rate constants of all possible events are accurately known, averages computed from a kMC trajectory would be *identical* to those obtained form an MD trajectory, but with a lower computational cost [45].

A general kMC algorithm is the following:

- generate an initial configuration $\mathbf{q}_0$;
- generate a list of events n_i with corresponding rates r_i and compute $r_{tot} = \sum_{i=1}^{N} r_i$;
- select an event out of the list and realize it with a probability $P_i = r_i/r_{tot}$;
- assign an inter event time with the 2.74;
- repeat the last three points until final time is reached.

This is one of the most common algorithm and belongs to a class called *rejection free* since a transition happens at each step, but there are many variations and alternatives [47].

There are two main limitations to the space and time scales accessible to a kMC scheme, both related to the fact that it is not possible to execute more than one transition per step. Increasing the system size will increase the number of possible events thus rising the value of r_{tot}. One limitation is due to the relation 2.73 which causes the inter event time to decrease, while the other is related to the computational cost for generating the event list.

A detailed analysis of these problems and a possible solution is presented in chapter 3.

2.4.3 Cellular Automata

A cellular automaton (CA) is a discrete, both in space and time, dynamical system consisting of finite-state variables called *cells* arranged on a uniform grid, whose evolution is governed by a simple, uniform local *rule* [48–51]. According to this rule, at each time step, the new state of a cell is computed from the current state of its neighborhood, and all cells are updated simultaneously.

The idea behind a cellular automaton is to reduce the computation to a fixed sequence of elementary operations. It can be surprising, but a great variety of phenomena can be faithfully modeled by reducing them to bits on a lattice that evolve according to simple local rules [49]. This is due to the fact that cellular automata, actually, are a discrete counterpart to partial differential equations, but unlike these, can be realized exactly by computers [48].

Cellular automata were developed in the late forties, not surprisingly as in the case of Monte Carlo, in coinciding with the development of the first computer. John von Neumann, who was involved in the making of the latter, was the pioneer of the field. Its original idea was to simulate the human brain behavior, so that it was possible to solve complex problems. To do this he wanted to build a machine equipped with self-control and self-repair mechanisms, in which the differences between processors and data were removed, in practice he wanted a machine able to build itself. This machine was then realized, thanks also to the suggestions of Stanislaw Ulam, in the framework of a fully discrete assembly of cells evolving in discrete time step, following a rule which defines the new state of a cell as a function only of the state of its neighboring cell, in analogy to biological systems. The most important feature of the von Neumann automaton lies in the fact that the evolution rule (called von Neumann rule) has the universal computation property, meaning that the rule can simulate any computer circuit [51].

In the following years the CA began to spread in the scientific community and, in 1970 they reached the notoriety even in a wider audience thanks to the John Conway's *game of life* [52]. This CA consists in a square lattice of cells whose state can only be on or off. The updating rule is very simple, if a cell is off and is surrounded by exactly three cell on it turns on, otherwise if a cell is on and is surrounded by less than two or more than three cell on then it turns off. Despite the simplicity of the rule, the automaton shows a rich and complex behavior. The ability of CA to produce complex behavior from simple rules further stimulated their study and, in the eighties, Stephen Wolfram [53] noticed that with CA was possible to study also continuous systems, with the advantage with respect to standard methods of the absence of numerical and truncation errors, thanks to the boolean nature of the automata. Other scientists like Tommaso Toffoli and Norman H. Margolus [54] started to investigate the possibility to produce artificial universes and developed specific hardware to realize this task. It was also in those years that CA started to be viewed as a tool to simulate real systems, considering them as an alternative to the microscopic reality of which preserves the important aspects like time reversibility and simultaneity of the motion [51]. The first step towards a wide adoption of CA for the modeling of physical systems was the recognizing that a model developed in the seventies by Hardy, Pomeau, and de Pazzis for the study of fundamental properties of a gas of interacting particles was actually a cellular automaton. This is the first example of a kind of CA known as *Lattice-Gas Cellular Automata* (LGCA) constituted by particles moving across nodes of a regular lattice. Over the years many improvements have been made to the model, but this has failed in replacing the traditional methods for the study of problems in hydrodynamics. Nonetheless LGCA have been successful in many areas where traditional approaches are not applicable, like flows in porous media, immiscible flows, and reaction-diffusion processes among the others [51].

In recent years, our group of research has developed a lattice-gas cellular automaton for the simulation of adsorption and diffusion in zeolites [55–61]. An application of this model, to which I have contributed, is presented in chapter 4 where details of the implementation are also reported.

2.5 Microporous materials

Zeolites are crystalline microporous aluminosilicates that have found a large number of uses in the chemical industry [62,63]. Their crystal structure consists of a definite channel and cage network extending in one, two or three dimensions. The presence of regular micro-pores provides an environment where the adsorbed molecules no longer move freely, but are restricted to reduced spatial dimensions where peculiar many-body effects make zeolites behave as solid solvents [64]. The diverse physical phenomena occurring in these systems embrace heterogeneous catalysis, percolation, and even a dramatic change of the phase diagram [65].

A lot of efforts have been made to improve such materials, in particular trying to incorporate transition metal ions and organic units as an integral part of the crystal structure, but for the most part have been unsuccessful [66]. Recently, however, an alternative class of materials called Zeolitic Imidazolate Frameworks (ZIFs) has been synthesized. Such materials have a three-dimensional structure consisting of tetrahedral metal ions (M) bridged by imidazolate (Im). Being the angle M−Im−M similar to the Si−O−Si angle in zeolites it is possible to synthesize ZIFs with the same framework topology of zeolites [66,67], with the advantage of a great flexibility in the choice of organic substituents. ZIFs have shown very good chemical and thermal stability, and are already of industrial interest being among the best materials for CO_2 capture [67].

All peculiar properties of those materials are ruled by the dimensionality resulting from the specific network of channels and cages that largely determines the nature of the local interactions and of the long-range order. Moreover the molecular mobility is strongly influenced by the topology of the surrounding medium, which provides the energy landscape through the multifarious interplay between adsorbent-adsorbate and adsorbate-adsorbate interactions. Ranging from electronic transitions to slow molecular migration, a hierarchy of timescales and distances are involved in the many processes happening in the interior of the crystal, whose consequences are at the same time essential and difficult to quantify. These phenomena are still far from being understood, and despite a great deal of effort in theory and computation [68], a fundamental description of the confinement effect is not yet available. In recent years a growing research in multiscale modeling/simulation schemes simple enough to be analyzed and able to capture the essential features of the real physical systems has been reported [61, 69]. The advance of microporous materials science is dependent on the development of an effective and efficient multiscale modeling approach, able to bridge the gap between molecular level interactions and macroscopic properties [70].

In each chapter a detailed description of the investigated structures is reported.

Chapter 3

Speeding up simulation of diffusion in zeolites by a parallel synchronous kMC

Adapted with permission from Andrea Gabrieli, Pierfranco Demontis, Federico G. Pazzona, and Giuseppe B. Suffritti; Physical Review E; 83, 056705 (2011). "Copyright 2011 by the American Physical Society."

http://dx.doi.org/10.1103/PhysRevE.83.056705

Understanding the behaviors of molecules in tight confinement is a challenging task. Standard simulation tools like kinetic Monte Carlo have proven to be very effective in the study of adsorption and diffusion phenomena in microporous materials, but they turn out to be very inefficient when simulation time and length scales are extended. The present study investigates the efficacy and potential of using a parallel kinetic Monte Carlo (kMC) algorithm for multiscale zeolites modeling, and addresses some of the challenges involved in designing *competent* algorithms that solve hard problems quickly, reliably, and accurately.

This chapter is organized as follows. Section 3.1 shortly summarizes the standard kMC method, and outlines the most significant challenges in improving its performance. In Sections 3.2 and 3.3 we introduce the basis of the architecture and the design limits of a parallel version of the algorithm on a discrete system, and in Section 3.4 we discuss an application to a selected system.

3.1 The model

In a kMC simulation [46, 71] a state of the system is represented by a configuration of molecules in a discrete network of sites, and a random walk is performed from state to state [45]. The most widely adopted kMC algorithm is *rejection-free*, meaning that at every step the system makes a transition from one state to another and the time t is advanced by extracting an interevent time from an exponential distribution, that is, $t = -\ln(u)/R_{tot}$ where u is a uniform (pseudo) random number in $(0, 1)$ and $R_{tot} = \sum_i^n r_i$ is the sum of the rates r_i of all possible events n. The standard kMC method scales badly with the size of the system (i.e., with the number of events) because of two factors, namely (i) the time spent in generating, searching, and updating the list of events, and (ii) the proportionality of the interevent time to $1/R_{tot}$ which implies that, given the same number of iterations, the trajectory length decreases with increasing system size.

In a large system it is a fair hypothesis to assume that distant regions do not interact significantly with each other. This is the ground for a parallel kinetic Monte Carlo algorithm able to improve the standard method by overcoming its limitations. The underlying idea in parallelizing kMC is the partitioning of the system in domains, where it is possible to execute a sequential algorithm. The domains are independent of each other, consequently by assigning each domain to a different processor the number of events will be reduced, along with the value of R_{tot}. This in turn will raise the efficiency and lengthen the trajectory.

The major problem in parallelizing kMC is the complete asynchronicity of the algorithm. In the rejection-free kMC, at every time step an event is selected and realized. The corresponding time depends on the rates of *all* the possible events. This implies that a parallel approach consisting only of executing serial kMC algorithms independently of each other is correct only if the noninteraction condition between domains is rigorously respected. In real systems this is unachievable since interactions or transfers of matter at the boundaries between domains cannot be avoided. Moreover, each domain has its own timeline and in order to avoid causality errors it is necessary to synchronize and correct them. Despite that, many methods were developed to rigorously treat these problems (see, for example, [72, 73]) thus permitting us to have a parallel kMC procedure able to solve the same master equation of a sequential one. The major drawback of these methods is that they can be highly expensive and complicated to be implemented. The work of Martínez *et al.* [74] shows that ignoring the interaction between domains introduces an error that can be controlled through a careful choice of the domain size. This leads to a great simplification of the algorithm and improves the efficiency. Moreover, this method avoids causality errors by synchronizing the time across domains through the introduction of a *null event*.

3.2 Parallel algorithm

Our parallel algorithm is a manipulation of the continuous synchronous kMC introduced by Martínez *et al.* [74], adapted to a discrete lattice. The first step is the spatial decomposition of the lattice in K domains (where K is equal to the number of processors) named Ω_k, $k = 1, \ldots, K$. The domain shape is arbitrary and the optimal choice, aimed to minimize the communication between domains, is strictly problem dependent. In principle, domains do not necessarily have to be equivalent. They can be assigned heterogeneous sizes and shapes to attain the best optimization possible. In the present case the domains are chosen to be (all equivalent) parallelepiped-shaped (see Section 3.4.2). The simulation proceeds as follows.

- In each domain, say Ω_k, a list of the possible events n_k and relative rates r_{ik} ($i = 1, \ldots, n_k$) is generated. Rates can be summed to give a total rate R_k for each of the K domains:

$$R_k = \sum_{i}^{n_k} r_{ik}. \tag{3.1}$$

  It is worth noting that if the system was not subdivided into domains, the value of R_k would be simply equal to the sum of the rates of all the events as in the sequential case. This implies that the subdivision does not alter the set of states the system can reach.

- The synchronicity of time horizon between domains is ensured by selecting the greatest among all values of R_k:

$$R_{\max} = \max_{k=1,\ldots,K} \{R_k\}, \tag{3.2}$$

  and introducing in each domain for which $R_k < R_{\max}$ the possibility of a *null event*, that is an event in which no particle moves. The rate of the null event in the k-th domain is defined as

$$r_{k0} = R_{\max} - R_k. \tag{3.3}$$

  As a consequence, the domain having the greatest relative total rate equal to $R_{\max}$ will have no null event. Introducing null events is necessary to align the interevent time for the entire system on the time of the *fastest* evolving domain: this way, the same interevent time can be chosen for all the domains as a function of only the maximum rate $R_{\max}$, and a random number. Despite the presence of null events in every of the $K - 1$ domains having $R_k < R_{\max}$, *globally* the algorithm is still rejection-free since inside the domain with $R_k = R_{\max}$ there is no null event, so that at each time step at least one molecule movement is realized.

- In each domain an event is selected out of the list of n_k events available to the k-th domain, and realized with probability $p_{ik} = r_{ik}/R_{\max}$.
- If the outcoming configurations of two or more domains conflict with each other at their shared boundary, they are subjected to a correction procedure. Section 3.3 is devoted to this topic.
- Interevent time is extracted from an exponential distribution:

$$\tau = \frac{-\ln(u)}{R_{\max}} \qquad \text{with } u \text{ random number} \in (0,1). \tag{3.4}$$

- The entire procedure is iterated until final time is reached.

3.3 Conflicting situations at the domain boundaries in discrete systems

In a continuous system a boundary conflict can arise if at a given time step the global outcoming configuration contains at least a pair of particles extremely close to each other. In discrete systems where a strict exclusion principle holds this translates to the much more likely situation where two or more particles are attempting to occupy the same lattice site.

There are basically two possible strategies for solving such a conflict: (i) the synchronous sublattice method [75] and (ii) a *rollback* procedure (see, for example, [73]). In the former every domain is further divided into sublattices having a size larger than the range of interactions. Conflicts are avoided by executing moves only in a randomly selected sublattice. In the latter instead conflicts are treated only when they occur. Indeed, rollbacks have a high computational cost. To speed up the simulation, Martínez *et al.* [74] avoided rollbacks by simply ignoring the conflicts. We have instead chosen to implement them anyway to avoid loss of synchronicity, and then to minimize their number by properly choosing the domains shape, thus compensating for the consequent slowing down of the simulation. The full time-horizon synchronicity of the domains allows the use of that procedure only when a violation of the exclusion principle occurs. In that case one proceeds as follows:

- Check for conflicting events across boundaries.

- For each conflicting pair of domains, the move to be undone is chosen through random selection of one of the two domains.

- Undo the chosen move, that is, restore the previous state of the list of rates.

- By using the same random number we have used for the realization of the conflicting event, a new move is performed [76] and the simulations goes on.

In other synchronous methods [73, 77] each domain has its own history and time. At a fixed time interval one has to check the boundary events in order to verify if the generated timeline is consistent, then correct possible problems and eventually assign the proper time to obtain synchronicity. This procedure can lead to a certain number of moves to be undone, and its implementation is rather complicated. On the contrary, the method presented in this work looks very simple since the time horizon has been set up to be flat. This permits boundary events to be communicated immediately, and the maximum number of moves to be undone at each time step to be just one. Moreover, no causality error can arise. The main drawback is the increased communication cost, but this can be minimized by properly choosing the shape and the dimension of the domains. Even though the best domain choice is problem dependent, in general the ideal shape is the one that minimizes the number of communicating domains, and the ideal size is the largest possible in order to reduce the probability of a boundary event while still benefiting from the use of multiple processors in parallel, as we will show in Section 3.4.2.

The method is not rigorous for interacting particles, where when a move happens to change the configuration at the boundaries then performing a roll-back may not lead back to the starting configuration. This changes the value of $R_{\max}$ due to the addition of a particle in the selected domain, thus introducing an error. Nevertheless, the range of values the change in $R_{\max}$ might fall in is limited, and independent of the size of the domains. Therefore a domain size can be found that minimizes such a range (e.g., enlarging the domain reduces the overall effect of the change). Moreover, such situations will happen with a relatively low frequency during the simulation if the domain size is chosen large enough so that the number of nonboundary sites is much greater than number of sites at the boundaries, thus making the effect of conflicts negligible.

As its major strength and main advantage with respect to more complicated procedures, the nonrigorous approach presented here enables the error to be easily controlled thus allowing the same results of a rigorous method to be obtained [74], but with a simpler implementation and a faster execution.

3.4 Application to a selected system: benzene in Na X

Aromatic hydrocarbons are among the crucial ingredients of many plastic and allied materials. With the increase in the prices of crude oil, there is an urgent need to reduce the processing costs of aromatics while increasing the efficiency. This makes it necessary to bring about new technologies. The proposed [78] high activity for alkylation reactions of benzene with ethylene of Faujasite (FAU)-type zeolites to make styrene, one of the most relevant industrial monomers, offers the advantage of a high selectivity toward the desired product due to the shape-selective properties of their microcrystalline pore structures. In these applications the diffusive molecular transport through zeolites needs to be described accurately for a predictive design of the processes. However, the number of fundamental studies that investigate aromatics diffusion and adsorption in porous solids is limited due to the complexity of the system, the sluggish motion of aromatics in zeolites caused by the strong interactions between π electrons and extraframework cations, and the large size of the aromatic species. Adsorption properties of aromatics in zeolites and other porous solids have been relatively less investigated, as compared to alkanes in zeolites. This is particularly true if we consider only theoretical or computational studies. Demontis *et al.* were the first who investigated diffusion of benzene in Na Y, belonging to the FAU-type zeolites (Figure 3.1a). Their simulations suggest that benzene is frequently localized near the sodium cation and the 12-ring windows [79], in excellent agreement with the neutron diffraction study of Fitch *et al.* [80]. Auerbach *et al.* [81–85] studied the jump motion of the guest benzene molecules in a lattice site model of Na Y, proving that cost-effective modeling techniques to simulate diffusive phenomena across multiple space and time scales lead to a significant gain, even if the price is loosing information at the intermediate scales. As a consequence, a multiscale modeling approach seems to be the proper choice to deal with this problem. It is our purpose in this work to test the parallel synchronous kMC method, with the aim of extending the modeling to the micro-millisecond time (and corresponding length-) scales.

3.4.1 Sequential algorithm

We applied our method to the study of benzene diffusion in Na X, belonging to the FAU-type zeolites.

The diffusion of benzene in this type of system can be represented in the framework of the rare events dynamics, since residence times are much longer than travel times between adsorption sites. This implies that kMC is best suited

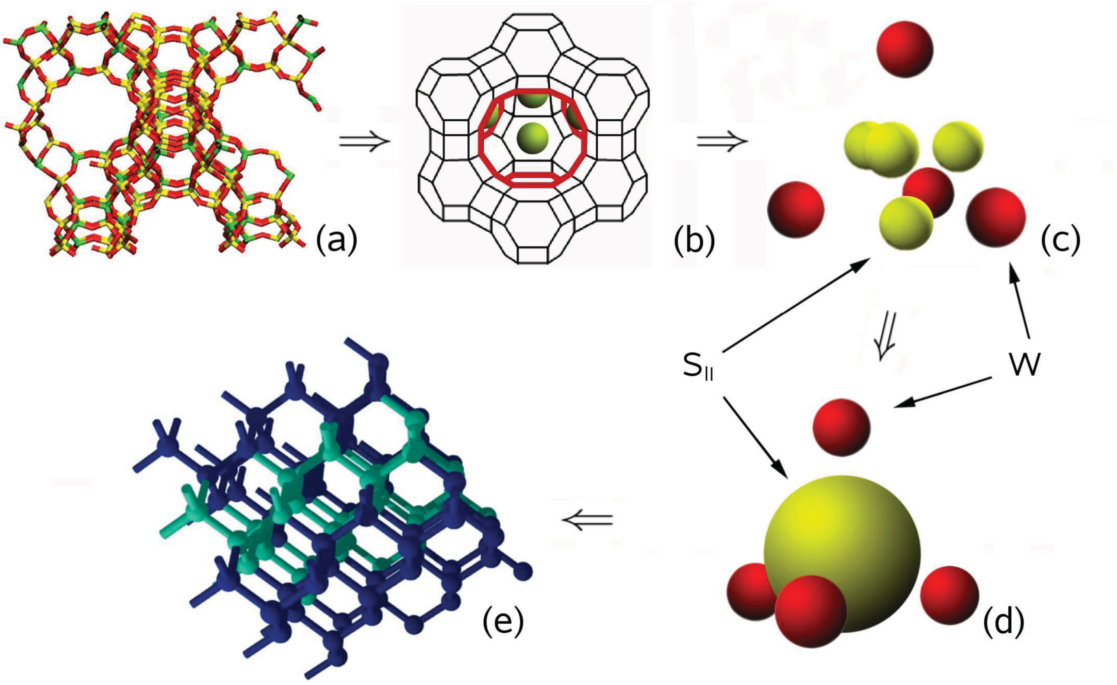


Figure 3.1: Molecular structure of FAU-type zeolites (a). The three-dimensional framework of zeolites is constitued by a network of cages (b) connected by windows. The cages can accommodate a number of guest molecules adsorbed in well-defined binding sites. In Na X zeolite there are two types of these sites: four S_{II} (yellow/light gray spheres) inside the cage and four $S_{III'}$ located in the window W connecting two cages (red/dark gray rings). In kMC simulations these sites are mapped on a detailed lattice (c) but it is possible to coarse grain the inner sites by stacking it on the center of the cage (d). Each particle can move from there to one of the four W sites (red/dark gray spheres). W to W moves are also possible. A move from S_{II} to S_{II} is possible but it produces no position change. (e) Schematic representation of zeolite FAU framework. Spheres represents coarse-grained S_{II} sites, while sticks represents W sites (for details refer to Section 3.4). Distances are proportional to the real distances among cages. Different colors correspond to different domains.

to study it.

To test the parallel algorithm we first developed a model based on a previous work on this subject [86]. The zeolite framework is represented by a three-dimensional lattice of binding sites in bi-univocal correspondence with real adsites. In the case of Na X and Na Y there are two types of sites, S_{II} located over the Na^+ cation inside the cage (Figure 3.1b) and W located on each window connecting two adjacent cages. Na Y and Na X zeolites differ in the Na content, but the same lattice can be used for both zeolites [86]. Although it is difficult to determine the exact distribution of cations in Na X, experiments show that benzene is adsorbed at the S_{II} and $S_{III'}$ sites [87] (Figure 3.1b). The latter is very close to the 12-term oxygen ring, thus permitting this site to be viewed like the

W site of zeolite Y. Assuming an Arrhenius behavior the dynamics of benzene is represented by jumps from site to site, with the rate constant calculated through considerations about the difference in energetic and geometric features of the two types of sites (see Table 3.1).

Table 3.1: Activation energies and preexponential factor at infinite dilution for benzene in Na X [86].

Jump	Activation Energy (eV)	Preexp. factor (s^{-1})
$S_{II} \to S_{II}$	0.15	0.8×10^{13}
$S_{II} \to W$	0.25	0.8×10^{13}
$W \to S_{II}$	0.10	1.1×10^{12}
$W \to W$	0.10	2.4×10^{11}

Table 3.2: Adsorption energies and entropies [86].

ϵ_W (eV)	$\epsilon_{S_{II}}$ (eV)	$\tilde{s}_W$ (eV/K)	$\tilde{s}_{S_{II}}$ (eV/K)
-0.63	-0.78	$1.7 \cdot 10^{-4}$	0

The Hamiltonian for this lattice is [86]:

$$\begin{aligned} H(\mathrm{s}, \boldsymbol{\sigma}) = & \sum_{i=1}^{M_W} s_i f_W + \frac{1}{2} \sum_{i,j}^{M_W} s_i J_{i,j}^{WW} s_j \\ & + \sum_{i=1}^{M_W} \sum_{j=1}^{M_{S_{II}}} s_i J_{i,j}^{W S_{II}} \sigma_j + \frac{1}{2} \sum_{i,j=1}^{M_{S_{II}}} \sigma_i J_{i,j}^{S_{II} S_{II}} \sigma_j \\ & + \sum_{i=1}^{M_{S_{II}}} \sigma_i f_{S_{II}} . \end{aligned} \tag{3.5}$$

In this equation s and $\boldsymbol{\sigma}$ are the number of particles adsorbed in W and S_{II} sites, respectively (occupation numbers), $f_i = \epsilon_i - T\tilde{s}_i$ (Table 3.2) is the free energy associated with the site i (ϵ_i is the adsorption energy and $\tilde{s}_i$ the entropy), J is the interaction energy between nearest neighbor particles, and $M_W = 2M_{S_{II}}$ are the number of W and S_{II} adsorption sites, respectively. It is a common choice to ignore attractive interactions between particles leading to a simple site blocking model, but in the present case this cannot be done due to the critical temperature of benzene being 560 K [88]. To account for these interactions a parabolic jump model is adopted [86, 89] where the change in the activation energy caused by the interactions is calculated as a function of the configuration in the neighboring sites. It assumes the transition state for a jump being located at the intersection of two parabolas, which is chosen to represent the minimum energy path among each pair of sites. The new value for the activation energy is obtained by [86]:

$$\begin{aligned} E_a(i,j) = E_a^{(0)}(i,j) + \Delta E_{i,j}\left(\frac{1}{2} + \frac{\delta E_{ij}^{(0)}}{k_{ij}a_{ij}^2}\right) \\ + \Delta E_{ij}^2\left(\frac{1}{2k_{ij}a_{ij}^2}\right). \end{aligned} \tag{3.6}$$

$E_a^{(0)}(i,j)$ is the activation energy at the limit of infinite dilution. ΔE_{ij} represents the variation in adsorption energy between sites i and j due to interactions: $\Delta E_{ij} = \delta E_{ij} - \delta E_{ij}^{(0)} = (E_j - E_i) - (\epsilon_j - \epsilon_i)$. In this equation $E_k = \epsilon_k + \sum_{l=1}^{M} J_{kl}n_l$ for a given configuration $\mathbf{n}$. Finally a_{ij} is the distance between two sites and k_{ij} is the harmonic force constant [86]:

$$\begin{aligned} k_{ij} = \left(\frac{2}{a_{ij}}\right)^2 \left[\frac{1}{2}\left(E_a^{(0)}(i,j) + E_a^{(0)}(j,i)\right)\right. \\ \left. + \sqrt{E_a^{(0)}(i,j)E_a^{(0)}(j,i)}\right]. \end{aligned} \tag{3.7}$$

Previous works of our group with cellular automata models applied to the study of zeolites [55–57] have shown that it is possible to *coarse-grain* space and time scales by treating adsorption sites inside a cage as one single site. This leads for large systems to improving the efficiency without losses of physical information. Application of this coarse-graining paradigm to the model presented here leads to a lattice where all the S_{II} sites competing to each cage are grouped into a multiple-occupancy site placed at the cage center. The correct time evolution is guaranteed through the use of kMC rates for all the possible jumps between the various sites making up the central multiple-occupancy site, which is the scenario for all the intra-cage motions, while every intercage move requires the passage through a W site (Figure 3.1).

Our sequential kMC was validated first by running several simulations to obtain self-diffusion coefficients to be compared with the experimental results [90]. We stress that the purpose of this comparison is to verify the correctness of the sequential algorithm and not to get new insight on the physical behavior of the system. The accomplishment of this task is postponed to a future work through the application of the parallel kMC method presented here and validated.

The simulations were carried out in a system containing 256 S_{II}-type sites and 128 W-type sites, corresponding to eight unit cells of Na X. As one can see

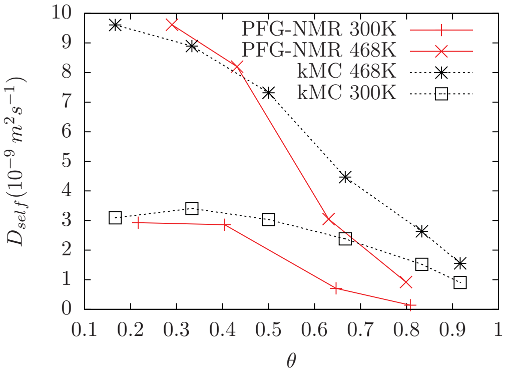


Figure 3.2: Diffusion coefficient as a function of coverage. Interaction parameter: $J = -0.03$ eV. Experimental values (red solid lines, taken from Germanus *et al.* [90]) are multiplied by a factor of 10 and are shown only for a qualitative comparison with the simulation data (black dashed lines).

in Figure 3.2 (where the self diffusivity D_{self} is plotted *vs* the coverage θ which is the number of molecules divided by the number of sites) the diffusion isotherms are in good qualitative agreement with the experimental data. The difference in the shape between the model and the experiments are expected, due to the coarse-graining of the S_{II} sites.

After that, other simulations were carried out to check the correctness of the parallel algorithm implementation. All calculation were executed on a cluster with Intel Xeon E5420 2.50 GHz processors and Infiniband communication link. For communications we made use of the MPI libraries. In Figure 3.3 diffusion

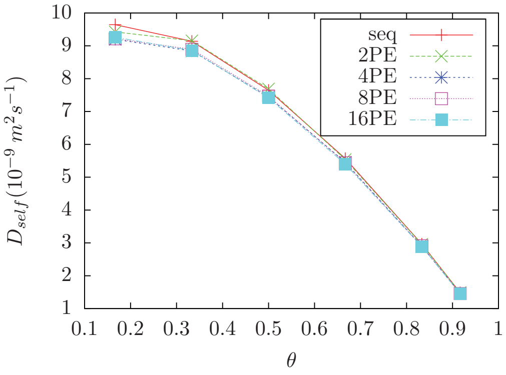


Figure 3.3: Diffusion isotherm for various number of processors on the same system size. Differences between sequential and parallel simulations are always small even for a relatively large number of processors (relative to the size of the system), and tend to converge for small numbers of processors.

isotherms obtained by simulating a system of 4096 cages with an increasing number of processors are reported. As one can see from the plot, reducing the dimension of domains causes a slight shift in the value of the diffusion coefficient. The origin of this behavior is the error introduced with the parallel algorithm that can be easily controlled by choosing appropriate dimension for the domains. The choice is strictly problem dependent and must be assessed in each case.

3.4.2 Efficiency

To determine the efficiency of the method we made use of two definitions in order to better quantify the factors involved. In the first one $\tilde{\eta}$ each processing unit involved in the parallel runs simulates a portion of the system having the same size as the system simulated in the single-processor runs. This way the definition quantifies the efficiency on the basis of the cost of communications between processing elements, holding fixed the size factor [74, 77]:

$$\tilde{\eta} = \frac{t_{S,n}}{t_{K,nK}} \cdot 100\%, \tag{3.8}$$

where $t_{S,n}$ and $t_{K,nK}$ are, respectively, the time spent in executing the serial algorithm with n particles and the time spent in executing the parallel version with K processors on a system containing nK particles. Clearly, the ideal efficiency of 100% is obtained when the time required to run the parallel code is the same as that required to execute the sequential one. This cannot be achieved in a real simulation because of the additional time required by the processors to communicate. In this particular implementation of the algorithm the major limitation is the need of global communications for updating the value of $R_{\max}$. It is important to note that despite this limitation, the impact of communication time over the global efficiency can be minimized by tuning the communication/calculation ratio. This is an easy task since almost every kMC algorithm scales with the size of the simulated system [47, 91], so that it can be achieved by just finding the optimal value for the size of each domain.

The second definition is the speedup [77]:

$$S = \frac{t_S}{t_K}, \tag{3.9}$$

where t_S is the time required to execute the serial code and t_K the time required to execute the parallel code on K processors. Here the simulated system is assumed to be exactly the same (i.e., same size and same number of iterations) for both the serial and the parallel simulation. With this definition, the speedup plot depends essentially on the scaling law the algorithm obeys [77] (in the

Table 3.3: The simulation sets considered in this work to estimate the importance of the communication/calculation ratio on the efficiency. Every subset spans several simulations at the same system size but different loadings starting from 1 ($\theta = 0.17$) up to 5.5 ($\theta = 0.92$) molecules per cage. The average number of sites per cage is six.

			Set A			
	subset 1	subset 2	subset 3	subset 4	subset 5	
N. cages	64	128	512	1024	2048	
N. processors	1	2	8	16	32	
			Set B			
	subset 1	subset 2	subset 3	subset 4	subset 5	subset 6
N. cages	512	1024	4096	8192	16384	32768
N. processors	1	2	8	16	32	64

present case the algorithm scales with the total number of particles) when the number of processors is low, and on the communication cost for higher numbers of processors.

Two sets of simulations were performed (see Table 3.3) to study systematically the behavior of the parallel algorithm, starting with executing the sequential algorithm (used as a reference), and then increasing both the domain size and the number of processors (e.g., when using two processors the system consists of two identical replicas of the reference system and so on).

All the simulations have been carried out at a temperature of 468 K and a value of -0.02 eV for J (the nearest neighbor interaction energy). As one can expect the effect of increasing the size of the domains is an improved efficiency. This is because the communication/calculation ratio decreases. The possibility of modifying the system size to change this ratio is limited by the efficiency of the sequential algorithm adopted, that is the maximum system size that one can simulate with the parallel algorithm without reducing the length of the trajectory can be estimated as K times the maximum size attainable with a standard simulation (we recall that K is the number of processors).

A determining factor of the efficiency is the domain shape. The domains must be chosen carefully on the basis of the system *topology* rather than the geometry. In the present case the node-to-node connections of the diamond-lattice topology of the FAU zeolite can be easily mapped onto a cubic grid. At this point, it is straightforward to notice that such a grid can be better partitioned into *slices* rather than cubes [73], so that every domain (i.e., every slice) does communicate with two neighboring domains instead of six (see Figure 3.4). Anyway, since the main reason of efficiency loss is the global communication caused by the need of synchronizing the domains, the choice of a cubic or a parallelepiped

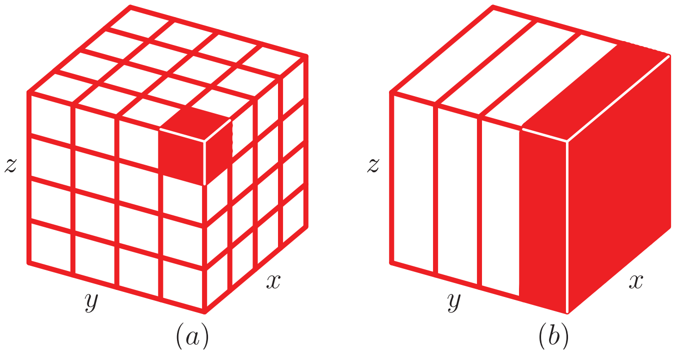


Figure 3.4: A comparison between (a) cubic domains and (b) slices. Each slice is a parallelepiped-shaped portion of the system spanning its whole extension in the y direction. This way, since with periodic boundary conditions each slice has no domain boundaries in the y direction, it does communicate with two domains only against the six of the cubic domain case.

decomposition does not significantly affect the overall value of the efficiency $\tilde{\eta}$, but the cubic shape presents as one can expect a greater number of conflicting events.

The behavior of the efficiency is similar to that of the original method [74] and some others given in the literature [77] with a fit of the form

$$\tilde{\eta} = \frac{1}{1 + a(\ln K)^b}, \tag{3.10}$$

where a and b are two constants. For the first set of simulations a ranges from 0.020 to 0.268 and b ranges from 2.177 to 4.472, while for the second set the values range from 0.038 to 0.152 and from 1.889 to 2.743 for a and b, respectively. Differences in the value of b among different simulations can be related to the different values of the communication/calculation ratio. In the A set (see Table 3.3) this ratio is greater than in the B set because domains are smaller, whereas the information exchanged between domains is the same (with a fixed number of processors), therefore the efficiency decays more steeply. Within each set the efficiency is influenced also by the communication/calculation ratio, which in this case, however, results from the combination of two opposite effects depending on the change of the total number of molecules adsorbed in the system. By increasing that number there can be more moves between domains that require more information exchanges and more rollbacks, thus increasing the ratio. On the other hand, increasing the number of molecules leads the number of events to increase as well, requiring then more computation. The balance between the different weights of these two effects causes the value of the parameters a and b to fluctuate. Moreover, these values differ from that obtained by Martínez and Merrick [74, 77] mainly because of technical and algorithmic differences.

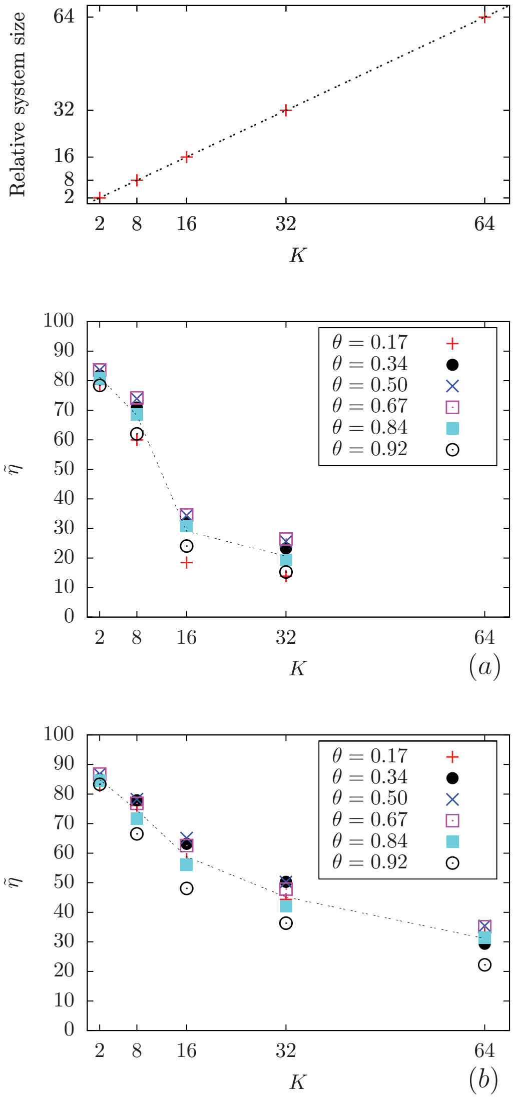


Figure 3.5: Parallel efficiency for simulation set (a) A and (b) B as a function of the number of processors at different loadings. For the parallel runs only, the system size increases linearly with the number of processors. On the top plot the ratio between the system size in the parallel runs and the system size in the serial, single processor run is shown. The ideal efficiency of 100% would be obtained only if the time required by the single-processor simulation of a system of a given size were the same as the time required by a parallel simulation on K processors of a system K times larger. Best results are obtained in set B because of the more favorable communication/calculation ratio. Dashed lines have been drawn to guide the eye.

In Figure 3.5a the parallel efficiency $\tilde{\eta}$ (Equation 3.10) is reported as a function of the number of processors (K) for the simulation set A. A comparison with Figure 3.5b, where the same data are reported for the set B, makes clear the importance of the communication/calculation ratio which favors the B-set simulations (where the computation is much more expensive than in the A set). This leads to a greater efficiency of the algorithm when applied to set B in all the cases studied here. As expected, differences in the efficiency are more pronounced for numbers of processors greater than eight, since the increased cost of communication is not compensated by an equal increase in the computation cost. Finally, in Figure 3.6 the speedup is reported. As stated before, its value is determined by a combination of two contributions, the cost of communications and the scaling law of the algorithm implemented. If we do not take into account the communication cost we would get the same computing time for both the serial and the parallel algorithm only if the algorithm were *not* scaling with the system size. Therefore, the parallel implementation of our algorithm gives a substantial gain in the execution time, for example, a simulation of the set B3 (Table 3.3) requires 59 h when using the serial algorithm and 8 h when using the parallel one on eight processors.

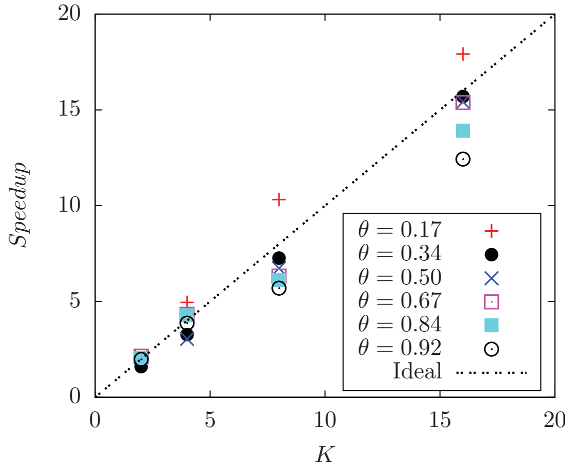


Figure 3.6: Speedup S [defined in Equation (3.9)] of the parallel algorithm with respect to the sequential one shown as a function of the number of processors. This plot refers to the B3 system (Table 3.3), other simulations show similar behavior and were not reported.

The only factor that may reduce the speedup is the number of rollbacks, since each roughly doubles the time spent for the current cycle. Anyway, this does not represent a problem in the present case where the number of rollbacks is kept relatively low (Figure 3.7) by the particular topology of the system. As a consequence, its influence over the efficiency of the method is limited and we obtain speedup values really close to the ideal efficiency of 100% (dashed line in Figure 3.6, however we remark that the speedup is expected to decrease for a very large number of processors).

The number of null events is small as well (Figure 3.7), and can be controlled

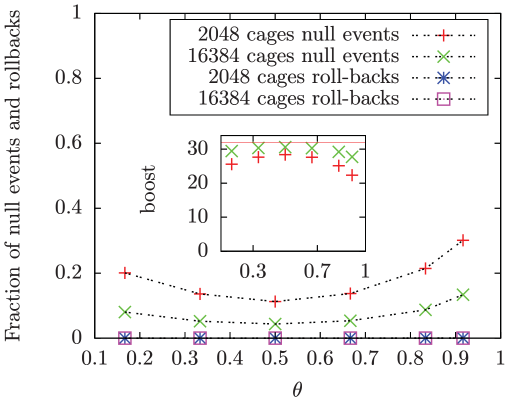


Figure 3.7: Fraction of null events and rollbacks for parallel simulations of systems A5 and B5 (Table 3.3) on 32 processing units. The fraction of null events decreases with increasing the system size (leading to a 90% of effective moves), whereas the fraction of rollbacks is always negligible, even for a small system. Discrepancies among different coverages reflect the change in the relative number of possible events. Other simulations showing analogous behavior are not reported. The inset shows the average number of moves realized for every kMC step. The maximum of those values equals the number of processors, which is 32 in this case.

through a proper choice of the system size. Consequently, during a simulation of set B approximately the 90% of the possible moves are performed with no need of redefining the domain shape. The boost in the number of events per cycle is reported in the inset of Figure 3.7.

3.5 Conclusions

A parallel kinetic Monte Carlo algorithm, originating from the synchronous algorithm of Martínez *et al.* [74], has been applied to the study of benzene diffusion in zeolite Na X. We have shown that, despite the presence of a rollback procedure in the algorithm, high efficiencies can be reached by exploiting the local nature of the molecule-molecule interactions inside the zeolite, allowing the need of rollbacks to be minimized through a proper spatial decomposition. In the present form the algorithm is still approximate, but the correct tuning of the domains size leads to obtaining results with the desired accuracy. We believe that the algorithm outlined here is applicable in general with little modification to other types of zeolites. Even better performances are expected to be found for other zeolites like the Linde Type A (LTA) family, ZSM5 [92], or for zeolitic imidazolate frameworks (ZIF) [67] because of the absence of shared sites between communicating cages. Adsorbate-adsorbate interactions does not extend significantly outside the cages, thus permitting an ideal domain decomposition. As for other similar methods [74,77], the efficiency of the algorithm is very sensitive to the value of the communication/calculation ratio that can be easily controlled by changing the size or the shape of the domains.

Chapter 4

The Central Cell Model: A mesoscopic hopping model for the study of the displacement autocorrelation function

Adapted with permission from F. G. Pazzona, A. Gabrieli, A. M. Pintus, P. Demontis and G. B. Suffritti; The Journal of Chemical Physics; Vol. 134, Page 184109 (2011). "Copyright 2011, American Institute of Physics."

http://dx.doi.org/10.1063/1.3587618

The diffusive motion of molecules in a generic medium is usually affected by memory effects introduced by their interactions with each other and with the medium itself. This is especially true when the diffusing molecules are subjected to the confining action of a microporous material such as a zeolite [62, 93]. In particular, the narrow windows of certain microporous materials can make the guest's diffusion profile (i.e., diffusivity *vs.* concentration at constant temperature) very different from what expected for the motion in a bulk phase as well as in any less strongly confining material.

Although the discreteness of the network of channels and cages of regular microporous materials suggests immediately an analogy with lattice-gas models, there is still no "definitive" coarse-grained, lattice simulation method for molecules in zeolites which is able to play as a *cheaper* mesoscale version of classical molecular dynamics (MD). Several approaches are available depending on what specific properties of the host-guest system the simulator is interested in. As an example, kinetic Monte Carlo simulations are suitable for all the dynamical properties which do not explicitly involve correlations among different particles [47, 94, 95] (e.g., the self diffusion coefficient), whereas thermodynamic models can be successfully adopted for the study of static equilibrium properties (e.g., adsorption isotherm and local density distribution).

Due to their intrinsically synchronous nature, the class of lattice-gas cellular automata (LGCA) can be thought of as the ideal candidate for a mesoscopic simulation of the collective properties. On the other hand, as a drawback of their synchronicity traditional LGCAs are much more difficult to handle than standard Monte Carlo (MC) models are. This makes it a hard task to *surely* achieve thermodynamic equilibrium, i.e., preserving both detailed balance and synchronicity, in the presence of explicit particle-particle interactions. To solve such a conflict, a partitioning technique has been proposed in our previous work, aimed to couple the LGCA computational framework with local MC (balanced) moves [56–61]. The idea underlying the resulting partitioning cellular automaton (PCA), inspired by a heterogeneous model for surface diffusion by Chvoj *et al.* [96], is that the peculiar cage-to-cage dynamics of molecules under tight confinement is well-represented in a model lattice with heterogeneous adsorption locations inside each cage. According to this representation, in each zeolite cage we distinguish two types of locations: those close to the exit windows, termed *exit sites*, and the rest of the cage pictured instead as a set of *inner sites*. The exit sites in each cage are then access points to the neighboring cages, and differ from the inner sites in their statistical weight (i.e., the probability of being occupied). As recently confirmed by other simulation studies [97], splitting the single cells into differently weighted locations provides a qualitatively correct mesoscopic representation of the problem (See Figure 4.1).

Even though more work has still to be done to make cellular automata the “definitive” environment for meso-simulations in micropores, our PCA approach captures many important aspects of adsorption and diffusion in zeolites, such as realistic (i.e., closely resembling those developed in MD simulations) density distribution, fluctuations, and *time correlations*. Concerning the single-particle diffusion process (at arbitrary concentration), the *backscattering effect* [28], a major source of time correlation causing the self-diffusivity to be less than what expected, can be properly mimicked in the PCA approach since it allows the amount of memory lost in each cell during a single time step to be tuned.

Thus, our PCA can be taken as a starting point for further developments in many directions. The one explored in this work is the realization of a further simplified coarse-grained simulation of the hopping process of a tagged particle in a confined lattice system, where all the other guest particles are moving as well but they are kept indistinguishable. Our aim is to reproduce the memory effects affecting the particle motion in the PCA at the minimum cost possible. The strategy is to make the tagged particle “feel” an environment very close to

Table 4.1: A list of the basic quantities involved in a numerical simulation with the Central Cell Model.

μ	chemical potential
β	inverse temperature
$K_{\text{ex}}, K_{\text{in}}, K$	exit, inner, and total site number per cell
s	micro-configuration of indistinguishable particles in a single cell
$n_{\text{ex}}, n_{\text{in}}, n$	exit sites, inner sites, and total occupancy of a single cell
$\mathbf{n} = (n_{\text{ex}}, n_{\text{in}})$	meso-configuration of the cell
$f^o_{\text{ex}}, f^o_{\text{in}}$	exit and inner site free-energy of adsorption in a singly-occupied cell (site deepness)
$\phi_{\text{ex}}(n), \phi_{\text{in}}(n)$	exit and inner site free-energy contribution due to the mutual interaction of n particles
$F(\mathbf{n}), F(\mathbf{s})$	cell free energy
$F^o(\mathbf{n}), F^o(\mathbf{s})$	cell free energy (non-interacting part)
$\Phi(\mathbf{n}), \Phi(\mathbf{s})$	cell free energy (interacting part)
$\epsilon_{\text{ki}}(n, m)$	kinetic barrier to intercell migration from an n- into an m-occupied cell
C_{ab}	probability of targeting the site b from departure site a during randomization
p_{jump}	acceptance probability for a single randomization jump
$\overline{s}_b\kappa(\mathbf{n}, \mathbf{m})$	acceptance probability for a jump from a cell with meso-conf. $\mathbf{n}$ into exit site b of a cell with meso-conf. $\mathbf{m}$
$p(\mathbf{n})$	equilibrium probability of a cell to be meso-configured as $\mathbf{n}$

the one it would have experienced in the full automaton simulation. Since the model is constructed in such a way that the host cell of the tagged particle always results to be located exactly in the middle of the system, we called it *Central Cell Model* (CCM).

The lengthy PCA simulation of a large system is thus reduced to a small set of connected cells, a limited neighborhood of whose is simulated by the lattice-gas evolution rule in the canonical ensemble while the border cells are treated as mean-field cells. In any case, the CCM approach cannot be taken as substitutive of a full lattice-gas simulations. Collective dynamic properties, self-organization, and long-range phenomena arising in non-equilibrium conditions cannot be simulated directly through a CCM implementation of a lattice-gas rule. This approach is limited to the reproduction of the correlated motion of a *single* particle in a lattice-gas at arbitrary loading (i.e., concentrations of guest particles, also known as *coverage*), but under conditions of thermodynamic equilibrium, strictly local interactions, and absence of long-range correlations. When one or more of such conditions are not fulfilled or if also the collective dynamics produced by some evolution rule need to be investigated, then a full lattice-gas simulation will be unavoidable to obtain reliable results. Nevertheless, the above mentioned conditions are fulfilled in many lattice-gas simulations of short-range interacting particles, so that for those cases the CCM will be the quickest way to retrieve the correct self-motion properties. This is of primary interest when, for example, one wishes to model the entity of memory effects in the single-particle motion (e.g., to mimic the diffusive behavior of some reference system in coarse-grained modeling) and therefore needs to check quickly how a particular setup of the parameters will affect the resulting diffusion isotherm.

The construction of the CCM version of a lattice-gas rule is a really direct way to uncover the basic mechanisms by means of which the tagged particle preserves memory of its previous moves in time. In fact, it is straightforward to pass from the numerical CCM to a mean-field representation of the tagged particle's diffusion process at arbitrary loading, inclusive of the time correlations. In this work, the CCM approach will be used to develop an approximated theory of self-diffusion for a lattice-gas automaton rule, based on a mathematical formulation of the displacement autocorrelation function (DACF), i.e., the key function embedding the memory effects of a generic diffusion process on the mesoscopic scale. The DACF plays the same role the velocity autocorrelation function (VACF) plays in atomistic simulations, although being more easily accessible for theoretical analysis. Earlier studies on LGCA emphasized the central importance of the discrete VACF in both the formulation of efficient computational schemes for the evaluation of transport properties and the understanding of the entire self-diffusion process [98–100]. In the present case, the analysis of the DACF (we do not call it VACF since, differently from traditional LGCAs, in our approach no proper *velocity* vector is associated with the cell-to-cell migration) will lead to a closed mathematical formulation for the self-diffusion coefficient.

After a brief resumé of the lattice-gas model, the Central Cell Model will be presented. Then, we will describe the probabilistic analysis of the DACF leading to the mean-field formulation of self-diffusivity. Results of numerical tests will be presented throughout the chapter and discussed in a separated section.

4.1 Local randomization and propagation

Here we will briefly outline the basic operations of the original automaton model. The interested reader can find a very detailed description in a previous work on this subject [58, 60]. The basic quantities that will be explicitly used in a simulation with the Central Cell Model are listed in Table 4.1.

In our approach, particles move within a three-dimensional network of structured points called *cells*. A single cell and a small cluster of connected cells of the automaton are sketched in Figure 4.1a and 4.1b respectively. The total number of particles in the system, N, and the temperature, T (and so the inverse temperature, $\beta = (k_B T)^{-1}$ with k_B the Boltzmann constant), are held fixed. The concentration $\langle n \rangle$ of the diffusing species in the lattice, termed *loading*, is the average number of particles per cell and is obtained just by dividing N by the total number of cells. Every cell is a discrete representation of a zeolite cage. It is made of K_{ex} exit sites and K_{in} inner sites, and every site can be free or singly occupied, thus giving a saturation occupancy of $K = K_{\text{ex}} + K_{\text{in}}$. As can be seen from Figure 4.1b, every pair of neighboring cages are interfaced by a pair of connected exit sites. The system evolves in discrete time steps. Guest molecular species are represented via point particles whose migration mechanism at each iteration is performed in two substeps: a *randomization* changes the configuration of guest particles on every cell according to a probabilistic scheme, and a *propagation* allows the particles in the exit sites to attempt to move into the respective neighboring cages.

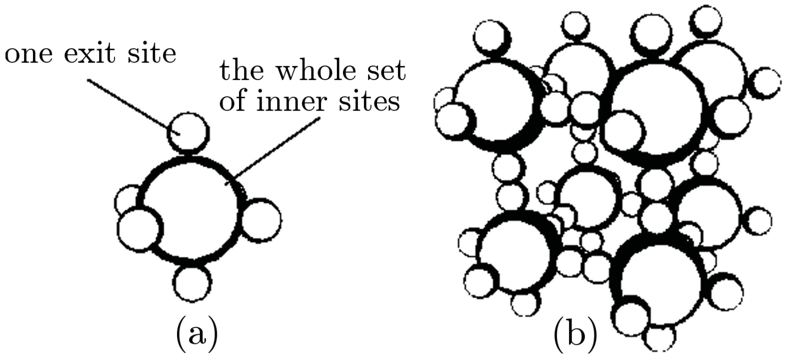


Figure 4.1: A three-dimensional sketch of (a) a single cell, and (b) a small cluster of connected cells of the automaton. Every cell is representative of a single zeolite cage. When looking at the single cell, (a), small spheres represent the *exit sites*, i.e., the locations closest to the cage-to-cage connections in a real zeolite (e.g., an LTA zeolite), whereas the big sphere, named *inner site*, represents the set of all the remaining locations.

The actual micro-configuration of (indistinguishable) particles in each cell has a primary importance and is denoted as

$$\mathbf{s} = \{s_1, s_2, \ldots, s_K\}, \tag{4.1}$$

where the first K_{ex} and the next K_{in} entries are respectively the occupancies of the exit and of the inner sites (i.e., $s_i = 1$ if the i-th site of the cell is occupied, and 0 if empty). The cell occupancies are defined as the *exit site*, the *inner site*, and the *total* cell occupancies:

$$n_{\text{ex}} = \sum_{i=1}^{K_{\text{ex}}} s_i, \qquad n_{\text{in}} = \sum_{i=K_{\text{ex}}+1}^{K} s_i, \qquad n = n_{\text{ex}} + n_{\text{in}}. \tag{4.2}$$

Exit and inner site cell occupancies make a *meso-configuration* of the cell, termed $\mathbf{n} = (n_{\text{ex}}, n_{\text{in}})$.

The static properties of each cell are determined by the adsorption (negative) free energy associated to every site, f^o_{ex} and f^o_{in} (also referred to as exit- and inner-site *deepness*), the actual cell occupancy n (i.e., the total number of particles in the cell), and an occupancy-dependent interaction term for every type of site, $\phi_{\text{ex}}(n)$, and $\phi_{\text{in}}(n)$. These parameters define the cell free energy function:

$$F(\mathbf{n}) = F^o(\mathbf{n}) + \Phi(\mathbf{n}), \tag{4.3}$$

with

$$F^o(\mathbf{n}) = n_{\text{ex}} f^o_{\text{ex}} + n_{\text{in}} f^o_{\text{in}}, \tag{4.4}$$

and

$$\Phi(\mathbf{n}) = n_{\text{ex}} \phi_{\text{ex}}(n) + n_{\text{in}} \phi_{\text{in}}(n). \tag{4.5}$$

In the numerical simulation we performed as a test for the model, our choice for the interaction parameters was $\phi_{\text{ex}}(n) = \phi_{\text{in}}(n) = \phi(n)$, with a smoothly increasing repulsive effect as the occupancy increases (see Figure 4.2). We set the number of exit and inner sites as $K_{\text{ex}} = K_{\text{in}} = 6$. Fixed adsorption free-energy f^o_{ex} and f^o_{in} take alternatively the values -50 and -40 kJ mol^{-1} in the various simulations.

4.1.1 Randomization

The randomization can be performed in two ways. The *memoryless randomization* changes probabilistically the actual configuration of every cell while preserving its occupancy according to the probability distribution $P(n_{\text{ex}}|n)$ defined as

$$P(n_{\text{ex}}|n) = \binom{K_{\text{ex}}}{n_{\text{ex}}} \binom{K_{\text{in}}}{n - n_{\text{ex}}} e^{-\beta F(n_{\text{ex}}, n - n_{\text{ex}})}. \tag{4.6}$$

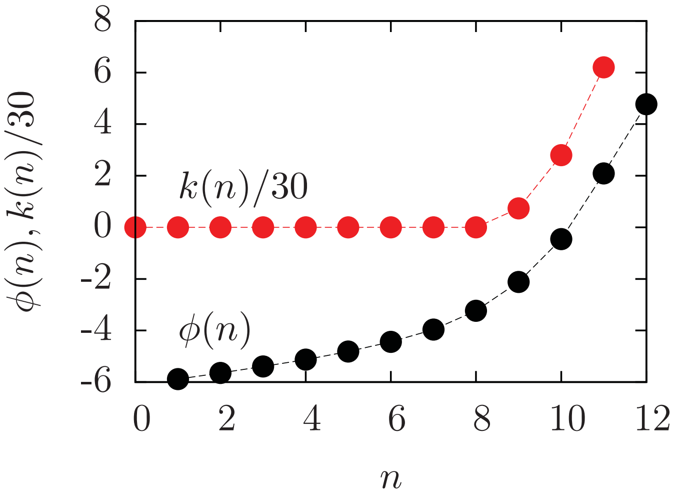


Figure 4.2: The interaction parameter, $\phi(n)$ (in kJ mol^{-1}, defined in Section 4.1), for $0 \leq n \leq K - 1$, and the parameter $k(n)$ (defined in Section 4.1.2) for the numerical tests we performed in this work. In all the simulations, $K_{\text{ex}} = K_{\text{in}} = 6$.

which is exactly the probability of an n-occupied cell to have n_{ex} particles in the exit sites (and consequently $n_{\text{in}} = n - n_{\text{ex}}$ in the inner sites). Such a choice causes no memory of the previous configuration(s) to be conserved (apart from the cell occupancy n, which is conserved).

In the *jump randomization* scheme instead all the n particles are invoked in a random sequence and every particle is asked to perform a jump toward a randomly selected target site within the same cell. Therefore, the cell configuration is changed here in n steps instead of one (as it was for the memoryless scheme). To illustrate the algorithm, let us take a single cell and store the identities of the n particles inside of it in the vector $\mathbf{I} = (I_1, \ldots, I_n)$. Let us then randomize the entries of $\mathbf{I}$, thus obtaining the random sequence of identities $\mathbf{I}^R = (I_1^R, \ldots, I_n^R)$. At this point, the following chain of jump events is realized:

$$\mathrm{s}^{(0)} \rightarrow \mathrm{s}^{(1)} \rightarrow \mathrm{s}^{(2)} \rightarrow \cdots \rightarrow \mathrm{s}^{(n)}, \tag{4.7}$$

where by definition $\mathrm{s}^{(0)} := \mathrm{s}$ is the first configuration of the chain, and $\mathrm{s}^{(k)}$ is the actual micro-configuration when the particle of identity I_k^R is invoked. Let us consider a transition $\mathrm{s}^{(k)} \rightarrow \mathrm{s}^{(k+1)}$ where $\mathrm{s}^{(k)}$ and $\mathrm{s}^{(k+1)}$ are two consecutive configurations in the chain (4.7). In this transition, the k-th particle in the random sequence of particles jumps from its departure site, say a, to the target site b chosen with a probability C_{ab}. The probability of such a jump to happen is then

$$p_{\text{jump}}(\mathbf{s}^{(k)} \to \mathbf{s}^{(k+1)}) = C_{ab}\overline{s}_b^{(k)}\gamma e^{\beta f_a^o} \times e^{\beta\left\{\Phi(\mathbf{s}^{(k)})-\max\left[\Phi\left(\mathbf{s}^{(k)}\right),\Phi\left(\mathbf{s}^{(k+1)}\right)\right]\right\}}, \tag{4.8}$$

where $\overline{s}_b^{(k)}$ is the *non*-occupancy of the target site b in the actual micro-configuration $\mathbf{s}^{(k)}$, i.e., $\overline{s}_b^{(k)} = 1 - s_b^{(k)}$, and γ is a normalization constant aimed to further control the particles' mobility during randomization (this will affect correlations as well). In our simulations, we put $\gamma = \exp\{-\beta \max(f_{\text{ex}}^o, f_{\text{in}}^o)\}$. Such an algorithm preserves some memory of the previous configuration, since in the case of half/high cell occupancy n, the (locally) sequential jump criterion constrains the configuration not to vary too much in the chain shown in (4.7).

A few words about the choice for C_{ab}. In order to preserve detailed balance, it preferably should be symmetric, that is forward and reverse jumps should be chosen with the same probability. It is interesting to introduce several kinds of constraints (without violating symmetry) in the configuration path during randomization, to study their effects on correlations, and to check to which extent they can be predicted by a mean-field theory of diffusion. As an example we could decide, during randomization, to allow every particle to target any site with the same probability $1/K$, this giving a C matrix with all entries such as

$$C'_{ab} = \frac{1}{K}, \qquad a, b \in [1, K], \tag{4.9}$$

or we could choose all targetings from an exit site toward a different exit site to be rejected. This would force the particles to spend some time in the inner site before changing direction of intercell migration. It would result in a C matrix such as

$$C''_{ab} = \begin{cases} 0, & \text{if } a, b \in [1, K_{\text{ex}}] \text{ and } a \neq b \\ \dfrac{1}{K}, & \text{otherwise.} \end{cases} \tag{4.10}$$

In the present work we will refer to the case of $C = C'$ in Equation (4.9) as "allowed ex-ex jumps", and to the case of $C = C''$ in Equation (4.10) as "forbidden ex-ex jumps".

4.1.2 Propagation

Once randomization changed the internal configuration of every cell independently one of the other (while preserving the cell occupancies), the propagation operation allows the cells to exchange the particles in their exit sites with their respective neighbors. In order to keep working with locally balanced Monte Carlo moves the propagation must be applied to every *pair* of communicating cells. Since some pairs can overlap, not all the pairs can be invoked at the same time. This is because of local interactions among the host-molecules of a given cell giving rise to different intercell migration barriers, depending on the loading of both departure and target cell. Therefore, either they have to be invoked in a random sequence, or they can be grouped into *partitions*, each containing the maximum possible number of non-overlapping pairs. Such a partitioning scheme [58], originally known as Margolus' Neighborhood [50, 54] allows no conflict to arise during such a substep.

At every pair, the two cells communicate through two adjacent exit sites, say a and b. Provided a particle to be in a and site b to be empty, a jump from a to b is accepted with a probability $\kappa(\mathbf{n}, \mathbf{m})$ where the departure and destination cell are meso-configured, respectively, as $\mathbf{n}$ and $\mathbf{m}$:

$$\kappa(\mathbf{n}, \mathbf{m}) = \frac{\gamma e^{\beta f^{o}_{\text{ex}}} e^{-\beta \epsilon_{\text{ki}}(n,m)}}{1 + e^{\beta \Delta\Phi(\mathbf{n},\mathbf{m})}}, \tag{4.11}$$

where $n = n_{\text{ex}} + n_{\text{in}}$ and $m = m_{\text{ex}} + m_{\text{in}}$ are the actual occupancies of the departure and the target cell, respectively, the quantity

$$\begin{aligned} \Delta\Phi(\mathbf{n}, \mathbf{m}) = & \Phi(n_{\text{ex}} - 1, n_{\text{in}}) + \Phi(m_{\text{ex}} + 1, m_{\text{in}}) \\ & - \Phi(n_{\text{ex}}, n_{\text{in}}) - \Phi(m_{\text{ex}}, m_{\text{in}}) \end{aligned} \tag{4.12}$$

is the difference in interaction free-energy between the outcoming and the incoming configuration of the pair of cells and $\epsilon_{\text{ki}}(n, m)$ is the kinetic barrier to intercell migration, given as the intersection energy, for $0 \leq x \leq 1$, between the two harmonics

$$E_{\text{dep}}(x) \quad = \tfrac{1}{2} k(n-1) x^2 \tag{4.13}$$

for the departure cell, and

$$E_{\text{arr}}(x) \quad = \tfrac{1}{2} k(m) (x-1)^2 \tag{4.14}$$

for the arrival cell [59]. The trend assigned to the parameter $k(n)$ in the numerical simulation performed in this work is quadratically increasing at the highest loading, as shown in Figure 4.2.

4.2 Jumps and time correlations

Numerical simulations [60] have shown that correlation effects can be modeled (or excluded, if wanted) in our PCA. While every application of the memoryless randomization described in Section 4.1 pushes each cell straightforwardly toward a condition of local equilibrium, via an abrupt collective move, the configuration changes occurring by means of the jump randomization are much less marked, and slow down strongly the evolution toward equilibrium. This is because the output configurations available in the jump randomization are *much less* than in the memoryless randomization, thus causing memory effects to show up spontaneously as the system evolves in time.

Let us illustrate this in more details. The definition of *configuration*, s, given in Equation (4.1) in Section 4.1 contains no information regarding the identity of the guest particles. In other words, such a kind of identity-less configuration will be referred to as "s-configuration".

Particles identities will be taken into account by the following σ-configuration instead:

$$\boldsymbol{\sigma}(\mathbf{r}) = \{\sigma_{iI}\}, \qquad i = 1, \ldots, K \text{ and } I = 1, \ldots, N \tag{4.15}$$

where N is the number of guests, and σ_{iI} has value 1 if the I-th guest of the system is located at the i-th site of cell $\mathbf{r}$, and 0 otherwise. We will consider now a single, closed cell with configuration $\boldsymbol{\sigma}$ just *before* a randomization operation. The *memoryless* randomization will determine the output configuration by choosing it out of the whole set of $\Omega_{\boldsymbol{\sigma}} = K!/(K-n)!$ possible arrangements of distinguishable particles in the cell. The *jump* randomization instead constrains the configuration path from $\boldsymbol{\sigma}$ to $\boldsymbol{\sigma}^R$ within a set of necessarily similar configurations, so that the number of possible output configurations, say $\Omega_{\boldsymbol{\sigma}}^{\text{jump}}$, results smaller than $\Omega_{\boldsymbol{\sigma}}$ with a discrepancy increasing as the cell occupancy increases. Such a discrepancy is the very origin of the memory effects in the self-diffusivity in the automaton [60] as well as in a host-guest system in general. Ideally, one should perform an infinite number of *jump* randomization cycles per time step in order to suppress it.

An analysis of the migration mechanism in the automaton will help a deeper understanding of the correlations introduced by the jump randomization. A low-cost study of correlations in the motion of a tagged particle induced by the local environment is the task of the Central Cell Model that we are about to introduce in Section 4.3 for the case of a discrete jump model.

4.3 The Central Cell Model

In the model we present here, the lattice is constituted by (see Figure 4.3)

(i) A *central cell*.

(ii) A finite number, N^{sh}, of cells surrounding the central one, organized into

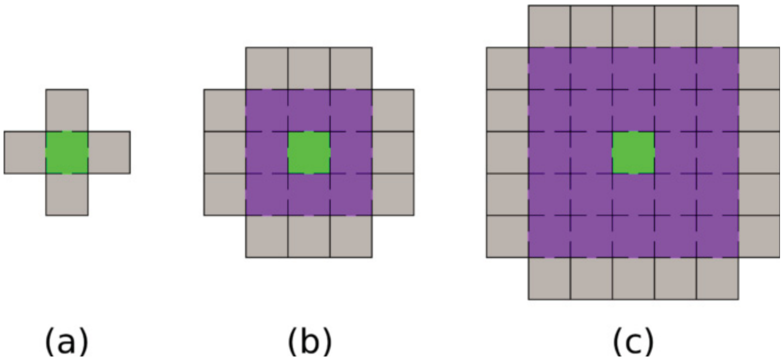


Figure 4.3: The lattice space of the Central Cell Model. The central cell (in green), hosting the tagged particle, and the cells in the core shells (violet) are simulated through the prescribed lattice-gas rule in the canonical ensemble. The external cells (gray), instead, are mean-field. They maintain the whole system at thermodynamic equilibrium and work as a reservoir of particles coming in/out of the border core cells. Broken cell-to-cell boundaries are meant as cell-to-cell links. Figures (a), (b), and (c) differ in the number of core shells, which is $L^{\text{sh}} = 0$ in (a), $L^{\text{sh}} = 1$ in (b), and $L^{\text{sh}} = 2$ in (c).

L^{sh} *shells*. Central cell and surrounding shells constitute the *core* of the system.

(iii) A casing of N^{mf} border *mean-field* cells enclosing the core. Mean-field cells are small grand-canonical systems, working for the core cells as a reservoir of particles and keeping the whole system in equilibrium at the desired value of chemical potential.

The cell-to-cell connections are established as follows: every cell in the core is connected with all the available first-neighboring cells in the system, so that if we consider a cubical arrangement of cells (so as to mimic the LTA zeolite topology, as an example) every cell of the core cells is then connected to six first neighbors. Core cells need not to be connected with each other only: cells at the borders of the core happen to have one or more mean-field cells in their neighboring list. Every cell of the mean-field cells instead are supposed to be connected with one cell at the border of the core only. No connection is assumed to exist between mean-field cells.

Since the mean-field cells exchange particles with an ideal reservoir, a chemical potential, μ, has to be selected first. This gives access to the absolute probability, $p(\mathbf{n})$, of a meso-configuration $\mathbf{n}$ defined as

$$p(\mathbf{n}) = [\Xi(\mu)]^{-1} \binom{K_{\text{ex}}}{n_{\text{ex}}} \binom{K_{\text{in}}}{n_{\text{in}}} e^{\beta\mu n} e^{-\beta F(\mathbf{n})}, \tag{4.16}$$

where the normalization factor $\Xi(\mu)$ is the grand-canonical partition function of a single cell:

$$\Xi(\mu) = \sum_{\mathbf{n}} \binom{K_{\text{ex}}}{n_{\text{ex}}} \binom{K_{\text{in}}}{n_{\text{in}}} e^{\beta\mu n} e^{-\beta F(\mathbf{n})}. \tag{4.17}$$

Occupancies n_{ex}, n_{in} and n in Equations (4.16) and (4.17) are meant as the occupancies of the exit sites, the inner sites, and the whole cell, respectively, when the meso-configuration is $\mathbf{n}$. Such a notation will be used throughout the whole chapter.

The average occupancy (often referred to as the *loading*) is then $\langle n \rangle = \sum_{\mathbf{n}} np(\mathbf{n})$.
The probability distribution in Equation (4.16) will be used to update the state of the mean-field cells at each time iteration.

Generating the initial configuration. The initial configuration is constructed by randomly assigning each cell a meso-configuration according to the distribution $p(\mathbf{n})$ (see Equation (4.16)). Such a meso-configuration is then converted into a micro-configuration s of indistinguishable particles, randomly chosen out of those satisfying the meso-configuration itself. Whereas not needed by the other cells, the central cell must contain at least one particle, that will be "tagged" thus allowing us to follow its dynamical path.

Time evolution

Once the initial configuration of the system is ready, the system evolves in discrete time steps, $t_0, t_0 + \tau, t_0 + 2\tau, \dots,$ each of physical duration τ (see Appendix A.1 and our previous work [58] for a discussion about the time step). At each time step (say, t):

(i) A jump randomization is performed at each cell.

(ii) The pairs of connected cells are chosen in a random sequence, and a propagation operation is performed at every pair. Until now, the whole lattice has preserved its total number of particles.

(iii) The move performed by the tagged particle is stored. If it has left the central cell, then the system has to be re-centered so that the newly occupied cell becomes the central cell. Such an operation is performed by simply transforming the coordinates of all the cells. If the tagged particle made a cell-to-cell jump, then the coordinates of the cells are transformed as follows:

$$\mathbf{r}(t+\tau) = \mathbf{r}(t) - \delta\mathbf{r}(t), \tag{4.18}$$

where $\delta\mathbf{r}(t)$ is the distance vector between the arrival and the departure cell. Due to this operation, the mean-field cells happening to fall outside of the lattice space are destroyed, whereas those resulting not configured at all will be assigned a new configuration in the next operation.

(iv) The mean-field cells are randomly assigned a new micro-configuration according to the same procedure of generation of the initial one (applied to the mean-field cells only though).

The update strategy described above is sketched in Figure 4.4. In Figure 4.5 we compare the self-diffusivity resulting from a simulation of the canonical $9 \times 9 \times 9$ lattice-gas with the one computed from a CCM simulation on an increasing number L^{sh} of shells around the central cell. We can clearly see that increasing L^{sh} improves the matching between the two types of simulations, and that two shells are enough to obtain a reasonable agreement.

4.4 Analysis of the self-diffusion process: the displacement autocorrelation function

The mean-field analysis is carried on in terms of the possible jump sequences a tagged guest can perform during the diffusion process, treated as a Markov chain, where jumps are meant as site-to-site migrations and can be categorized into (i) jumps within the same cell and (ii) jumps between neighboring cells. Each jump category has a certain probability to occur which is *dependent* on the actual position of the guest itself and of the surrounding particles. Due to the complexity of such a multi-body problem, a mean-field approach must be used to derive readable equations linking correlations in the self-motion to some macroscopic quantities (e.g., densities, total transfer rates, etc.).

A dynamical quantity of major importance in the analysis of the diffusion process in a mesoscopic lattice is the *instantaneous cell-to-cell displacement*, $\delta\mathbf{r}(t)$, of the tagged guest at time t, introduced in Equation (4.18). The instantaneous displacement can take values in the set of lattice vectors $\{\mathbf{e}_j\}$, with $j = 0, \ldots, K_{\mathrm{ex}}$, listed in Table 4.2 for the case (considered in this work) of a cubic lattice.

The displacement autocorrelation function (DACF), given by $\langle \delta\mathbf{r}(z\tau) \cdot \delta\mathbf{r}(0) \rangle$ (where $z \geq 0$ is an integer and τ is the duration of a time step), correlates in

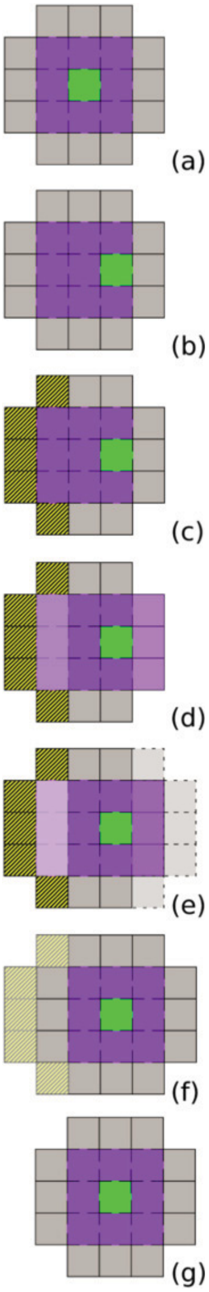


Figure 4.4: A schematic representation of the update strategy when the particle leaves the central cell to move in the right neighboring cell (a,b). In (c) the mean-field cells at the very left are destroyed. Then (d) the *core* cells at the very left become mean-field, whereas the mean-field cells at the right retain their actual configurations and enter the new core. Finally (e) new mean-field cells are created from scratch at the very right to complete the mean-field casing, the proper cell-to-cell connections are established (f), and the system is ready to undergo the next randomization-propagation cycle (g).

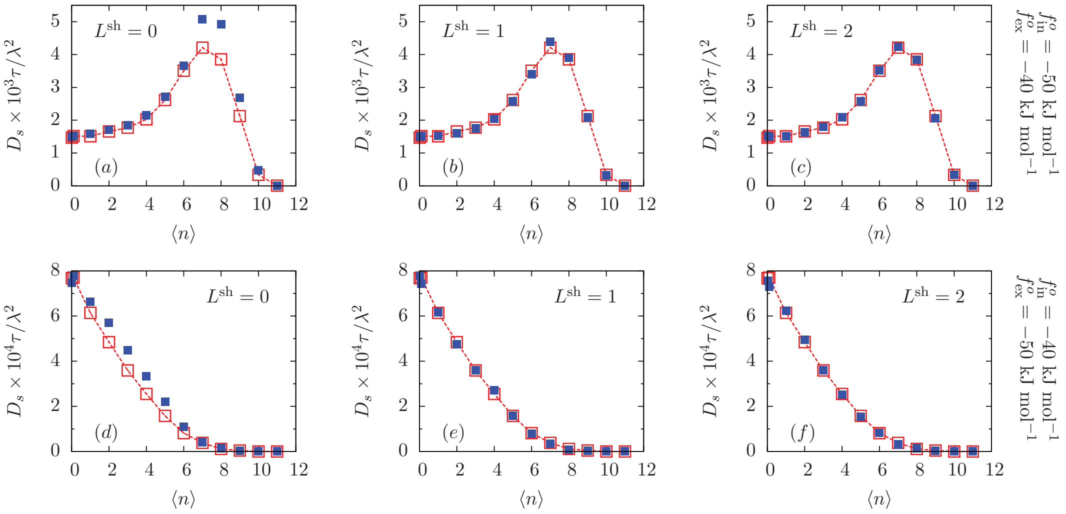


Figure 4.5: The self-diffusivity, D_s, resulting from numerical simulations (in the canonical ensemble) of the traditional lattice-gas automaton model for a *closed* test system of $9 \times 9 \times 9$ cells, in comparison with the results of (grand-canonical) simulations of the Central Cell Model with increasing number of shells L^{sh}. In the first row, the inner sites have been set as deeper than the exit sites, and vice-versa in the second row. Ex-ex jumps are allowed.

Table 4.2: The set of direction vectors (cubic lattice).

$\mathbf{e}_0 = (0,0,0)$		
$\mathbf{e}_1 = (\lambda,0,0)$	$\mathbf{e}_2 = (0,\lambda,0)$	$\mathbf{e}_3 = (0,0,\lambda)$
$\mathbf{e}_4 = (-\lambda,0,0)$	$\mathbf{e}_5 = (0,-\lambda,0)$	$\mathbf{e}_6 = (0,0,-\lambda)$

time the cell-to-cell displacements. It is related to the self-diffusivity via the Green-Kubo formula [60]:

$$D_s = \frac{1}{2d\tau}\left[\langle\delta\mathbf{r}(0)\cdot\delta\mathbf{r}(0)\rangle + 2\sum_{z=1}^{\infty}\langle\delta\mathbf{r}(z\tau)\cdot\delta\mathbf{r}(0)\rangle\right], \tag{4.19}$$

where $d = 3$ is the number of dimensions of a cubic lattice. Details about the derivation of Equation (4.19) can be found in Appendix A.2. The peculiarity of the DACF in a regular lattice is that it is strictly connected to the jump probability. It is the aim of this section to reconstruct the terms appearing in Equation (4.19) starting from the list of the possible movements of the tagged particle.

Contribution at the initial time

First of all the contribution at $t = 0$,

$$D_0^{\mathrm{mf}} = \frac{1}{2d\tau} \left\langle \delta\mathbf{r}(0) \cdot \delta\mathbf{r}(0) \right\rangle , \tag{4.20}$$

that is the uncorrelated diffusivity, proportional to the DACF at time zero, turns out to be also proportional to the escape probability of the guest from the host cell. The escape event will be indicated with the symbol $\Diamond$. In terms of the randomization-propagation dynamics, such an event can be rewritten as:

$\Diamond$ The guest reaches any of the K_{ex} exit sites of the current cell during randomization, and then the propagation step lets it migrate to the corresponding neighboring cell during propagation.

Since at the initial time $\delta\mathbf{r}(0) \cdot \delta\mathbf{r}(0)$ equals λ^2 if the guest migrates to a neighboring cell and 0 otherwise, then Equation (4.20) can be rewritten as

$$D_0^{\mathrm{mf}} = \frac{1}{2d}\frac{\lambda^2}{\tau} p(\Diamond), \tag{4.21}$$

where $p(\Diamond) = \lambda^{-2} \left\langle \delta\mathbf{r}(0) \cdot \delta\mathbf{r}(0) \right\rangle$ is the escape probability.

Contribution after one iteration: a probabilistic interpretation of the normalized DACF

Now, let us suppose that at time zero the particle escaped its host cell along a generic *non-null* direction $\mathbf{e}_j$ picked out of the set of direction vectors, listed in Table 4.2 for a cubic lattice. This is the starting point for the listing of all the subsequent events along with their respective probabilities, represented as a Markov Chain. In this approach the choice of a (hyper)cubic topology turns out to be the most convenient, since $\delta\mathbf{r}(t') \cdot \delta\mathbf{r}(t)$ is non-zero if and only if the displacements at the times t and t' are parallel and non-null. More specifically, it is positive if the displacement direction are the same, and it is negative if they are equal but opposite. Therefore the normalized DACF, $\left\langle \delta\mathbf{r}(z\tau) \cdot \delta\mathbf{r}(0) \right\rangle / \left\langle \delta\mathbf{r}(0) \cdot \delta\mathbf{r}(0) \right\rangle$, represents the conditional probability of a guest to migrate at time $z\tau$ in the same direction of displacement at time 0, given that at time 0 the displacement was not null, *minus* the conditional probability of a migration in the opposite direction.

We will proceed now with the listing of the basic in-cage and cage-to-cage jump events at the time $t = \tau$, given a successful propagation at the previous time. Every event will be associated a symbol, ς, taking values in the following set:

$$S = \{\Rightarrow, \rightarrow, \Leftarrow, \leftarrow, \Updownarrow, \updownarrow, \bigcirc\} , \tag{4.22}$$

meaning respectively, for a given direction of motion (say the x axis), ($\Rightarrow$) successful and ($\rightarrow$) unsuccessful step forward, ($\Leftarrow$) successful and ($\leftarrow$) unsuccessful step backwards, ($\Updownarrow$) successful and ($\updownarrow$) unsuccessful step out of the direction of motion, $\bigcirc$ no attempt of leaving the cell.

The main approximation in the mean-field analysis is a factorization of the joint probability, $p(\Diamond, \varsigma)$, of an escape event ($\Diamond$) followed by the event ς at the next time step:

$$p(\Diamond, \varsigma) = p(\Diamond)p(\varsigma|\Diamond) \tag{4.23}$$

For the sake of clarity, in the list that follows we will give a short description of the events mentioned in Equation (4.22). Those events are also sketched in Figure 4.6.

$\Rightarrow$ A step forward. The randomization moves the particle from the exit site into the opposite one. After this, the propagation is successful and the particle migrates in the corresponding neighboring cell. This happens with conditional probability $p(\Rightarrow|\Diamond)$.

$\Leftarrow$ A backscattering event. At the end of randomization the particle finds itself in the same exit site it entered by the event $\Diamond$. The propagation is successful and the particle jumps back into the cell it occupied before event $\Diamond$. (Conditional probability: $p(\Leftarrow|\Diamond)$).

$\Updownarrow$ A change of direction. The particle performs a migration jump whose direction is not parallel to the direction of the jump performed during the event $\Diamond$. (Conditional probability: $p(\Updownarrow|\Diamond)$).

$\bigcirc$ The guest reaches an inner site of the current cell during randomization. (Conditional probability: $p(\bigcirc|\Diamond)$).

Single arrows, i.e., $\rightarrow$, $\leftarrow$, and $\updownarrow$, differ from $\Rightarrow$, $\Leftarrow$, and $\Updownarrow$ respectively in the fact that the propagation event is unsuccessful.

Let us now introduce the quantity $\chi(\varsigma|\Diamond)$, returning a value 1 if the cell-to-cell displacement at time t (represented by the symbol ς) has equal sign of the displacement at time 0, a value -1 if the sign is opposite, and 0 in all other cases:

$$\chi(\varsigma|\Diamond) = \begin{cases} 1, & \text{if } \varsigma = \Rightarrow \\ -1, & \text{if } \varsigma = \Leftarrow \\ 0, & \text{otherwise} . \end{cases} \tag{4.24}$$

Therefore, since the process is Markovian one can define

$$\langle \delta\mathbf{r}(\tau) \cdot \delta\mathbf{r}(0) \rangle = \lambda^2 p(\Diamond) \sum_{\varsigma \in S} \chi(\varsigma|\Diamond) p(\varsigma|\Diamond), \tag{4.25}$$

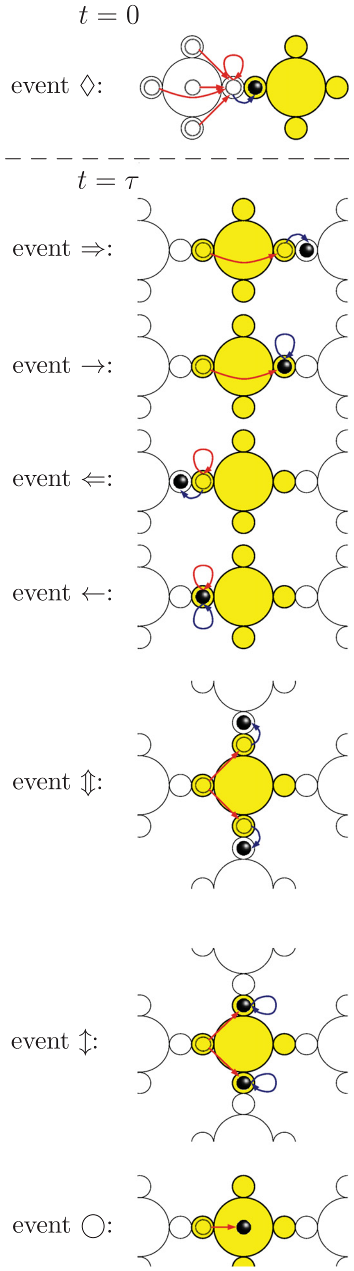


Figure 4.6: A graphical 2-d representation of the main events contributing to the diffusive motion of a single particle in the automaton. The events pictured here for times $t = \tau$ are assumed to echange their role in time with the event for $t = 0$ according to a Markov chain. For each event, the black 3-d sphere represent the actual position of a tagged particle (other guest particles eventually present are omitted), while the empty circles represent its possible(s) position(s) at the immediately preceding time step. Red and blue arrows represent respectively the possible randomization and propagation outcomes.

where the set S has been defined in Equation (4.22), which returns

$$\langle \delta\mathbf{r}(\tau)\cdot\delta\mathbf{r}(0)\rangle = \lambda^2 p(\Diamond)\left[p(\Rightarrow|\Diamond) - p(\Leftarrow|\Diamond)\right]. \tag{4.26}$$

Contribution after several iterations

Since we are assuming the migration process to be Markovian, the conditional migration probabilities for $t = 2\tau$ will depend only on the outcome at time $t' = \tau$. Relations between the conditional probabilities after two steps and those after one step are listed in Table 4.3. It should be noted that a guest starting from an inner site or from an exit site not pointing toward the direction e_j nor $-\mathrm{e}_j$ will have equal probability to reach those sites during randomization. This means that when the starting position is $\bigcirc$, or $\updownarrow$, or $\Updownarrow$, the net average displacement is null. Therefore, only the moves $\Rightarrow, \rightarrow, \Leftarrow, \leftarrow$ do contribute in the general formula for the mean-field DACF:

$$\begin{aligned}\langle \delta\mathbf{r}(z\tau)\cdot\delta\mathbf{r}(0)\rangle = &\lambda^2 p(\Diamond)\sum_{\varsigma_1\in S}\cdots\sum_{\varsigma_z\in S}\chi(\varsigma_z|\Diamond)p(\varsigma_1|\Diamond)\\ &\times\prod_{j=1}^{z-1}p(\varsigma_{j+1}|\varsigma_j), \qquad z\geq 1.\end{aligned} \tag{4.27}$$

Therefore, general mean-field expressions can be formulated for both the DACF and the self-diffusivity, Equation (4.19):

$$\begin{aligned}\langle \delta\mathbf{r}(z\tau)\cdot\delta\mathbf{r}(0)\rangle = &\lambda^2 p(\Diamond)\Big[p(\Rightarrow|\Diamond) - p(\Leftarrow|\Diamond)\Big]\\ &\left\{p(\Rightarrow|\Diamond) - p(\Leftarrow|\Diamond) - \Big[p(\rightarrow|\Diamond) - p(\leftarrow|\Diamond)\Big]\right\}^{z-1}.\end{aligned} \tag{4.28}$$

$$D_s^{\mathrm{mf}} = \frac{\lambda^2}{d\tau}p(\Diamond)\left\{\frac{1}{2} + \frac{p(\Rightarrow|\Diamond) - p(\Leftarrow|\Diamond)}{1 + p(\rightarrow|\Diamond) - p(\leftarrow|\Diamond) - \Big[p(\Rightarrow|\Diamond) - p(\Leftarrow|\Diamond)\Big]}\right\}. \tag{4.29}$$

Equation (4.29) is a quite general approximated equation. The terms in it can be obtained straight from a numerical simulation of the Central Cell Model. One can proceed as follows: for evaluating $p(\Diamond)$ it is enough to store the number of cell-to-cell jumps, $N_\Diamond$, of the tagged particle, and then dividing it by the number of time iterations (say, N_τ):

$$p_{\mathrm{num}}(\Diamond) = \frac{N_\Diamond}{N_\tau}, \tag{4.30}$$

Table 4.3: Possible guest jumps after two time steps for the case where during the jump randomization each guest can select any of the K sites in the cell as target sites.

$p(\varsigma\mid \Rightarrow)$	
$p(\Rightarrow \mid \Rightarrow) = p(\Rightarrow \mid\Diamond)$	$p(\rightarrow \mid \Rightarrow) = p(\rightarrow \mid\Diamond)$
$p(\Leftarrow \mid \Rightarrow) = p(\Leftarrow \mid\Diamond)$	$p(\leftarrow \mid \Rightarrow) = p(\leftarrow \mid\Diamond)$
$p(\Updownarrow \mid \Rightarrow) = p(\Updownarrow \mid\Diamond)$	$p(\updownarrow \mid \Rightarrow) = p(\updownarrow \mid\Diamond)$
$p(\bigcirc\mid \Rightarrow) = p(\bigcirc\mid\Diamond)$	
$p(\varsigma\mid \rightarrow)$	
$p(\Rightarrow \mid \rightarrow) = p(\Leftarrow \mid\Diamond)$	$p(\rightarrow \mid \rightarrow) = p(\leftarrow \mid\Diamond)$
$p(\Leftarrow \mid \rightarrow) = p(\Rightarrow \mid\Diamond)$	$p(\leftarrow \mid \rightarrow) = p(\rightarrow \mid\Diamond)$
$p(\Updownarrow \mid \rightarrow) = p(\Updownarrow \mid\Diamond)$	$p(\updownarrow \mid \rightarrow) = p(\updownarrow \mid\Diamond)$
$p(\bigcirc\mid \rightarrow) = p(\bigcirc\mid\Diamond)$	
$p(\varsigma\mid \Leftarrow)$	
$p(\Rightarrow \mid \Leftarrow) = p(\Leftarrow \mid\Diamond)$	$p(\rightarrow \mid \Leftarrow) = p(\leftarrow \mid\Diamond)$
$p(\Leftarrow \mid \Leftarrow) = p(\Rightarrow \mid\Diamond)$	$p(\leftarrow \mid \Leftarrow) = p(\rightarrow \mid\Diamond)$
$p(\Updownarrow \mid \Leftarrow) = p(\Updownarrow \mid\Diamond)$	$p(\updownarrow \mid \Leftarrow) = p(\updownarrow \mid\Diamond)$
$p(\bigcirc\mid \Leftarrow) = p(\bigcirc\mid\Diamond)$	
$p(\varsigma\mid \leftarrow)$	
$p(\Rightarrow \mid \leftarrow) = p(\Rightarrow \mid\Diamond)$	$p(\rightarrow \mid \leftarrow) = p(\rightarrow \mid\Diamond)$
$p(\Leftarrow \mid \leftarrow) = p(\Leftarrow \mid\Diamond)$	$p(\leftarrow \mid \leftarrow) = p(\leftarrow \mid\Diamond)$
$p(\Updownarrow \mid \leftarrow) = p(\Updownarrow \mid\Diamond)$	$p(\updownarrow \mid \leftarrow) = p(\updownarrow \mid\Diamond)$
$p(\bigcirc\mid \leftarrow) = p(\bigcirc\mid\Diamond)$	
$p(\varsigma\mid \Updownarrow)$	
$p(\Rightarrow \mid \Updownarrow) = p(\Rightarrow \mid \updownarrow)$	$p(\rightarrow \mid \Updownarrow) = p(\rightarrow \mid \updownarrow)$
$p(\Leftarrow \mid \Updownarrow) = p(\Leftarrow \mid \updownarrow)$	$p(\leftarrow \mid \Updownarrow) = p(\leftarrow \mid \updownarrow)$
$p(\Updownarrow \mid \Updownarrow) = p(\Updownarrow \mid \updownarrow)$	$p(\updownarrow \mid \Updownarrow) = p(\updownarrow \mid \updownarrow)$
$p(\bigcirc\mid \Updownarrow) = p(\bigcirc\mid \updownarrow)$	
$p(\varsigma\mid \updownarrow)$	
$p(\Rightarrow \mid \updownarrow) = p(\Leftarrow \mid \updownarrow)$	$p(\rightarrow \mid \updownarrow) = p(\leftarrow \mid \updownarrow)$
$p(\varsigma\mid\bigcirc)$	
$p(\Rightarrow \mid\bigcirc) = p(\Leftarrow \mid\bigcirc)$	$p(\rightarrow \mid\bigcirc) = p(\leftarrow \mid\bigcirc)$

where the subscript “num” denotes that the quantity has been evaluated from a numerical simulation.

For evaluating the conditional probability, it will be enough to store the jump direction every time the tagged particle performs a cell-to-cell jump. At the next time

(i) if the particle performs another jump in the same direction as before, the quantity $N_{\Rightarrow}$ is increased by one,

(ii) if the particle fails a jump attempt toward the same direction as before, the quantity $N_{\rightarrow}$ is increased by one,

(iii) if the particle performs a jump toward the *opposite* direction, then the quantity $N_{\Leftarrow}$ is increased by one,

(iv) if the particle fails a jump attempt towards the opposite direction, then the quantity $N_{\leftarrow}$ is increased by one.

Then the conditional probabilities are obtained as

$$p_{\text{num}}(\varsigma|\Diamond) = \frac{N_{\varsigma}}{N_{\Diamond}}, \qquad \varsigma \in \{\Rightarrow, \rightarrow, \Leftarrow, \leftarrow\}\,. \tag{4.31}$$

Results of the numerical mean-field evaluation of Equation (4.29) will be compared with the self-diffusivity obtained by explicit calculation of the DACF from the output of the simulations in the Results and Discussion section.

4.5 Mean-field DACF: theoretical prediction of self-diffusivity

In this Section we derive an approximate mean-field expression for the DACF. We will first apply the general mean-field DACF formula in Equation (4.28) to the limiting case of infinite dilution. Then, we will propose further approximations to apply Equations (4.28) and (4.29) to the case of diffusion at arbitrary loading.

4.5.1 Exact DACF in the limit of infinite dilution

When the motion of a lone particle in an empty system is considered, correlations with the motion of other particles are absent and an exact mathematical formula for the DACF can be written. In this limit the migration probability during propagation if the particle stays in an exit site is

$$J_{\text{prop}} = \frac{1}{2}\gamma e^{\beta[f^{o}_{\text{ex}} - \epsilon_{\text{ki}}(1,0)]} \tag{4.32}$$

Table 4.4: Probability values for events of jump starting from initial condition $\Diamond$ at time 0 for the case of jump randomization with allowed (upper part) and forbidden (lower part) ex-ex jumps, where $\gamma_{\text{ex}} = [(K_{\text{ex}} - 1)/K]J_{\text{ex}-\text{ex}}$ is the probability of the guest to jump into an exit site different from the departure one, and $\gamma_{\text{in}} = (K_{\text{in}}/K)J_{\text{ex}-\text{in}}$ is the probability to jump to an inner site.

Allowed ex-ex jumps
$p(\Rightarrow \vert\Diamond) = (1/K)J_{\text{ex}-\text{ex}}J_{\text{prop}}$
$p(\rightarrow \vert\Diamond) = (1/K)J_{\text{ex}-\text{ex}}\,(1 - J_{\text{prop}})$
$p(\Leftarrow \vert\Diamond) = (1 - \gamma_{\text{ex}} - \gamma_{\text{in}})\,J_{\text{prop}}$
$p(\leftarrow \vert\Diamond) = (1 - \gamma_{\text{ex}} - \gamma_{\text{in}})\,(1 - J_{\text{prop}})$
Forbidden ex-ex jumps
$p(\Leftarrow \vert\Diamond) = [1 - \gamma_{\text{in}}]\,J_{\text{prop}}$
$p(\leftarrow \vert\Diamond) = [1 - \gamma_{\text{in}}]\,(1 - J_{\text{prop}})$

and $p(\Diamond)$ is given by

$$p(\Diamond) = p_{\text{ex}}J_{\text{prop}}, \tag{4.33}$$

where

$$p_{\text{ex}} = \frac{K_{\text{ex}}e^{-\beta f^o_{\text{ex}}}}{K_{\text{ex}}e^{-\beta f^o_{\text{ex}}} + K_{\text{in}}e^{-\beta f^o_{\text{in}}}} \tag{4.34}$$

is the equilibrium probability of the lone particle to occupy an exit site. The other terms in Equation (4.28) can be determined by properly weighting every possible randomization jump. They are listed in Table 4.4 for both the case of allowed and forbidden ex-ex jumps [i.e., use of $\{C'_{ab}\}$ or $\{C''_{ab}\}$ matrix, Equations (4.9) and (4.10), during the randomization procedure]. In the infinite dilution limit the quantities $J_{\text{ex}-\text{ex}}$ and $J_{\text{ex}-\text{in}}$ mentioned in the formulas of Table 4.4 have the same value:

$$J_{\text{ex}-\text{ex}} = J_{\text{ex}-\text{in}} = J_{\text{ex}} := \gamma e^{\beta f^o_{\text{ex}}}. \tag{4.35}$$

Since its value depends only on the departure (exit) site, we simply called it J_{ex}.

$$\begin{aligned}\lim_{\langle n\rangle\to 0}\langle \delta\mathbf{r}(z\tau)\cdot\delta\mathbf{r}(0)\rangle = &- \lambda^2 p_{\text{ex}}J^2_{\text{prop}}\,(1 - 2J_{\text{prop}})^{z-1}\\ &\times (1 - J_{\text{ex}})^z\,,\end{aligned} \tag{4.36}$$

for the case of allowed ex-ex jumps, and

$$\begin{aligned}\lim_{\langle n\rangle\to 0}\langle \delta\mathbf{r}(z\tau)\cdot\delta\mathbf{r}(0)\rangle = &- \lambda^2 p_{\text{ex}}J^2_{\text{prop}}\,(1 - 2J_{\text{prop}})^{z-1}\\ &\times \left(1 - \frac{K_{\text{in}}}{K}J_{\text{ex}}\right)^z\,,\end{aligned} \tag{4.37}$$

for the case of forbidden ex-ex jumps We remark that Equation (4.36) is independent of the number of exit/inner sites in the cell, while Equation (4.37), where jumps between different exit sites are forbidden, shows an explicit dependence on the number of sites constituting the cell. Therefore the accessibility of the adsorption sites plays a fundamental role in determining the entity of correlations.

4.5.2 Approximated mean-field DACF and self-diffusivity at arbitrary loading

At arbitrary loadings the tagged particle is likely to share its host and neighboring cells with other particles. This means that, during randomization, the variety of sequences in which the particles can be invoked to attempt a jump have an effect on the probability of the tagged particle to reach an exit site, as well as they affect the tendency of the cell to keep memory of its previous configurations from time to time. Since we are interested in improving our understanding of the self-diffusion process by obtaining a *readable* equation,

(i) we will treat as a mean-field the other guests sharing the cell with the tagged particle. That is, we assume that when the tagged guest is invoked to attempt a jump during the randomization process, the other guests in the cell are distributed according to the equilibrium distribution. This is equivalent to approximating the jump randomization scheme with a different local operation where, just before the tagged guest is invoked, all the other guests in the cell undergo a memoryless randomization (see Section 4.1.1). Such an approximation will become less accurate the more binding are the sites and the more restricted the dynamics is, since given these conditions the cell reaches local equilibrium more slowly,

(ii) we will treat mean-field randomization and propagation separately. In other words, the probability of jumping toward some direction will be factorized into probability of reaching some exit site during randomization and probability of performing a successful propagation, treated as independent one of the other.

The DACF at $t = 0$ is not affected by time correlations and can be well approximated with

$$\begin{aligned}\langle \delta\mathbf{r}(0) \cdot \delta\mathbf{r}(0) \rangle =& \lambda^2 \frac{1}{\langle n \rangle} \sum_{\mathbf{n}} \sum_{\mathbf{m}} n_{\text{ex}} \left(1 - \frac{m_{\text{ex}}}{K_{\text{ex}}} \right) \\ & \times p(\mathbf{n})\, p(\mathbf{m}) \kappa(\mathbf{n}, \mathbf{m}),\end{aligned} \tag{4.38}$$

where λ is the lattice spacing, and $\langle n \rangle$ is the loading (average number of occupied sites in a cell). The relations among D_0^{mf}, $\langle \delta\mathbf{r}(0) \cdot \delta\mathbf{r}(0) \rangle$ and $p(\Diamond)$ are given in Equations (4.20) and (4.21).

As we can see in Equations (4.28) and (4.29), the probabilities of interest refer to jumps starting from an exit site position. Thus, when evaluating the DACF terms for $z \geq 1$, one has to consider the conditional probability of the tagged guest *already located in an exit site* to stay in a cell with meso-configuration $\mathbf{n}$, rather than the absolute probability of $\mathbf{n}$ itself. Therefore we introduce $g_{\mathrm{ex}}(\mathbf{n})$, that can be re-interpreted as the conditional probability of a cell with an occupied exit site to be meso-configured such as $\mathbf{n}$, i.e., to have $n_{\mathrm{ex}} - 1$ of the remaining $K_{\mathrm{ex}} - 1$ exit site and n_{in} of the K_{in} inner sites filled,

$$g_{\mathrm{ex}}(\mathbf{n}) = \frac{n_{\mathrm{ex}} p(\mathbf{n})}{\sum_{\mathbf{n}'} n'_{\mathrm{ex}} p(\mathbf{n}')}, \tag{4.39}$$

where the quantity

$$\frac{n_{\mathrm{ex}}}{K_{\mathrm{ex}}} p(\mathbf{n}) = [\Xi(\mu)]^{-1} \binom{K_{\mathrm{ex}} - 1}{n_{\mathrm{ex}} - 1} \binom{K_{\mathrm{in}}}{n_{\mathrm{in}}} e^{\beta\mu n} e^{-\beta F(\mathbf{n})} \tag{4.40}$$

is the total probability of one particular exit site, $n_{\mathrm{ex}} - 1$ of the remaining exit sites, and n_{in} inner sites to be occupied in a cell.

Mean-field jump randomization

Once defined the probability distribution g_{ex} in Equation (4.39), it is straightforward to derive mean-field expressions for the probability that, once the tagged particle has targeted another exit site, it reaches

$$J_{\mathrm{ex-ex}} = \gamma e^{\beta f^o_{\mathrm{ex}}} \sum_{\mathbf{n}} \left(1 - \frac{n_{\mathrm{ex}} - 1}{K_{\mathrm{ex}} - 1} \right) g_{\mathrm{ex}}(\mathbf{n}), \tag{4.41}$$

This is the average acceptance of an exit-to-exit jump during randomization. Similarly, the average acceptance of an exit-to-inner jump is

$$\begin{aligned} J_{\mathrm{ex-in}} = & \gamma e^{\beta f^o_{\mathrm{ex}}} \sum_{\mathbf{n}} \left(1 - \frac{n_{\mathrm{in}}}{K_{\mathrm{in}}} \right) g_{\mathrm{ex}}(\mathbf{n}) e^{\beta \Phi(\mathbf{n})} \\ & \times e^{-\beta \max[\Phi(n_\alpha - 1, n_\nu + 1), \Phi(\mathbf{n})]} \end{aligned} \tag{4.42}$$

where γ has been defined when illustrating Equation (4.8).

Mean-field propagation

The mean-field propagation probability, that is the probability that during propagation a guest located in an exit site effectively migrates into the corresponding neighboring cell (this is sometimes referred to as *transmission coefficient*), can be formulated as

$$J_{\text{prop}} = \sum_{\mathbf{n}} \sum_{\mathbf{m}} \left(1 - \frac{m_{\text{ex}}}{K_{\text{ex}}}\right) g_{\text{ex}}(\mathbf{n}) p(\mathbf{m}) \kappa(\mathbf{n}, \mathbf{m}). \tag{4.43}$$

Mean-field jump probabilities

We are now ready to write down mean-field expressions for the conditional probabilities included in Equations (4.28) and (4.29), for both the case of allowed and forbidden ex-ex jumps. These are listed in Table 4.4. Including them into Equations (4.28) and (4.29) gives:

$$\begin{aligned} \langle \delta\mathbf{r}(z\tau) \cdot \delta\mathbf{r}(0) \rangle = {} & -2d\tau D_0^{\text{mf}} J_{\text{prop}} \left(1 - 2J_{\text{prop}}\right)^{z-1} \\ & \left[1 - \frac{K_{\text{ex}}}{K} J_{\text{ex-ex}} - \frac{K_{\text{in}}}{K} J_{\text{ex-in}}\right]^z \qquad \text{allowed ex-ex jumps,} \end{aligned} \tag{4.44}$$

$$D_s^{\text{mf}} = D_0^{\text{mf}} \left\{ 1 - 2J_{\text{prop}} \frac{1 - \frac{K_{\text{ex}}}{K} J_{\text{ex-ex}} - \frac{K_{\text{in}}}{K} J_{\text{ex-in}}}{1 - (1 - 2J_{\text{prop}}) \left[1 - \frac{K_{\text{ex}}}{K} J_{\text{ex-ex}} - \frac{K_{\text{in}}}{K} J_{\text{ex-in}}\right]} \right\},$$

$$\text{allowed ex-ex jumps} \tag{4.45}$$

$$\langle \delta\mathbf{r}(z\tau) \cdot \delta\mathbf{r}(0) \rangle = -2d\tau D_0^{\text{mf}} J_{\text{prop}} \left(1 - 2J_{\text{prop}}\right)^{z-1} \left[1 - \frac{K_{\text{in}}}{K} J_{\text{ex-in}}\right]^z,$$

$$\text{forbidden ex-ex jumps,} \tag{4.46}$$

$$D_s^{\text{mf}} = D_0^{\text{mf}} \left\{ 1 - 2J_{\text{prop}} \frac{1 - \frac{K_{\text{in}}}{K} J_{\text{ex-in}}}{1 - (1 - 2J_{\text{prop}}) \left[1 - \frac{K_{\text{in}}}{K} J_{\text{ex-in}}\right]} \right\},$$

$$\text{forbidden ex-ex jumps,} \tag{4.47}$$

Where the series

$$\sum_{z=1}^{\infty} A^z B^{z-1} = \frac{A}{1 - AB} \tag{4.48}$$

has been used to perform the summation of the correlated part.

Unlike the more general mean-field formulas in Equations (4.28) and (4.29), the various quantities in Equations (4.44) to (4.47) do not depend on whether the ex-ex jumps are allowed or forbidden in the randomization algorithm. As can be seen, forbidding the ex-ex jumps has the only effect of dropping the term $-\frac{K_{\rm ex}}{K}J_{\rm ex-ex}$ out of the mean-field formulas.

Although the formulas above lead to a qualitatively correct representation of correlations, they do not always match quantitatively with the results of numerical simulations. Nevertheless, the obtained discrepancies are of great help in understanding the correlation mechanism, as we will discuss in Section 4.6.

4.6 Discussion of the mean-field results

In Figure 4.7 we plot the results of numerical simulations of the Central Cell Model applied in the cases where the deepest sites are the inner or exit ones respectively, each studied with two different levels of time correlation entity. The

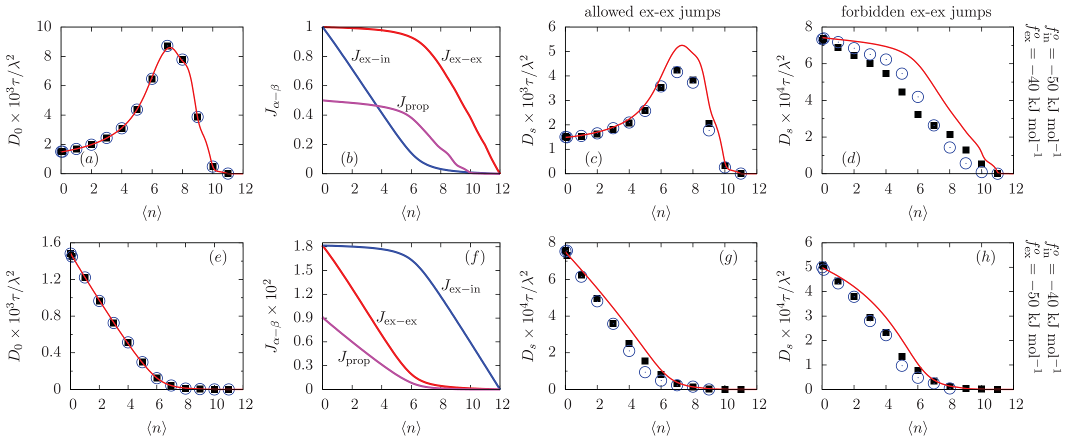


Figure 4.7: Comparison between diffusivity obtained from numerical simulations of the Central Cell Model and from the mean-field theory. Black squares are obtained from the trajectory data outcoming from numerical simulations through the Green-Kubo formula, Equation (4.19). Blue circles are obtained by applying on the same trajectory data the more general mean-field approximation, Equation (4.29). Solid lines are theoretical prediction values obtained from the more specific mean-field approximations in Equations (4.45) and (4.47). In the first row, Figures (a), (b), (c), and (d) the inner sites are 10 kJ mol^{-1} deeper than the exit sites, and vice-versa for the second row, Figures (e), (f), (g), and (h). In the first column, Figures (a) and (e), the zero-time diffusivity, Equation (4.20) is shown. In the second column, Figures (b) and (f), the trends of the macroscopic quantities $J_{\rm ex-ex}$, $J_{\rm ex-in}$, and $J_{\rm prop}$ constituting the theoretical mean-field approximation are shown. In the third and the fourth columns, respectively, Figures (c), (g) and (d), (h) the case of allowed and forbidden ex-ex jumps are considered.

values calculated explicitly from the numerical simulations through the Green-Kubo formula, Equation (4.19), are reported as black squares, whereas general mean-field values and mean-field theoretical predictions are reported as blue circles and solid lines, respectively.

As expected, the self-diffusion coefficient when the inner sites are the deepest ones increases from low to intermediate loadings as a consequence of the increasing probability of the tagged particle to occupy an exit site (thus being able to attempt a cage-to-cage jump), and starts decreasing at higher loadings when the exit sites tend to be saturated so that each pair of adjacent exit sites of communicating cells is more likely to be saturated, this leading the cells to exchange no particles. When the exit sites are the deepest ones instead, the pairs of exit sites tend to be saturated from the beginning (i.e., at low loadings), this leading to the expected decreasing diffusivity.

The escape probability, and thus D_0^{mf}, does not vary depending on whether or not the ex-ex jumps are allowed (see Figures 4.7a and d). This is because there are no correlations to be taken into account. As a consequence, both the general mean-field equation (4.29) and the more specific one obtained through the DACF value in Equation (4.38) perfectly match with the explicit numerical value of D_0^{mf}.

The functions constituting the specific mean-field equations, Figures 4.7b and f, give some insights about the migration probability of the individual processes. The way the average jump acceptances $J_{\mathrm{ex-ex}}$, $J_{\mathrm{ex-in}}$, and J_{prop} behave with respect to loading is the basis of the mean-field treatise of correlations. They are strictly connected to the choice of the difference between the site adsorption free-energies f_{ex}^{o} and f_{in}^{o}. As we described above, when the inner sites are the deepest ones the exit sites are poorly occupied. The acceptance of ex-in jumps, $J_{\mathrm{ex-in}}$, starts to decrease from low loadings whereas the acceptance of (allowed) ex-ex jump, $J_{\mathrm{ex-ex}}$, is almost unity and does not decrease significantly as long as the inner sites are not close to saturation, around $\langle n \rangle \approx K_{\mathrm{in}}$. The behavior of J_{prop} is similar to $J_{\mathrm{ex-ex}}$: it remains almost constant (about 1/2) until the loading becomes high enough so that the exit sites start being filled. Inverting the site depths f_{ex}^{o} and f_{in}^{o} exactly inverts the respective behaviors.

The average jump acceptances are combined together by Equations (4.45) and (4.47) to give approximated values for the correlated self diffusivity D_s^{mf} (see solid lines in Figures 4.7c, d, g, and h). Although the theoretical predictions are qualitatively correct, they are close to the simulation values especially at low loadings, while usually they fail at higher loadings. The more general (numerical) diffusivity equation (4.29) gives a better approximation than the theoretical prediction. This is because the numerical evaluation of D_s^{mf} through Equation (4.29) does not suffer from the separation of mean-field randomization and propagation operations, which was the leading assumption when we derived the theoretical diffusivity formulas in Section 4.5.2. However, the general diffusivity equation becomes less accurate in situations where the memory of the previous local configurations is lost slowly, as for the case shown in Figure 4.7d. When discussing about the amount of memory locally lost during each randomization step, it is interesting to find out the main sources of correlations and to identify which of the cases above is the most memory-preserving.

Memory preserved in exit and inner sites. Since the cell-to-cell migrations occur via the exit sites, and their connectivity from one cell to the other determines the topology of the whole grid of cells, all events involving them will introduce more correlation than the events occurring in the inner sites, which instead are structureless so that they can be considered as the less memory-preserving part of the cell.

Memory-preserving backscattering. When a tagged guest migrates from cell to cell during propagation, the probabilities related to every next move *do* depend on the configuration of both cells before the propagation occurred. In other words, the assumption in Equation (4.23) is strong and this is especially true when correlation effects are particularly evident, such as in the case of forbidden ex-ex jumps shown in Figure 4.7d. In that case, (i) forbidding the ex-ex jumps gives the backscattering contribution a major role in the production of correlations (this is because the randomization will produce only very small changes in the local configuration), and (ii) cage-to-cage jumps are infrequent because $f^o_{\mathrm{in}} < f^o_{\mathrm{ex}}$, so that the configuration of the exit sites tends not to change significantly from step to step. Due to these two facts, a backscattering particle which has left the cell $\mathbf{r}$ at time t and backscatters into it at time $t + \tau$ is very likely to find $\mathbf{r}$ just little changed or not changed at all. If the exit sites are the deepest instead, even though ex-ex jumps are forbidden one has that propagation events are more likely to occur at low-intermediate loadings than what expected when the inner sites were the deepest. This causes the memory-preserving attitude of the exit sites to be less marked when the migration events are frequent. Therefore, as it can be seen from Figure 4.7, the d case (deepest inner sites and forbidden ex-ex jumps) is the more affected by time correlations in the self-diffusion process. The approximation in Equation (4.23) becomes then less accurate, whereas in all the other cases it is acceptable.

4.7 Conclusions

In this work, we laid down the basis of a simple computational framework, the Central Cell Model (CCM), aimed to be specific for the study of the motion on the mesoscopic scale of a single particle in a system of connected cavities in the presence of other diffusants, in conditions of thermodynamic equilibrium. Our model is local and discrete in both space and time, and in the numerical applications we have shown here it has been constructed starting from the algorithm of a lattice-gas model for diffusion in microporous material. We have shown that, although being not possible for the CCM to sample all the informations obtainable by a full lattice-gas, a CCM simulation provides an accurate reproduction of the memory effects in the self-diffusion (and thus, of the diffusion isotherm) at a minimum computational cost.

The way the CCM is constructed suggested how to carry on a mean-field study of the self-diffusion process produced by the particular evolution rule adopted. This has led to two approximated mathematical expressions for self-diffusion. The first one, more general, can be applied with data coming straight from the CCM simulation. The second one, more case-specific and derived by assuming fast local equilibration, is theoretical and yelds a more accurate approximation the weaker the correlations and the lower the loadings are. Interpretation of the discrepancies between the self-diffusivity trends obtained from the numerical simulations and their two different mean-field approximations helped to understand how, and how strongly, memory effects can emerge depending on the very general features of the model parametrization.

The obtained results suggest the CCM approach to be suitable for other theoretical studies, e.g., the time correlations in the local density [101, 102], as well as for direct applications in the field of the molecular coarse-graining. For example, the CCM approach could be further extended to the sampling of both the adsorption and the self-diffusion isotherm through a single simulation when the lattice-gas rule includes an explicit cell-to-cell interaction potential which makes (in principle) impossible do derive the equilibrium probability distribution of states *a priori*. This could be done by performing a grand-canonical Monte Carlo on the border cells while keeping the core evolving with the prescribed dynamic lattice-gas rule in the canonical ensemble. Also, an even more intriguing extention of the CCM approach could be made in the field of hybrid MC-MD schemes aimed to realistically mimic the bulk effects in the motion of a tagged guest in an atomistic simulation.

Chapter 5

Development and optimization of a new Force Field for flexible aluminosilicates, enabling fast Molecular Dynamics simulations on parallel architectures

Adapted with permission from Andrea Gabrieli, Marco Sant, Pierfranco Demontis, and Giuseppe B. Suffritti; The Journal of Physical Chemistry C; 2013, 117 (1), pp 503-509. "Copyright 2012 American Chemical Society."

http://dx.doi.org/10.1021/jp311411b

Despite the increase of computational performance, thanks to the spreading of parallel architecture, it is still not feasible to follow the dynamical evolution of a system including more than several hundred atoms via *ab-initio* methods for more than a few picoseconds.

For this reason, classical molecular dynamics (MD) computations are not going to disappear in the near future and are still widely adopted by the scientific community. In recent years, a number of MD packages that can exploit massively parallel architecture have been developed [103–108]; among these we choose NAMD [103] for its open source policy joined to its high performance in our computing facility. Among the great variety of functional forms for classical force fields available, a widely used one is the CHARMM [109] type, on which we will rely as well. The CHARMM force field has been originally developed to simulate biological macromolecules possessing a carbon-based backbone. Silicates share with tetravalent carbon a local tetrahedral symmetry, so it is reasonable that, after a suitable parameter optimization, the CHARMM functional form will be able to reproduce the structural and vibrational properties of silicates (e.g., quartz structure parameters are already available from Lopes *et al.* [110]).

The aim of this work is to adapt and optimize the force field previously developed in our laboratory for MD simulations of aluminosilicates via serial codes [111–117] in a new different functional form to enable fast MD simulations in a parallel environment. In particular, we will focus our attention on the following zeolitic structures: silicalite, zeolite Na A, Ca A, Na Y, and Na X.

5.1 Theory and models

5.1.1 Framework structures

Zeolites are microporous aluminosilicates [118, 119]. Their structure consists of a regular network of channels and/or cages of molecular dimensions (up to 1.2 nm), interconnected by windows (up to 0.8 nm in diameter). The aluminosilicate framework is built up by corner sharing TO_4 tetrahedra (in which the T-sites are occupied by either silicon or aluminium), giving rise to a rather complex but precisely periodic atomic network. Cavities and channels are studded with cations (usually metallic), which compensate for the charge deficit due to the substitution of silicon by aluminium, when present. Molecules can be adsorbed inside these materials and manifest several unexpected behaviors generated by the framework confinement [62–64].

The structures involved in this work vary considerably in both crystal geometry and free volume connectivity (for structure visualization, we use the VMD [120] software). In particular, silicalite is the purely siliceous form of MFI-type zeolite, one of the most studied crystals in the literature [119, 121], thanks to its widespread industrial employment. At low temperatures, its evacuated structure presents a monoclinic symmetry. At high temperatures, it undergoes a phase transition, and the symmetry becomes orthorhombic. Its free volume has a peculiar connectivity made of straight channels along the y direction and sinusoidal channels along the z direction, with ten-membered ring pore openings of $\sim$0.56 nm in diameter [122, 123].

At the same time, Na A and Ca A are two well-characterized cationic forms of LTA-type zeolite. This crystal is constituted by cubooctahedral sodalite cages arranged cubically around larger ($\sim$1.12 nm) $\alpha-$cages that are interconnected by eight-membered ring pores with kinetic diameter of $\sim$0.43 nm. The small pore size makes this zeolite suitable for separation processes thanks to its permeability limited to small molecules like N_2, CO_2, CH_4 [62]. The framework of

Table 5.1: Number of atoms per unit cell and lattice parameters for the simulated structures. In parentheses, number of cations.

zeolite	atoms	a (nm)	b (nm)	c (nm)
Sil	288	2.0022	1.9899	1.3383
Na A	672(96)	2.4555	2.4555	2.4555
Ca A	624(48)	2.4555	2.4555	2.4555
Na Y	632(56)	2.4850	2.4850	2.4850
Na X	664(88)	2.5051	2.5051	2.4051

this zeolite contains Si and Al in a 1:1 ratio (i.e., there is a regular alternation of the two atom kinds). The unbalance in total charge caused by the presence of the 96 Al^{3+} cations in the crystallographic unit cell is compensated by the presence of exchangeable cations: in the case of Na A, 96 Na^{+} are required, while in the case of Ca A the number is halved to 48 Ca^{2+}. Cations within zeolite A structure occupy well-defined sites, and those in the literature have been divided into three groups according to their position with respect to the pores, in a given unit cell: 64 type I sites, eight for each sodalite cage in the six-membered rings; 24 type II sites, one for each eight-membered ring window connecting two adjacent $\alpha-$cages; 48 type III sites, one for each four-membered ring pore [62]. In the case of Na A, most of the favorable sites are occupied, and a digital reconstruction of this structure is relatively straightforward [124] (i.e., type I and type II sites are fully occupied while the remaining 8 cations are distributed among the type III sites, one in between each couple of adjacent sodalite cages). On the other hand, crystallographic data [125] show that Ca cations prefer to stay in the type I sites [126]. Then, one has to choose which of these 64 type I sites are occupied by the 48 Ca. We used an *ad-hoc* procedure to randomly distribute six cations per sodalite cage. Considering all the possible configurations, we identified the ones having the most occurring potential energy and randomly chose one of these (a discussion of the full procedure used is reported in Appendix B).

Finally, to investigate the portability of our new force field, we study the FAU-type structure, in its two variants: Na Y and Na X. This structure is very important from an industrial point of view, in particular for petrochemical applications [62, 63, 78]. The cages are arranged in a tetrahedral array, with wide pores (∼0.75 nm) made of 12-membered rings. The difference between Na Y and Na X lies in the Si to Al ratio: this ranges from 1.0 to 1.5 for X [62] and from 1.5 to 3 for Y; in this work we use 1.18 for X [127] and 2.43 for Y [80].

In Table 5.1 the unit cell sizes for each investigated structure are reported.

5.1.2 Theoretical background

Due to its widespread use within the main MD packages, we choose to develop a CHARMM type force field, having the following functional form:

$$\begin{aligned} E_{\mathrm{pot}} = & \sum_{\mathrm{bonds}} k_{\mathrm{b}}(b-b_0)^2 + \sum_{\mathrm{angles}} k_{\theta}(\theta-\theta_0)^2 \\ & + \sum_{\mathrm{UB}} k_{\mathrm{u}}(u-u_0)^2 \\ & + \sum_{\mathrm{vdW}} \epsilon \left[\left(\frac{R_{\mathrm{min}_{ij}}}{r_{ij}} \right)^{12} - 2 \left(\frac{R_{\mathrm{min}_{ij}}}{r_{ij}} \right)^{6} \right] \\ & + \sum_{i<j} \frac{q_i q_j}{\epsilon r_{ij}}. \end{aligned} \tag{5.1}$$

Here, each term is related to a specific interaction contributing to the total potential energy:

- the first one is an harmonic term representing the stretching, k_{b} being the force constant and b_0 the equilibrium distance;

- the second is the harmonic potential for angles, with k_θ being force constant and θ_0 the equilibrium angle between three bonded atoms;

- the third is the Urey-Bradley (UB) potential term which acts as a fictitious bond between two atoms, 1 and 3, connected to a common atom 2, where k_{u} is the force constant and u_0 the equilibrium distance;

- the last two are nonbonded terms, with a Lennard-Jones (12-6) for the van der Waals interaction (where R_{min} is the minimum location and ϵ is the well depth) and a Coulomb term for the electrostatic interactions.

For completeness, here we report the force field that was formerly developed in this laboratory [111] (implemented only in a serial code, thus not readily usable in fast parallel computations via modern packages). Once again, the total potential energy comes from the summation of a bonded part and a nonbonded part (both parts are divided in two terms):

$$E_{\mathrm{pot}} = E_{\mathrm{b1}} + E_{\mathrm{b2}} + E_{\mathrm{n1}} + E_{\mathrm{n2}} \tag{5.2}$$

The first bonded term accounts for the first-neighbor $\mathrm{T}-\mathrm{O}$ interaction:

$$E_{\mathrm{b1}} = \sum D\left\{1-\exp[-B(b-b_0)]\right\}^2 \tag{5.3}$$

where D and B represent the potential well depth and width, respectively, and b_0 is the equilibrium distance.

The second bonded term is related to the interaction between two atoms, 1 and 3, connected to a common atom 2 (i.e., oxygen atoms of the same tetrahedron $\mathrm{O}-(\mathrm{T})-\mathrm{O}$, and T atoms of adjacent tetrahedra $\mathrm{T}-(\mathrm{O})-\mathrm{T}$):

$$E_{\mathrm{b2}} = \sum \begin{cases} \frac{k}{2}(u-u_0)^2 + \frac{A}{6}(u-u_0)^3, & u < (u_0 - \frac{2k}{A}) \\ 0, & u \geq (u_0 - \frac{2k}{A}) \end{cases} \tag{5.4}$$

where k and A are the harmonic and anharmonic constants, respectively, and u_0 is the equilibrium distance.

Regarding the nonbonded part, the first term represents the vdW interaction between the cations and the framework oxygens (i.e., $\mathrm{Na}-\mathrm{O}$ or $\mathrm{Ca}-\mathrm{O}$):

$$E_{\mathrm{n1}} = \sum A \exp(-Br_{ij}) - \frac{C}{r_{ij}^6} \tag{5.5}$$

where A, B, and C are the Buckingham potential constants, and r_{ij} are the interatomic distances.

The second term, instead, is related to the electrostatic interaction between all atoms of the system:

$$E_{\mathrm{n2}} = \sum_{i<j} \frac{q_i q_j}{\epsilon r_{ij}} \tag{5.6}$$

where q_i and q_j are the atomic charges.

The main features of this former force field are: the anharmonic form of all the terms and the absence of direct angular dependence. This simplifies the calculations but does not allow a correct reproduction of the frequency difference between symmetric and asymmetric bond stretching modes [111]. In addition, no “exclusion policy” is considered, and the strong electrostatic interactions between nearest-neighbor atoms are compensated by suitable large constants of interatomic interactions.

5.1.3 Force field development

As a starting point in our development we rely on former force fields for aluminosilicates developed in both this laboratory [111] and other research groups [128]. On the other hand, the force field available in the CHARMM database [110] is optimized for quartz crystals; some preliminary MD simulations with these parameters show that this force field is not suitable to reproduce properly the experimental structure of the zeolites under investigation in this work, most probably due to the Al/Si substitution, entailing different $\mathrm{T}-\mathrm{O}$ interactions and forcing the presence of the cations.

We aim at developing a unique force field, common to all five zeolitic structures under investigation. Moreover, we want this force field to be a basis for further studies on other zeolites, requiring only minimal tuning to obtain reliable results. The greatest problem in this task, then, comes from the presence of the *free cations* within the framework: a good flexible force field should reproduce correctly the position and dynamics of these cations; this task is more delicate than just assuring a good reproduction of the overall zeolitic structure.

We start working with the Na A structure, where the crystallographic position of the cations has a high degree of symmetry. In our first parametrization, we choose to rely only on bonds and angles to reproduce the experimental structure. To avoid crystal collapses and distortions, we initially set the force constants at an arbitrarily high value. With this parametrization, the starting crystallographic structure of the Na A framework is maintained fairly well, but the Na cations readily lose their position entering the sodalite cages, which is unrealistic at the temperatures of interest.

Our effort, then, is to modify the parameters of the simulation to ensure the correct dynamics for the cations. This may be achieved working on various aspects: bonded interactions, exclusion policy, and nonbonded parameters.

Bonded interactions. We reduce the values of the force constants related to the bonded terms (from the initially set very high value), trying to match the potential energy interactions of our previous force field [111](i.e., splitting the various contributions of the former functional into two main parts: one representing the bonds and the other representing the angles). At this point we plot the potential energy function of the two parts and compare it with the corresponding one coming from our former force field. A subsequent comparison and tuning of the new force constant with the values taken from Nicholas *et al.* [128] gives us a good starting point for preliminary test runs, while an accurate optimization will be performed in the next section on the basis of experimental spectroscopic data.

Exclusion policy. Modern force fields (e.g., AMBER) apply a specific *exclusion policy* for the computation of nonbonded interactions among bonded atoms. The standard policy *scaled* 1-4 implies that both the vdW and electrostatic interactions between couples of bonded atoms (1-2) or between atoms bonded to a common atom (1-3) are excluded, while for the interaction between atoms separated by two other atoms (1-4) the vdW ϵ parameter for the given couple is divided by 2.0 and the electrostatic interaction is by 1.2.

Starting the simulation with the atoms located at their crystallographic position and setting the equilibrium values for bonds and angles according to the experimental data, we expect that a good choice of the force field parameters will keep the potential energy to a minimum (from an absolute value point of view), with respect to other parametrizations that will tend to deform the crystal, putting more strain on the bonds and angles springs, which counteract to stabilize the structure.

On this basis, we use the potential energy as a discriminant in developing our force field to choose the most suitable parametrization. Then, we perform various test runs to study which is the exclusion policy that gives the lowest potential energy: using no exclusion gives an unacceptable value of electrostatic interaction energy, distorting completely the framework structure; the situation improves slightly using a (1-2), (1-3) or (1-4) exclusion policy. Nonetheless, the best results are achieved with the *scaled* 1-4 exclusion. For the latter, another cycle of studies is performed to understand which are the best scaling factors for the vdW and electrostatic interactions. For the structures here investigated, we find that a good choice is to divide the electrostatic interactions of the (1-4) couple by a factor of 2.0 and leave the vdW ϵ parameter unchanged (i.e., 1.0).

Nonbonded parameters. From the tests performed in the previous paragraph, we understand that the vdW interactions have a positive role in stabilizing the crystal structure (i.e., they help to reduce the bond and angle potential energy). From this hint, we decide to take into account in our computations the vdW related to the Si and Al atoms (in the literature, in fact, for computational convenience the vdW interactions with these atoms are usually ignored because they are considered to be shielded by the strongest oxygen vdW term contribution).

At this point, we make a final adjustment modifying the values of the vdW interaction for the Na cation. It should be noted, in fact, that the charges used in our previous potential, have high values in comparison to those adopted in the most popular force fields.

For the framework atoms, they are about one-half of the formal ionic charges (2.0 e for Si, 1.5 e for Al, -1.0 e for O), whereas for the cations the full formal charge (1.0 e for Na, 2.0 e for Ca) was adopted. As the total structure electrical neutrality must be ensured, the actual value of the charges depends on the Si/Al ratio and is derived by imposing that the charge of O atoms is as close as possible to -1.0 e. This choice was made based on several reasons: it reproduces within 20% (or less) the values estimated by quantum mechanical calculations of zeolitic structures; it is in agreement with the raw chemical statement that $\mathrm{Si(Al)-O}$ bonds are *half ionic and half covalent* in character and; last but not least, it is in line with the charges derived from available X-ray diffraction experiments [129].

After testing various possibilities, optimal results for Na are found using the vdW ϵ parameter taken from Pantatosaki and Papadopoulos [130] and the $R_{\min}$ taken from the "par_all27_lipid.prm" CHARMM database [131, 132] (for Ca, both ϵ and $R_{\min}$ parameters are taken from this database [133]). With these settings, the correct position of the Na cations has been achieved, and there has been no entrance of Na inside the sodalite cages for all the duration of the test simulation (tens of nanoseconds) for temperatures up to 1000 K.

5.2 Results and discussion

5.2.1 Optimization

At this point, we start a more accurate refinement process of our force field. The procedure used to optimize the parameters relies on the computation of the Infra Red (IR) spectrum and its comparison with experimental data [111, 134–136].

First, we optimize the force field for silicalite (since it needs less parameters thanks to the absence of cations), adjusting the force constants until a good match between experimental and simulated spectra is attained, and then we move to the zeolite A, Y, and X structures to complete the refinement of those parameters that are not present in the silicalite force field, namely, the Al and cation-related terms. To obtain the simulated spectra, we perform MD runs in the microcanonical (*NVE*) ensemble, with a time step of 1.0 fs, applying periodic boundary conditions (PBC) to a eight unit cells ($2 \times 2 \times 2$) simulation box and treating the electrostatic interactions via Particle Mesh Ewald (PME) method [103]. Simulated IR spectra are derived squaring the Fourier transform of the total dipole momentum [137–139], with Blackmann-Harris windowing [140], following the Welch method [141].

All simulations follow this procedure:

- 1000 steps of structure minimization, keeping the framework atoms fixed at their crystallographic positions to relax only the cations;

- 1.0 ns run to heat the system from 1 K up to the target value of 300 K (first 0.4 ns) and thermalize it at the fixed temperature of 300 K (last 0.6 ns), all this via rescaling of atom velocities;

- 10.0 ns *NVE* production run to validate the force field parametrization and, in particular, its ability to keep the cations in their correct position;

- 0.4 ns extra run for accumulation of detailed trajectory data (written every 4.0 fs) for subsequent spectral analysis.

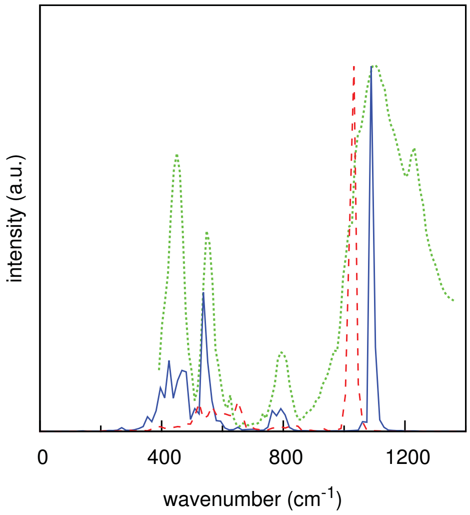


Figure 5.1: IR spectra for silicalite: experimental [111] (dots), initial unoptimized model (dashes), final optimized model (continuous line).

For every given set of force field parameters, we also perform a test run at the temperature of 1000 K checking that the cations keep their correct position (i.e., they do not enter into the sodalite cages).

In Figure 5.1 we compare the experimental spectrum for silicalite with the one coming from our simulations before the optimization process. The greatest differences are found: (1) in the region from 400 to 850 cm^{-1}, where peaks are overlapped and ill defined and (2) in region from 1000 to 1200 cm^{-1}, where the band is shifted toward too low wavenumbers.

To improve the agreement with the experimental spectrum, we first try to understand the influence of every force constant term on the spectrum bands. The most important point is that a given term may affect more than one band, and moreover, the influence of a given force constant on the bands varies, in a nontrivial way, according to the value of the other force constants to which it is coupled. Nonetheless, some trends can be isolated and help us in the optimization: the $\mathrm{O-Si-O}$ bending force constant affects mainly the 400 - 850 cm^{-1} region, while the $\mathrm{Si-O}$ bond term is mainly concerned with the 1000 - 1200 cm^{-1} region.

We adjust the parameters so that the high-frequency band is shifted toward the experimental value of 1100 cm^{-1}. The fact that the band is so narrow can be explained by comparing the analytical form of our new force field with the former one: in the new force field most terms, and in particular the $\mathrm{Si-O}$ bond term, have a harmonic form, while in the former one all terms are anharmonic

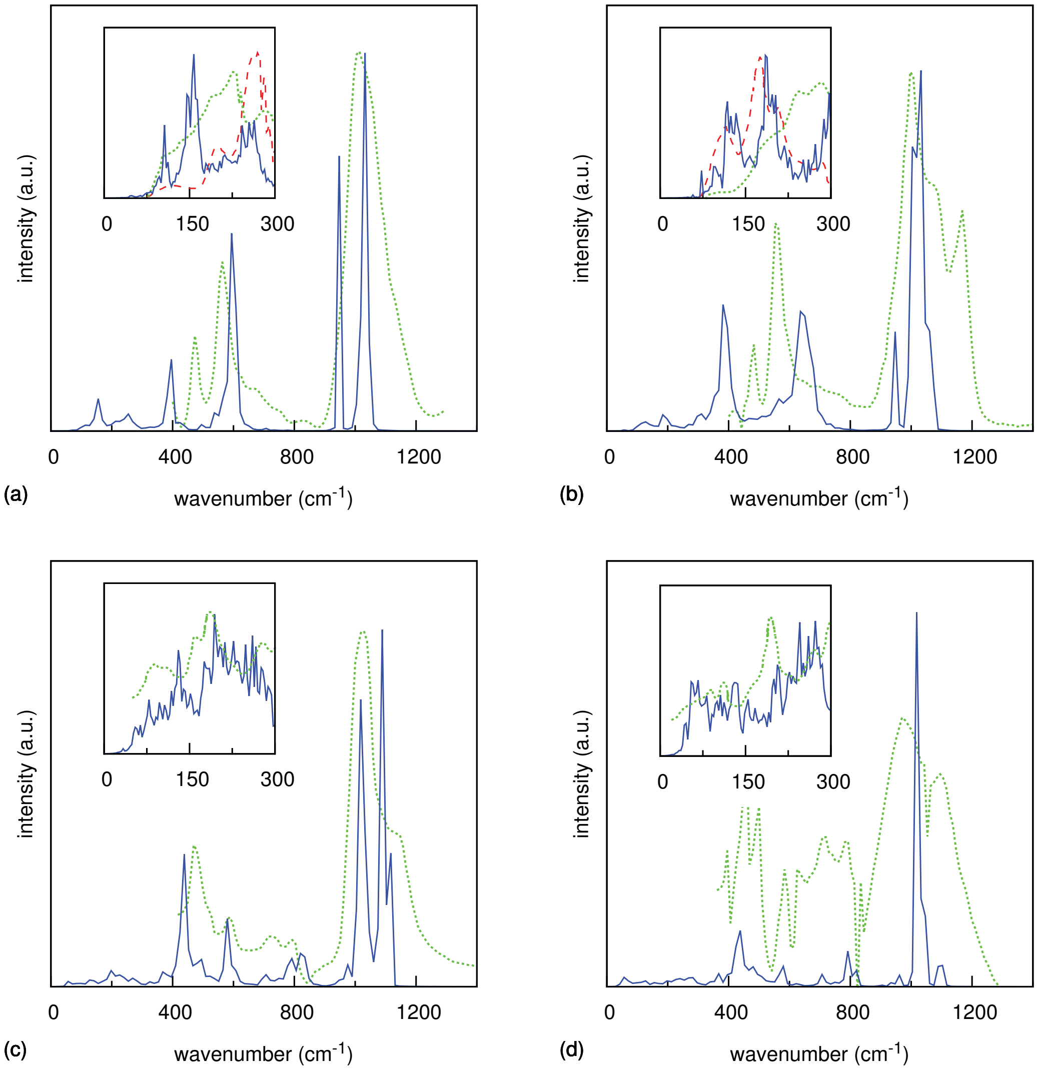


Figure 5.2: IR spectra for Na A (a), Ca A (b), Na Y (c), and Na X (d): experimental [dots, (a,b) [111], (c) [134], (d) [135]] and optimized model (continuous line); in the inset, zoom of far-infrared region related to cations vibrations against experimental data [dots [142] and dashes [143] in (a,b); dots [136] in (c,d)].

(for harmonic oscillators the frequency is independent of the energy).

To enhance the separation of the bands in the 400 - 850 cm^{-1} region, in accordance to the work of Nicholas *et al.* [128], we introduce a UB term between T atoms of adjacent tetrahedra (fictitious $\mathrm{T-(O)-T}$ bond angle interaction), see Equation 5.1. This is particularly useful when the $\mathrm{T-O-T}$ equilibrium angle θ_0 is large (approaching $180°$): in these cases the distance u between the two T atoms is weakly coupled to the oscillations of the angle θ, thus needing the addition of an extra functional term to model properly the two-atom interaction.

The final result for silicalite can be appreciated in Figure 5.1, where the optimized spectrum (continuous line) matches closely the experimental one; the differences in the relative intensities between high and low wavenumber bands are in line with other computational works and are mainly due to the neglect of quantum mechanical corrections in the spectrum computation [137, 138].

In the literature, there have been many attempts to interpret the spectral bands [144, 145], but a clear-cut classification is not possible [136] due to the complex coupling of the framework atom dynamics. A simple qualitative characterization of the bands, with respect to the functional form of our force field, is the following:

950 - 1250 cm^{-1}, $\mathrm{T-O}$ bond stretching modes;

650 - 850 cm^{-1}, $\mathrm{T-(O)-T}$ UB term stretching modes;

550 - 650 cm^{-1}, coupling of $\mathrm{T-(O)-T}$ stretching with $\mathrm{O-T-O}$ bending;

400 - 500 cm^{-1}, $\mathrm{O-T-O}$ angle bending.

Moving to the remaining structures, we determine the parameters related to Al: the bond constant for $\mathrm{Al-O}$ is found to be weaker with respect to the $\mathrm{Si-O}$ constant, and this can be understood based on the different bond energies of the two couples of atoms (which slightly affects the corresponding angles as well). The optimized spectra of these structures can be found in Figure 5.2; an inset zoom of the far-infrared region is also presented to better appreciate the vibrational bands related to the cations. The overall agreement with the experimental data is fairly good.

In Table 5.2 we report all the numerical values of our new force field (vdW parameters for Si, Al, and O, are taken from Lopes *et al.* [110]). Summarizing, the characterizing features of our force field are: strong intratetrahedral $\mathrm{O-T-O}$ angles, weaker intertetrahedral $\mathrm{T-O-T}$ angles, extra UB term between T atoms, and differentiation between $\mathrm{Si-O}$ and $\mathrm{Al-O}$ force constant values. The development of this force field is a first and necessary step toward the detailed study of diffusion phenomena for molecules sorbed within this class of porous media. The improvement coming from a reliable flexible force field is twofold: on one side, the flexibility plays a key role in correctly thermostating the sorbate, thanks to the heat bath effect [146]; on the other, a good reproduction of the lattice vibrations increases the accuracy of the simulated trajectories, especially in case the of tight fitting of the sorbate in the pore openings [147]. Our force field, then, owning the above-mentioned characteristics, will be particularly useful for the computation of accurate diffusion coefficients. In this context, the ability to run on massively parallel architectures (including graphics processing units, GPUs [103]), will extend the MD simulation time scales. This will ultimately allow the study of slow diffusive systems (i.e., diffusivities of the order of 10^{-10} $m^2 \cdot s^{-1}$).

5.2.2 Validation

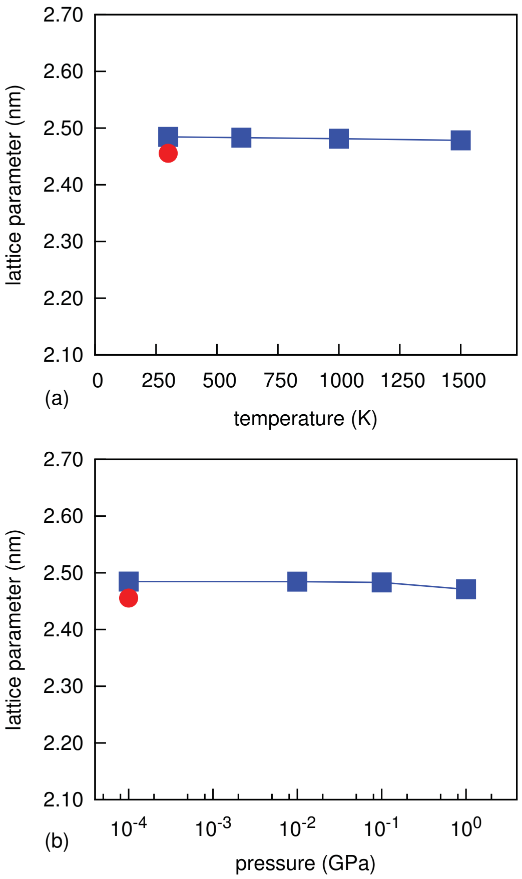


Figure 5.3: Evolution of simulated lattice parameters (squares) for Na A zeolite, over a wide range of temperatures (a) and pressures (b); lines are to guide the eye. Also plotted, crystallographic reference data (circles) [124].

We test now the ability of our model to reproduce properly the crystallographic structures. After equilibrating the system [147], we perform 1 ns *NVE*

Table 5.2: Force field parameters.

bonds	$k_b(\mathrm{kcal}\cdot\mathrm{mol}^{-1}\cdot\mathrm{Å}^{-2})$	$b_0(\mathrm{Å})$
Si-O	300.0	1.61
Al-O	222.0	1.73

angles	$k_\theta(\mathrm{kcal}\cdot\mathrm{mol}^{-1}\cdot\mathrm{rad}^{-2})$	$\theta_0(^\circ)$
O-Si-O	75.0	109.5
O-Al-O	65.0	109.5
Al-O-Si	7.0	149.5
Si-O-Si	7.0	149.5

UB	$k_u(\mathrm{kcal}\cdot\mathrm{mol}^{-1}\cdot\mathrm{Å}^{-2})$	$u_0(\mathrm{Å})$
Al-(O)-Si	30.0	3.18
Si-(O)-Si	30.0	3.12

vdW	$\epsilon(\mathrm{kcal}\cdot\mathrm{mol}^{-1})$	$\sigma(\mathrm{Å})$	$R_{\mathrm{min}}(\mathrm{Å})$
Si	-0.600	3.92	4.40
Al	-0.650	3.92	4.40
O	-0.152	3.15	3.54
Na	-0.159	2.43	2.72
Ca	-0.120	2.44	2.73

partial charges $q(e)$ in each structure					
atom	Sil	Na A	Ca A	Na Y	Na X
Si	2.0	1.85	1.85	1.890	1.760
Al		1.27	1.27	1.274	1.288
O	-1.0	-1.03	-1.03	-1.001	-1.001
Na		1.00		1.000	1.000
Ca			2.00		

Table 5.3: Greatest deviation, for each atom-type group, between mean computed and experimental coordinates in crystallographic units ($\times\mathbf{10^{-2}}$).

Structure	T	O	Na	Ca
Sil	3.74	6.12		
Na A	0.10	0.45	3.32	
Ca A	0.11	0.72		0.26
Na Y	0.14	0.61	2.60	
Na X	0.50	0.75	3.96	

simulations and post process the trajectories evaluating the atomic coordinates by reversing the symmetry transformations. This way, we can compute the mean crystallographic coordinates of the asymmetric unit atoms and their distributions, and also detect the discrepancy with respect to the experimental crystal symmetry, which is revealed by multimodal or asymmetric coordinate distributions. In Table 5.3 we report the greatest deviations of crystallographic coordinates for each investigated structure (rows) and for each atom type (columns). As can be seen, our model reproduces well the experimental data.

At the same time, we analyze qualitatively the shape of the distributions of the corresponding (i.e., symmetrically equivalent) atom positions: these are found to be Gaussian. Note that some deviations from a symmetric shape are found for the cations, due to the anharmonic character of the potential terms (i.e., vdW plus electrostatic) and the existence of various energetically equivalent sorption sites among which the cations are redistributed during the system equilibration phase; clearly these effects arise from the use of an approximated model force field on a size-limited system [111].

For Na A, we also check the stability of our model framework, over a wide range of temperatures and pressures, performing various 1 ns *NPT* (isothermal-isobaric ensemble) simulations with the barostat fluctuations controlled via Nosé-Hoover Langevin piston [103]. Control parameters are: piston oscillation period 0.2 ps, piston decay time 0.1 ps, and damping coefficient for temperature coupling set to 10.0 giving a decay time of 0.1 ps. Our model framework ensures an excellent structural stability to the system (see Figure 5.3): in plot (a) a slightly negative thermal expansion [148–151] can be appreciated, while in plot (b) the shrinkage in response to external increasing pressure [152, 153] is shown. The discrepancy between our model lattice parameter at 300 K and 10^5 Pa is about 1%.

Finally, we give a rough estimate of the speedup attainable using our new force field with a parallel simulation package like NAMD in a small size Beowulf cluster. Running on 16 cores (4 Intel Xeon E5420 processors with Infiniband) a 1 ns simulation of a $2 \times 2 \times 4$ silicalite system (4608 atoms) takes about 1 h. On the other hand, a 1 ns simulation of the same system, computed with the former force field on a single core, takes about 100 h. The same trend holds also for the remaining structures.

5.3 Conclusions

In this work, a new force field has been developed enabling fast molecular dynamics simulations in flexible aluminosilicates and, thus, extending the time and space scales accessible to classical MD simulations. The structures here investigated are silicalite, Na A, Ca A, Na Y, and Na X, chosen to ensure a good degree of force field portability, allowing an extension to affine structures with minimal effort. We adopted a CHARMM-type functional form which allows, using the NAMD package, the simulation of a 1 ns trajectory per wall clock hour in systems consisting of about 4000 atoms, running over 16 cores of small Beowulf clusters.

The new force field has been optimized by carefully tuning the simulated structures and IR spectra to experimental data. The resulting parametrization allows correct modeling of the system dynamics, without introduction of spurious deformations. Moreover, the structural stability of model Na A over a wide range of temperatures and pressures has been successfully tested.

This work is a starting point for future studies of sorbed molecules in zeolites, especially for the development of more reliable coarse-grained models which will further expand the accessible time and space simulation scales.

Chapter 6

Optimization of Molecular Dynamics Force Fields via Force Matching of *ab-initio* data

One of the most widespread tools to investigate the dynamics of sorbed molecules within microporous materials, keeping into account also the flexibility of the framework [146], is the classical Molecular Dynamics (MD) technique [21,22]. This kind of simulations can follow the time evolution of a million atom system up to the microseconds scale. The drawback is the need to feed the program with a *force field* (FF) ruling the atoms interactions, on which will ultimately depend the quality of the results.

At the same time, the field of *ab-initio* molecular dynamics computations is rapidly growing. These give accurate results without need of an explicit FF (requiring as input only atoms types and initial positions). The drawback of this technique, on the other hand, is the large computational cost which becomes prohibitive at the time and space scales accessible to classical MD.

To exploit the advantages of both techniques, we could use short but detailed *ab-initio* computations to develop accurate FFs for classical MD, by means of the *force matching* method [154–163].

Aim of this work is to investigate the potential of the force matching technique and efficiently apply it to obtain the FF constants that more closely reproduce the reference system dynamics.

Having our group of research recently published two papers dealing with the refinement of FF parameters for classical MD simulations in both Silicalite [164] and ZIF-8 [147], porous materials and their sorbates become good candidates for this study. In those papers, the parameters optimization has been done with a trial and error procedure which is tedious and time consuming (more than one month for each structure). Moreover, this approach becomes practically unfeasible as the crystals complexity grows (e.g., ZIF-8 has already 46 bonded interaction terms to be tuned). The study of such systems requires a great amount of time even via the force matching technique, for this reason we made a big effort in improving the overall implementation performance.

The final force fields will be based on the CHARMM [109] functional form, tuned via an automated optimization procedure. In this work we will focus on the bonded part of the force field, which is responsible for the vibrational spectrum of modeled molecules and crystals, and can thus be accurately validated on the basis of this macroscopic property. In the CHARMM formalism, in fact, the exclusion policy has the effect of zeroing the weight of the nonbonded interactions with respect to the molecular frequencies of vibration (still, partial charges and vdW parameters taken from the literature can be fully incorporated on top of our bonded parametrization).

In the first part of this chapter we will illustrate the theory behind the force matching technique and our implementation of the whole method. In the second part we will apply the procedure to systems of increasing complexity, starting with CH_4, then CO_2, Silicalite and finally ZIF-8.

6.1 Theoretical background

Force matching technique. With this procedure the interaction parameters of a model system (e.g., a molecule or a crystal) are adjusted until they reproduce, within the wanted degree of accuracy, the forces of a given reference system [154]. In general, the reference is an highly detailed and thus accurate set of data, still very expensive from a computational point of view (here, *ab-initio*). The model system is in general a coarse-graining of the reference one, where some details are averaged out to attain high computational speed (here, classical MD).

The core of the whole procedure is the minimization of the sum of the squared residuals (merit function) between reference (F) *ab-initio* forces and model (f) MD forces:

$$\chi = \sum_{j=1}^{S} \sum_{i=1}^{3N} (F_{i,j} - f_{i,j})^2, \tag{6.1}$$

where N is the number of atoms in the system, clearly each atomic force has 3 components (x, y, z), and S is the number of snapshots (system configuration frames) taken with an arbitrary stride.

In accordance to the original work of Ercolessi and Adams [154], we evaluate the quality of the match looking at

$$h = \sqrt{\chi/(3NS)},$$

the root mean square deviation per atomic force component, and compare it, h/g, against

$$g = \sqrt{\frac{1}{3NS} \sum_{j=1}^{S} \sum_{i=1}^{3N} F_{i,j}^2},$$

the root mean square of the reference forces, representing their magnitude.

In applying the force matching procedure to the optimization of classical FFs, one needs to fulfill three requirements: 1. *the system dynamics should be well reproduced*, here this is realized minimizing the merit function; 2. *the system structure should be preserved*, this is accomplished setting the FF equilibrium values (distances and angles) to the ones taken from the averaging of the *ab-initio* trajectory; 3. *the system thermodynamics should be satisfied*, this is implicit in the usage of the CHARMM functional form.

DFT computations. *Ab-initio* MD simulations are more accurate than the classical ones, yet they run about 1000 times slower. For this reason they are not suitable to follow the time evolution of some macroscopic properties like self-diffusion. They have the big advantage of not requiring a structure dependent force field, but just knowledge of atom types and positions. From this comes the idea of the force matching, where the system forces are stored over a short accurate trajectory, trying subsequently to obtain the same forces during a classical MD simulation.

In this work, the reference data are obtained performing Born-Oppenheimer Molecular Dynamics (BOMD) simulations using the CP2K open source code [33, 165–167]. The energy of the system is evaluated via Density Functional Theory (DFT) [8, 9] computations in the framework of the Gaussian and Plane Waves (GPW) [168] method.

The accuracy of DFT computations is continuously improving, thanks to the refinement of theoretical models and the growing of computational power [11]. This fact makes reasonable the expectation that DFT results will approach more and more the experimental limit. It becomes clear, then, that the force matching technique bridging detailed but expensive DFT computations and fast (but often based on too approximated FF) MD simulations will become more and more valuable.

Classical MD force field development. The FFs developed in this work rely on the CHARMM functional form:

$$\begin{aligned} E_{\text{pot}} = &\sum_{\text{bonds}} k_{\text{b}}(b-b_0)^2 \\ &+ \sum_{\text{angles}} k_\theta(\theta-\theta_0)^2 + \sum_{\text{UB}} k_{\text{u}}(u-u_0)^2 \\ &+ \sum_{\text{dihedrals}} k_\psi(1+\cos(n\psi-\delta)) \\ &+ \sum_{\text{impropers}} k_\omega(\omega-\omega_0)^2 \\ &+ \sum_{\text{vdW}} \epsilon\left[\left(\frac{R_{\text{min}_{ij}}}{r_{ij}}\right)^{12} - 2\left(\frac{R_{\text{min}_{ij}}}{r_{ij}}\right)^{6}\right] \\ &+ \sum_{i<j} \frac{q_i q_j}{\epsilon r_{ij}}. \end{aligned} \tag{6.2}$$

where the first five terms refer to the bonded interactions, namely bonds, angles, Urey-Bradley (UB), dihedrals, and impropers, while the last two refer to the nonbonded ones, Lennard-Jones and Coulomb [164]. We choose this functional form because it is suitable to model the systems here under investigation and because it is implemented in most modern MD simulation packages, exploiting the full power of massively parallel architectures and even GPUs [103–108].

It is important to remark that in CHARMM FFs an *exclusion policy* is employed. Here the van der Waals and electrostatic interactions between $1-2$ and $1-3$ connected atoms are implicitly taken into account within the bonded terms (without explicit computation of Lennard-Jones and coulombic terms). For the $1-4$ connected atoms, instead, the Lennard-Jones and coulombic terms are computed and scaled with appropriate factors [147]. This exclusion policy, then, enables a clear distinction between bonded and nonbonded interactions, splitting them into two complementary parts: a short range and a long range one.

In particular, the bonded interactions are much stronger than the nonbonded ones (at least one order of magnitude). For this reason, the bonded force constants can be optimized independently from the nonbonded ones, with negligible loss of accuracy (see Section 6.2.1).

On this basis, here we focus our attention on developing an automated procedure to obtain reliable bonded parametrizations. These can be subsequently coupled with opportune nonbonded parameters: for the time being taken from the literature, and, in the future, directly optimized starting from independent *ab-initio* data (e.g., the electric field for the coulomb interaction).

In support to this approach, one can consider the vibrational spectra computed with FFs optimized using only the bonded interactions and FFs where also the nonbonded ones are taken into account: the frequencies of the resulting spectra are practically unaffected (see Figures C.5 to C.8 in Appendix C).

6.1.1 Implementation

For all cases studied in this work we followed a general strategy:

- BOMD simulation to obtain atoms positions and corresponding forces;
- force matching to obtain the classical MD force field parameters;
- validation of the classical force field against the BOMD reference.

BOMD simulation. After generating [120] the system structure starting from experimental data, and properly thermalizing at 300 K, we follow the evolution of the system in the NVT ensemble (weak coupling with CSVR thermostat [169]) for 5 ps (10^4 steps with a time step of 0.5 fs). PBE [13] functional along with GTH pseudopotential [170–172], and GTH basis sets [33] are used throughout this work, giving the best compromise between accuracy and computational cost for the investigated systems. More specifically, basis sets for each atom kind are TZV2P, except for Zn in ZIF-8 (TZVP) in accordance to previous works [173, 174]. Dispersion interactions are taken into account as well, using the DFT-D3 method [175]. The energy cutoff is 700 Ry. The system is fully periodic. The accuracy for the SCF is 10^{-7}.

Force Matching. To perform the merit function χ minimization we wrote a Python program [176, 177] which relies on the L-BFGS-B [178, 179] algorithm as implemented in the *SciPy* [180, 181] minimize module.

The program requires as input an initial guess of the parameters to be matched and the interval over which these parameters can span. Throughout this work, the initial values are taken as the mid point of the interval.

The minimizer computes the value of the merit function for the current parameters and adjusts them until the convergence criterion is met: either the relative change in merit function between two subsequent function evaluations, or the greatest component of the projected merit function gradient, becomes smaller than the dsired value (in SciPy nomenclature, *factr* = 10^2 and *pgtol* = 10^{-8}, respectively).

In order to evaluate the merit function for a given set of parameters, we need to compute the classical MD forces to be compared with the BOMD reference ones. This is done using LAMMPS, built as a python library [105].

This approach is very general and can be applied to a wide range of problems, the only limit being the functional forms supported by LAMMPS (which actually are many).

Before starting the real analysis, the procedure has been tested matching the forces of a reference trajectory generated via classical MD with a known FF. The code has proven to be able to recover, within machine precision, the full reference FF.

Validation. The parameters optimized during the force matching procedure are tested performing a classical MD simulation using the NAMD [103] package (which is faster on our hardware but less flexible than LAMMPS, allowing a cross check of the parametrization). The time step used is 0.5 fs for a total simulation time of 320 ps in the *NVE* ensemble, after having properly thermalized the system at a temperature of 300 K. The thus obtained trajectory is then post processed to compute the IR spectrum squaring the Fourier transform of the total dipole moment [137–139], with Blackmann-Harris [140] windowing according to the Welch method [141]. The same procedure is followed also for the BOMD data, over a 10 ps trajectory, computing the total dipole moment from the Berry phase [182], to compare the two resulting spectra.

Note that the classical MD trajectory is here produced using only the optimized bonded interactions. In order to compute the total dipole moment, then, one needs to associate reasonable partial charges to the atoms, which do not affect the frequencies of the resulting IR spectrum and can thus be arbitrarily chosen.

6.1.2 Force matching speedup

The force matching procedure presented in the previous section may be very heavy from a computational point of view (depending on the number of parameters to be optimized and the size of the system). Most of the computational time is spent calling the LAMMPS library (to compute the classical MD forces) at every evaluation of the merit function. Things get even worse, the L-BFGS-B optimization algorithm we used, in fact, requires knowledge of the merit function gradient, which has to be numerically estimated calling LAMMPS $(N_p + 1)$ times at each χ evaluation, being N_p the number of working parameters. Moreover, the gradient is computed using the same fixed delta for all parameters, Δp. This can be a source of error, due to the different orders of magnitude of the parameters themselves (e.g., bonds are in the order of hundreds, while angles in the order of tens).

To overcome these limitations and increase both speed and accuracy, we sacrifice some flexibility making the code *functional form* specific. Thanks to the CHARMM FF additivity, in fact, the contribution to the total force deriving from each term of Equation 6.2 can be computed independently from the others.

Envision now the vector $\mathbf{f}$ containing all the $3NS$ force elements $f_{i,j}$ appearing in Equation 6.1 (one element for each of the N system atoms, for the 3 cartesian components, and for the S frames forming the whole reference trajectory). Exploiting the FF additivity property, this vector $\mathbf{f}$ can be decomposed into N_p individual vectors, one for each parameter to be optimized, so that the following holds:

$$\mathbf{f} = \sum_{m=1}^{N_p} \mathbf{f}_m. \tag{6.3}$$

With this in mind, before starting the main matching loop, we compute and store each of the N_p vectors; here, to compute the m-th one, we call LAMMPS (over the reference trajectory) setting to 0.0 all the FF parameters, except for the m-th one which is set to a convenient value of 1.0.

After calling LAMMPS just N_p times, then, we obtain all the individual force vectors (associated to a unitary value of their corresponding parameter). From here on, there will be no need to call LAMMPS anymore: at each change of the m-th parameter performed by the matcher, we will simply multiply the corresponding m-th force vector by the new parameter value. Thus, at each merit function evaluation, we can reconstruct the model forces with:

$$\mathbf{f} = \sum_{m=1}^{N_p} p_m \mathbf{f}_m. \tag{6.4}$$

As a working example, we can envision a given molecule, and focus on the forces arising from a specific bond type. We call LAMMPS, setting $k_\mathrm{b} = 1.0$ for the first energy term in Equation 6.2, $k_\mathrm{b}(b - b_0)^2$, and store the output forces in a vector $\mathbf{f}_\mathrm{b}$. When the minimizer will choose a new k_b value, we will compute the corresponding forces by the simple multiplication $k_\mathrm{b}\mathbf{f}_\mathrm{b}$.

This procedure can be applied, with minor algebraic manipulation, also to forces having nonlinear dependence over the working parameters (e.g., b_0 in the above example).

The approach here depicted (with just N_p LAMMPS calls) allows a speed up of the whole procedure of 1000 times or more (according to the investigated system).

Table 6.1: Methane equilibrium values, parameters ranges and final optimized values.

	eq^a	range^b	final^b
bond			
C−H	1.099	[50.0, 1200.0]	333.74
angle			
H−C−H	109.47	[10.0, 200.0]	28.52
UB			
H−(C)−H	1.793	[0.0, 200.0]	18.39

^{a}Bond and UB (Å), angle $(^\circ)$.

^{b}Bond and UB $(\mathrm{kcal \cdot mol^{-1} \cdot Å^{-2}})$, angle $(\mathrm{kcal \cdot mol^{-1} \cdot rad^{-2}})$.

6.2 Results and discussions

6.2.1 Methane

We start our investigation with a molecule of great interest for the energy and the environment, widely studied, both in computer and laboratory experiments [183–186].

The 5 ps (10^4 frames) BOMD simulation, to get the reference forces, is performed over a system consisting in 12 CH_4 molecules, using a cubic box of side 1.69856 nm with periodic boundary conditions (PBC) applied along all the three main coordinates directions. The choice of the box size is related to the crystallographic lattice parameter of the ZIF-8 unit cell, in which methane sorption and diffusion has been successfully probed by PFG NMR experiments [187].

Equilibrium values for the C−H bond and H−(C)−H UB term are obtained by post processing the BOMD trajectory, while the H−C−H angle is set according to the tetrahedron geometry. All values are reported in Table 6.1.

Moving then to the force matching, the associated working parameters are the k_b, k_θ, and k_u force constants (while nonbonded interactions are set to zero). The ranges over which these parameters can be changed by the matcher and final results are also reported in Table 6.1. We recall that the parameters values used as initial guess in the matching procedure are the mid points of the corresponding ranges. These ranges are wide on purpose, to test the ability of the matcher to find the right parameters without any human bias.

The matching procedure ended with $h = 1.77\ (\mathrm{kcal \cdot mol^{-1} \cdot Å^{-1}})$. Comparing this to the magnitude of the reference forces gives $h/g = 0.11$ (in a gross estimate this means that about 68% of all the 1.8×10^6 matched force components differs less than 11% from their reference). The wall time necessary for the force matching to converge over a 10^4 steps trajectory, is about 1 minute on a common desktop pc.

Running a classical MD simulation with the obtained parameters we compute the IR spectrum (using dummy charges to get the dipole moment), shown in Figure 6.1, to be compared with the reference BOMD spectrum, also plotted. The overall agreement is excellent.

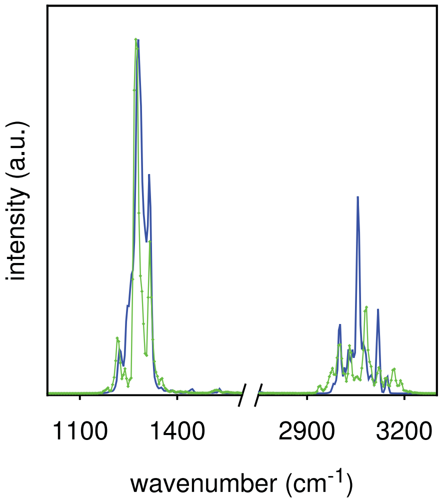


Figure 6.1: Methane IR spectrum from classical MD with optimized parameters (solid line), compared with the BOMD reference one (dots).

Comparison of our reference BOMD spectra with the experimental ones is out of the scope of this work, being related to the improvement of the DFT computations themselves. Nonetheless, these plots are in qualitative agreement with the experiments (see Figures C.1 to C.4 in Appendix C).

The above procedure is the result of extensive tests on the matcher capabilities. The underlying assumptions are now briefly discussed.

Influence of nonbonded interactions. Here we show that the nonbonded interactions have negligible influence over the bonded parameters during the optimization process. This is true, particularly in the context of the CHARMM formalism, thanks to its exclusion policy.

We repeat the previously described force matching procedure, this time setting the vdW and charge values to the ones taken from Nicholas and coworkers [183].

Comparing the resulting parameters to the ones obtained without nonbonded interactions, the change in final values is less than 1%. Clearly, the IR spectra computed with the two parametrizations are in close agreement, see Figure C.5 in Appendix C.

Equilibrium values. Several options for the choice of equilibrium b_0, θ_0, and u_0, exist: one is to employ the output values from a geometry optimization procedure at the DFT level (or better), another is the use of the experimental values, a third is to extract them averaging the BOMD trajectory. This last option gives the lowest merit function value, χ, and this is the reason why it has been preferred.

There is actually a fourth option, i.e., to let the matcher optimize also the equilibrium values. The drawback of this approach is the arising of a competition between bond and UB equilibrium distances which leads to unphysical values (i.e., zeroing of u_0 and anomalous growth of b_0).

Convergence with respect to the trajectory length. At this point, the most expensive part of the whole procedure, from a computational point of view, is the acquisition of detailed BOMD reference data. In particular, to produce 1 ps of trajectory for a system containing about 300 atoms, requires about 2 days of wall time on 64 cores (Intel Xeon E5420 @ 2.5 GHz). For this reason, it is important to know what is the minimum trajectory length needed to obtain a well converged parametrization.

We repeat the force matching procedure, using an increasing number of frames (i.e., 1, 10, 100, 1000) to study the parameters convergence behavior, in comparison with the results obtained with 10^4 frames, taken as a reference. As shown in Figure 6.2, the parameters mean absolute percentage deviation (MAPD), $(100/N_p)\sum_{i=1}^{N_p}|(p_i^{\text{ref}} - p_i)/p_i^{\text{ref}}|$ where p_i^{ref} is the optimized value for the i-th reference parameter, drops to less than 1% at 1000 frames. This implies that a good parametrization can be obtained already with a BOMD trajectory of 500 fs. Clearly, for convergence, a long enough simulation time (given by the number of frames times the sampling interval) is essential.

It is important to note that the convergence rate depends on the parameters magnitude. As a rule of thumb, the stronger the interaction the fastest the convergence; in general, then, bonds are the fastest, followed in order by UB, angles, and four-body interactions.

Annealing. The L-BFGS-B is a widely used algorithm to find local minima. To ensure that the solution found by our code is the best one for a given set of boundaries, we executed also a simulated annealing procedure (again relying on

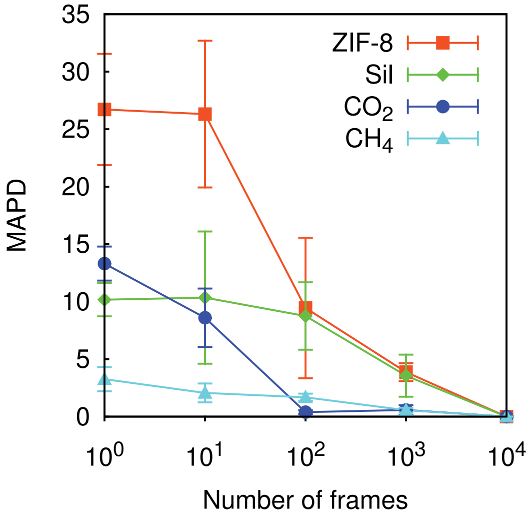


Figure 6.2: MAPD convergence as a function of frames number, with respect to 10^4 frames. Each point is the mean of 3 optimizations over independent trajectory blocks, with error bars representing the standard deviation.

the SciPy tools) decoupling the final results from the initial parameters values. In agreement to the observations of other groups [154], the annealing is not necessary for the problems here investigated (i.e., no change in the final results).

6.2.2 Carbon dioxide

Moving to CO_2 [188–192], we repeat the procedure used for methane, keeping the same box size and number of molecules. After obtaining the BOMD reference data and computing the mean equilibrium values, except for the angle which is set to its geometric value, we launch the force matching. The relevant parameters are reported in Table 6.2.

In Figure 6.3 we compare the classical MD spectrum obtained using the optimized parameters (dipole moment computed with dummy charges) with the reference BOMD one. Also for CO_2, the agreement is excellent. This is confirmed also by the exiting $h = 1.86$ $(\mathrm{kcal} \cdot \mathrm{mol}^{-1} \cdot \mathrm{Å}^{-1})$ and $h/g = 0.096$.

Peculiar in this parametrization is the inclusion of the UB term. Running the matcher without it, in fact, gives a $h = 2.38$ $(\mathrm{kcal} \cdot \mathrm{mol}^{-1} \cdot \mathrm{Å}^{-1})$ and $h/g = 0.12$, meaning that the latter optimized model is less accurate in reproducing the reference one. This occurs because the equilibrium angle is close to $180°$ and, in this case, the bend term is unable to efficiently model the O—O interaction, and the need of an UB term becomes evident (see Figure C.9 in Appendix C).

Table 6.2: Carbon dioxide equilibrium values, parameters ranges and final optimized values.

	eq[a]	range[b]	final[b]
bond			
C−O	1.178	[500.0, 1200.0]	979.46
angle			
O−C−O	180.00	[10.0, 200.0]	52.75
UB			
O−(C)−O	2.353	[50.0, 200.0]	86.77

[a] Bond and UB (Å), angle (°).

[b] Bond and UB ($\mathrm{kcal \cdot mol^{-1} \cdot Å^{-2}}$), angle ($\mathrm{kcal \cdot mol^{-1} \cdot rad^{-2}}$).

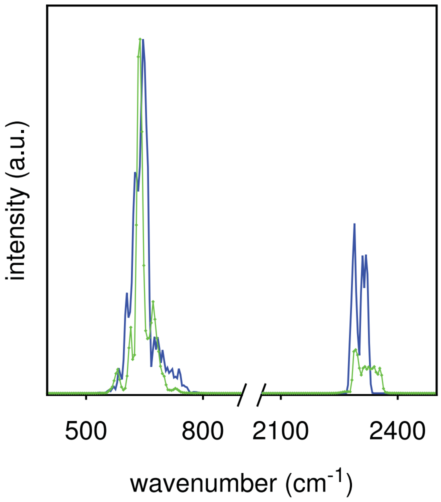


Figure 6.3: Carbon dioxide IR spectrum from classical MD with optimized parameters (solid line), compared with the BOMD reference one (dots).

Table 6.3: Silicalite equilibrium values, parameters ranges and final optimized values.

	eq[a]	range[b]	final[b]
bond			
Si−O	1.636	[50.0, 1000.0]	274.10
angles			
O−Si−O	109.43	[0.0, 200.0]	29.51
Si−O−Si	144.95	[0.0, 200.0]	24.78
UB			
O−(Si)−O	2.669	[0.0, 200.0]	1.06
Si−(O)−Si	3.108	[0.0, 200.0]	29.29

[a] Bond and UBs (Å), angles (°).

[b] Bond and UBs ($\mathrm{kcal \cdot mol^{-1} \cdot Å^{-2}}$), angles ($\mathrm{kcal \cdot mol^{-1} \cdot rad^{-2}}$).

6.2.3 Silicalite

We test now the ability of our procedure to mimic the Silicalite dynamics. This microporous crystal is of great industrial interest [62, 111, 119, 121, 128, 183]. Its unit cell contains 288 atoms with lattice constants $a = 2.0022$ nm, $b = 1.9899$ nm, and $c = 1.3383$ nm [122, 123].

In Table 6.3 the parameters relevant for the matching procedure are reported.

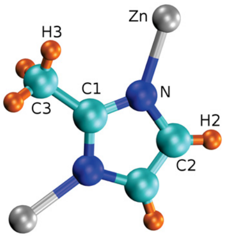
In Figure 6.4 a comparison between spectra from optimized parameters (with dummy charges) and from reference BOMD data is presented. The overall agreement is fair: the shape of the spectrum is well caught, with a small frequency underestimation of the midrange bands.

The exiting $h = 5.73$ $(\mathrm{kcal \cdot mol^{-1} \cdot Å^{-1}})$ and $h/g = 0.29$ are large compared to the other systems here investigated. This suggests that some terms may be missing from the FF functional form, for example a term ruling the interactions among all the atoms forming a given Silicalite window: these atoms, in fact, are close neighbors from the spatial point of view, but far apart (much more than $1-4$) from the connectivity point of view. This refinement is beyond the scope of our work, but it may be important for further modeling of other crystal properties, apart from the vibrational spectrum.

6.2.4 ZIF-8

We finally move to a recently synthesized structure [66, 67, 190], called zeolitic imidazolate framework 8 (ZIF-8), which is currently receiving great attention from the scientific community [189, 193–196]. This structure has a cubic unit

Table 6.4: ZIF-8 initial parameters, ranges, and final optimized values.

					*bonds*	eq[a]		range[b]	final[b]
					C2−N	1.386		[50, 1000]	289.65
					C2−H2	1.088		[50, 1000]	369.75
					C3−H3	1.102		[50, 1000]	321.77
					N−Zn	2.017		[50, 1000]	67.62
					C2−C2	1.377		[50, 1000]	402.32
					C1−N	1.355		[50, 1000]	337.26
					C1−C3	1.498		[50, 1000]	225.63
angles	eq[a]		range[b]	final[b]	*UB*	eq[a]		range[b]	final[b]
C1−N−Zn	125.721		[0, 200]	12.15	C1−(N)−Zn	3.014		[0, 200]	10.32
C2−N−Zn	127.028		[0, 200]	12.23	C2−(N)−Zn	3.057		[0, 200]	7.35
C1−N−C2	106.252		[0, 200]	46.33	C1−(N)−C2	2.193		[0, 200]	111.65
N−Zn−N	109.360		[0, 200]	5.24	N−(Zn)−N	3.288		[0, 200]	11.37
H3−C3−H3	107.741		[0, 200]	27.18	H3−(C3)−H3	1.779		[0, 200]	18.67
C2−C2−N	107.995		[0, 200]	33.58	C2−(C2)−N	2.235		[0, 200]	99.05
C2−C2−H2	130.034		[0, 200]	19.39	C2−(C2)−H2	2.236		[0, 200]	14.78
N−C1−N	111.169		[0, 200]	32.34	N−(C1)−N	2.236		[0, 200]	107.37
C3−C1−N	124.197		[0, 200]	39.07	C3−(C1)−N	2.522		[0, 200]	30.61
H2−C2−N	121.317		[0, 200]	31.58	H2−(C2)−N	2.160		[0, 200]	20.43
C1−C3−H3	110.963		[0, 200]	36.31	C1−(C3)−H3	2.153		[0, 200]	19.16
dihedrals	shift[a]	n	range[b]	final[b]	*dihedrals*	shift[a]	n	range[b]	final[b]
H2−C2−C2−N	180	2	[0, 50]	3.55	H2−C2−N−C1	180	2	[0, 50]	3.65
N−C1−C3−H3	180	2	[0, 50]	0.27	C2−C2−N−C1	180	2	[0, 50]	6.64
H2−C2−N−Zn	180	2	[0, 50]	0.00	C3−C1−N−Zn	180	2	[0, 50]	0.00
C3−C1−N−C2	180	2	[0, 50]	3.64	N−C1−N−Zn	180	2	[0, 50]	1.26
N−C1−N−C2	180	2	[0, 50]	10.77	C1−N−Zn−N	0	3	[0, 50]	0.00
H2−C2−C2−H2	180	2	[0, 50]	0.34	*impropers*	eq[a]		range[b]	final[b]
N−C2−C2−N	180	2	[0, 50]	15.33	N−C2−H2−C2	180		[0, 50]	0.00
C2−C2−N−Zn	180	2	[0, 50]	0.82	N−C1−N−C3	180		[0, 50]	6.99
C2−N−Zn−N	180	3	[0, 50]	0.10	C1−C2−N−Zn	180		[0, 50]	0.00

[a] Bonds and UBs (Å), angles, dihedrals, and impropers (°).

[b] Bonds and UBs ($\mathrm{kcal \cdot mol^{-1} \cdot Å^{-2}}$), angles, dihedrals, and impropers ($\mathrm{kcal \cdot mol^{-1} \cdot rad^{-2}}$).

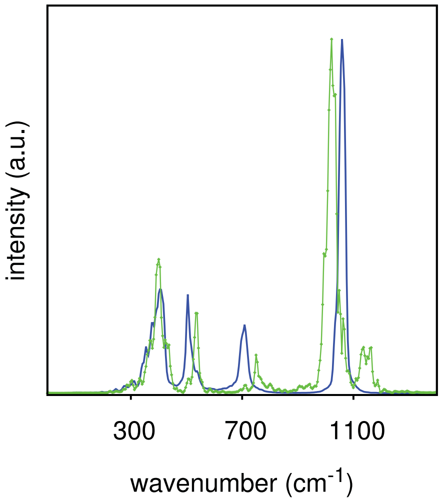


Figure 6.4: Silicalite IR spectrum from classical MD with optimized parameters (solid line), compared with the BOMD reference one (dots).

cell of side 1.69856 nm, formed by 276 atoms of 4 different atomic types, namely: H, C, N, and Zn.

A CHARMM type FF requires knowledge of 46 parameters to properly model the system dynamics. The dihedral part of the FF requires great care: a close inspection of the BOMD trajectory, in fact, is needed to find the correct multiplicity n and phase δ. We choose a value of $n = 2$ and $\delta = 180\,^\circ$ for most dihedrals. This value is particularly convenient, since the phase is the same for all of them. Looking at the BOMD trajectory, a parametrization with $n = 1$ seems also feasible, but in this case the correct phase ($0\,^\circ$ or $180\,^\circ$) should be specified; this approach has been avoided, since it gave no improvement over the match quality. Only for the C1−N−Zn−N and C2−N−Zn−N dihedrals, a value of $n = 3$ has been used, with a phase of $0\,^\circ$ and $180\,^\circ$, respectively (see Figures C.11 and C.12 in Appendix C). Regarding the improper angles, we choose the atoms order so that the first three always belong to the imidazole ring (i.e., N−C2−H2−C2, N−C1−N−C3, C1−C2−N−Zn).

All the parameters optimized during the force matching are reported in Table 6.4. The final results are attained for $h = 3.24\ (\mathrm{kcal \cdot mol^{-1} \cdot \AA^{-1}})$ and $h/g = 0.15$, which is remarkable for such a complex system. The wall time for convergence over 10^4 frames, is less than 3 hours. This can be reduced to just 7 minutes, with negligible loss of accuracy, striding the reference trajectory by keeping one frame every ten (practically, no change in the amount of phase space explored).

In Figure 6.5 we compare the model vibrational frequencies (with dipole moment computed with a reasonable set of partial charges [147, 174]) and the reference ones. The agreement is very good.

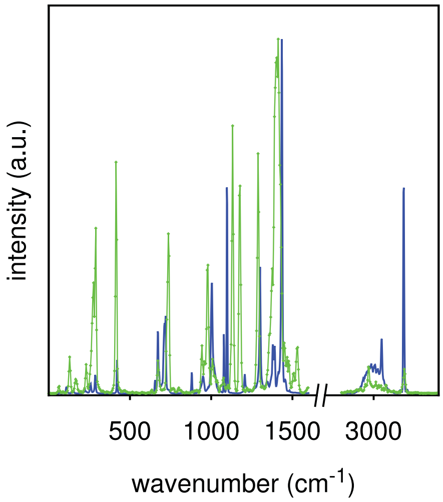


Figure 6.5: ZIF-8 IR spectrum from classical MD with optimized parameters (solid line), compared with the BOMD reference one (dots).

Looking at Figure 6.2, we see that ZIF-8 rate of convergence is slower than that of other systems, this is mainly due to the weaker four-body interactions, still the mean absolute percentage deviation of the parameters drops to 3% with 1000 frames.

6.3 Conclusions

In this work, a new implementation of the force matching technique, to obtain accurate classical force field parameters starting from detailed *ab-initio* data, has been presented. Thanks to its speed, it can be applied to a wide class of materials, of considerable complexity. In particular, this is the first time that a force matching procedure is applied to Silicalite and ZIF-8 microporous crystals, where the number of parameters to be optimized can be greater than 40. In spite of this large number, the task can be successfully accomplished in just minutes on a standard desktop pc. The quality of the final results has been assessed comparing the frequencies of the IR spectra, computed with the optimized parameters, against the reference BOMD ones. The overall agreement between model and reference systems is excellent.

Until now, due to the huge cost in terms of time and computational resources to properly parametrize a force field, the portability among various similar structures was a key feature, to be achieved even sacrificing some accuracy. In this work, instead, we have shown that it is feasible to optimize a specific force field for each investigated structure, in a reasonable amount of time, being the production of the reference BOMD trajectory (a few days of simulation on a small cluster), the most time consuming part.

Chapter 7

Conclusions

Computer modeling of large complex chemical systems is a challenging task. Most of the difficulties stem from the fact that phenomena governing the behavior of such systems, happen in a wide range of spatial and temporal scales. As a consequence, it is not possible to effectively simulate those systems with just a single simulation technique.

This work has focused on the improvement of several simulation methods, with the aim of both improving the accuracy, and extending the space and time scales accessible to such techniques. The resulting methods have been applied for the study of large size microporous systems. Nevertheless, they can be, without too much effort, employed in the study of other systems.

At first, a parallel implementation of the kinetic Monte Carlo algorithm has been applied to the study of Benzene diffusion in zeolites, showing that, with a proper tuning of the simulation parameters, it is possible to reach high efficiencies and thus effectively extend the space and time scales. Then, starting from a Partitioning Cellular Automaton, a simplified coarse-grained model of the hopping process of a tagged particle in a confined lattice system has been developed, providing an accurate reproduction of the memory effects in the self-diffusion at a minimum computational cost.

Being the accuracy of this kind of methods dependent on the quality of the parametrization, the attention was moved to a more detailed technique, the molecular dynamics, with the development of a new force field for simulations in flexible aluminosilicates. The functional form of such force field has been chosen in such a way that it could be used in molecular dynamics packages which can exploit massively parallel architectures, again extending the space and time scales accessible. Finally a new, fast implementation of the force matching technique has been proposed. Thanks to its speed, it has been possible to obtain reliable classical force fields, tailored to each specific structure, also for materials of considerable complexity.

In this work has been laid the foundations of an automatic procedure which, when completed, will provide a comprehensive classical parameterization of the system, with an accuracy comparable to that of the initial *ab-initio* data. With this parametrization it will be then possible to produce high quality data to be used in coarse-grained simulations, thus permitting to perform reliable large scale simulations.

Appendix A

Further details on the Central Cell Model

A.1 The time step and the scaling parameter γ

Although in principle the CCM computational approach can be used in any diffusive lattice model (with possible extensions to atomistic simulations, as we suggest in Section 4.7), in this work we chosen to apply it to a specific mesoscopic model which evolves through iterations with the homogeneous time step τ. It is related to the diffusivity of a lone particle moving from cage to cage in a reference system (experimental or MD simulation) through

$$\tau = \frac{\lambda^2 \lim_{\langle n\rangle \to 0} D_s^{\mathrm{int}}}{\lim_{\langle n\rangle \to 0} \mathscr{D}_s}, \tag{A.1}$$

where D_s^{int} is the self-diffusion coefficient in internal lattice units (related to D_s through $D_s = \frac{\lambda^2}{\tau} D_s^{\mathrm{int}}$), and $\mathscr{D}_s$ is the reference self-diffusivity, in units of $\mathrm{m}^2\ \mathrm{s}^{-1}$.

τ cannot be assigned an arbitrarily large value, since the evolution algorithm is devised in such a way that, during one time step, a particle can migrate at most into one of the first neighbors of the leaving cell. Therefore, τ is constrained to be not greater than the mean time required to a (real) sorbate molecule to migrate from a cage to a neighboring one in an atomistic simulation.

When constructing a correspondence between this theoretical model and some molecular reference system, e.g., an MD simulation, a proper time interval should be found in the atomistic system which ensures that the list of possible movements of a lone particle in the lattice model (zero loading limit) covers the whole list of movements realized in the MD simulation within that time interval. Apart from the energy parameters, a crucial role in the work of refining the resemblance between the two systems (in terms of jump probabilities and the resulting memory effects) is played by the scaling parameter γ, introduced in Equation (4.8). γ affects the mean residence time in the lattice sites, and consequently the entity of correlations, since the smaller its value is, the longer it will take for a cell to modify its current configuration. The practical effect of lowering γ is to shorten the time step length τ. In a parametric study such as the present one, this causes no consistency problems. Instead, in the effort of making the lattice-gas simulation resemble an atomistic one, the lattice-gas dynamics cannot be arbitrarily slowed down nor accelerated without interfering with the list of possible site-to-site jumps. In other words, once the lattice-gas system have been set up, if one wishes to slow down or to speed up the lattice-gas dynamics, only relatively small variation of γ will be consistent with the same list of possible site-to-site jumps. Therefore, in realistic lattice-gas simulations γ is limited to the usage of a *tuner* of the memory effects and cannot be manipulated arbitrarily as a time step controller.

A.2 Green-Kubo formulation of the self-diffusivity in a lattice-gas with a homogeneous time step

Since the model evolves in discrete time step of equal length τ, a generic instant of time t can be read as $t = z\tau$, with $z \in \mathbb{N}$ as the number of time steps needed to let the system evolve of an amount t from the time origin. From Equation (4.18), the net displacement of the tagged particle from instant 0 to t is

$$\mathbf{r}(t) - \mathbf{r}(0) = \sum_{k=0}^{z-1} \delta\mathbf{r}(k\tau). \tag{A.2}$$

The mean-square displacement (MSD) after a time interval t, defined as $\langle \Delta\mathbf{r}^2(t) \rangle = \langle [\mathbf{r}(t) - \mathbf{r}(0)]^2 \rangle$, where d is the number of dimensions of the lattice, reads

$$\left\langle \Delta\mathbf{r}^2(t) \right\rangle = \left\langle \sum_{h=0}^{z-1} \sum_{k=0}^{z-1} \delta\mathbf{r}(h\tau) \cdot \delta\mathbf{r}(k\tau) \right\rangle. \tag{A.3}$$

Using the symmetry properties of time-correlation functions, we get

$$\begin{aligned} \left\langle \Delta\mathbf{r}^2(t) \right\rangle &= z \left\langle \delta\mathbf{r}(0) \cdot \delta\mathbf{r}(0) \right\rangle \\ &\quad + 2 \sum_{k=0}^{z-1} (z-k) \left\langle \delta\mathbf{r}(k\tau) \cdot \delta\mathbf{r}(0) \right\rangle. \end{aligned} \tag{A.4}$$

The same procedure can be used to get the MSD after a time $t+\tau$ which reads

$$\begin{aligned}\left\langle \Delta\mathbf{r}^2(t+\tau)\right\rangle = {} & (z+1)\left\langle \delta\mathbf{r}(0)\cdot\delta\mathbf{r}(0)\right\rangle \\ & + 2\sum_{k=0}^{z}(z-k+1)\left\langle \delta\mathbf{r}(k\tau)\cdot\delta\mathbf{r}(0)\right\rangle . \end{aligned} \tag{A.5}$$

For a continuous-time system where the MSD goes linearly with time in the long-time limit, the self-diffusion coefficient, D_s, can be retrieved through the time-derivative of the mean square displacement,

$$D_s = \frac{1}{2d}\lim_{t\to\infty}\frac{\mathrm{d}\left\langle \Delta\mathbf{r}^2(t)\right\rangle}{\mathrm{d}t}, \tag{A.6}$$

or, alternatively

$$D_s = \frac{1}{2d}\lim_{t\to\infty}\frac{\left\langle \Delta\mathbf{r}^2(t)\right\rangle}{t}. \tag{A.7}$$

Equation (A.6) can be reformulated for the discrete-time case as

$$D_s = \frac{1}{2d}\lim_{t\to\infty}\frac{\left\langle \Delta\mathbf{r}^2(t+\tau)\right\rangle - \left\langle \Delta\mathbf{r}^2(t)\right\rangle}{\tau}, \tag{A.8}$$

were we do *not* take the zero limit of τ, since it is a fixed parameter. Inserting Equations (A.4) and (A.5) into Equation (A.8) and taking account of the fact that the limit $t\to\infty$ corresponds to $z\to\infty$, we get Equation(4.19). The same result is obtained by inserting Equation (A.4) into Equation (A.7).

Appendix B

Initial cations configuration in Ca A

Reported here is the procedure adopted in our work to determine the initial configuration of the Ca cations within zeolite A. In this structure, experimental studies show that Ca cations prefer to occupy the type I sites [126]. There are 64 of these sites per unit cell, located at the vertexes of an ideal cube within each of the 8 β-cages. Since the Ca cations are only 48, one has to choose which are the empty sites. As a first approach, we have isolated the minimum

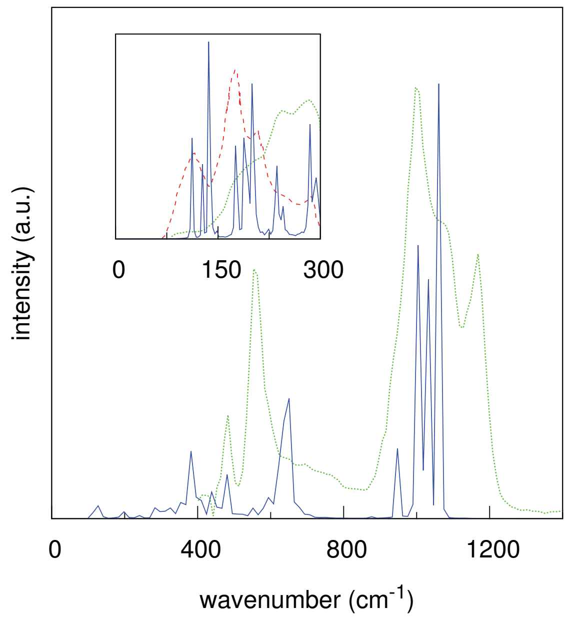


Figure B.1: IR spectra for Ca A, experimental (dots) [111] and minimum potential energy configuration model (continuous line); in the inset, zoom of far infrared region related to cations vibrations against experimental data (dots [142] and dashes [143]).

potential energy configuration of the whole framework. To do this, we have generated a large number of configurations, randomly distributing the 48 Ca cations over the 64 type I sites. Sorting all these configurations according to decreasing potential energy, we noticed that the minimum energy configuration is the one having 6 cations per β-cage, arranged in such a way that the vacancies lie along the main diagonal of each ideal cube and, moreover, all the diagonals are parallel-oriented with respect to each other. The IR spectrum, computed via an MD simulation starting from this minimum energy configuration, shows very narrow and isolated bands, see Figure B.1. This spiky shape deviates considerably from the experimental one and is probably due to the high degree of order in the cations distribution.

From this evidence, it seems reasonable to think that in real Ca A samples the cations are distributed in a more random way among the type I sites. This can be understood considering that the cations distribution is driven not only by the energetics but also by statistical factors: if there are many configurations which are energetically equivalent to each other and their energy is slightly higher than the minimum one, then the real crystal structure may be a combination of these configurations. Note that these considerations, far from being exhaustive, are aimed at giving us a plausible structure: several other factors play a key role in determining the real cationic distribution (e.g., synthesis procedures) but this topic goes beyond the scope of our work [197].

At this point, we considered a sample of 10^5 configurations, once again generated by randomly distributing the 48 Ca cations over the 64 type I sites, and plotted the probability density distribution of finding a configuration with potential energy in between the interval U and $U + dU$, see Figure B.2. Looking at the plot, the most probable energy for a configuration taken from a completely random sample is about $6500\,\text{kcal} \cdot \text{mol}^{-1}$ per unit cell. Choosing arbitrarily one configuration with this energy we compute the IR spectrum, see Figure B.3. Clearly, the shape resembles more closely the experimental one.

As a final refinement, we looked only for configurations having 6 cations per each β-cage. This requirement is less demanding than having the vacancies only on the main diagonals and the diagonals parallel to each other, but assures at the same time a good degree of symmetry with less distortion of the framework structure, thanks to more evenly distributed cations among the various cages. Generating a sample of 10^5 configurations, randomly distributing 6 cations in each β-cage, we recompute the probability distribution and plotted it, see Figure B.4. The most probable energy for a random configuration (given the constraint of 6 cations per β-cage) is about $6100\,\text{kcal} \cdot \text{mol}^{-1}$ per unit cell. From the population we take one configuration having this energy and compute the IR spectrum, see Figure B.5.

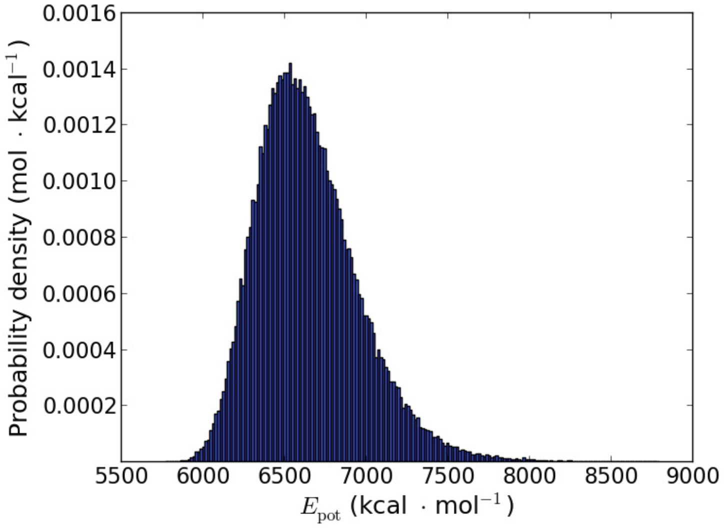


Figure B.2: Distribution of fully random cations configurations according to total system potential energy.

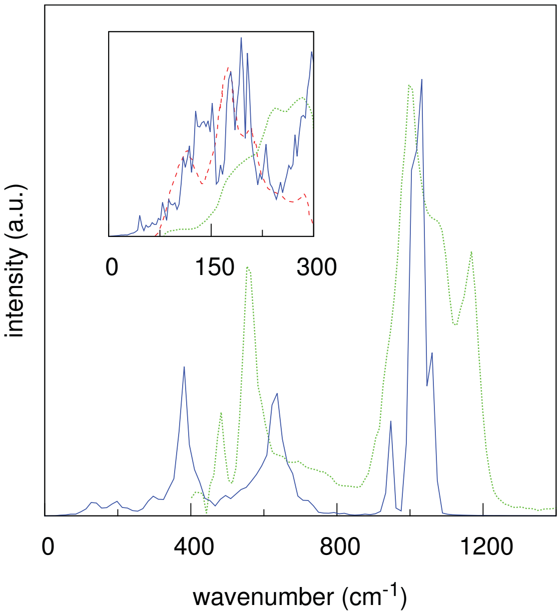


Figure B.3: IR spectra for Ca A, experimental (dots) [111] and fully random configuration model (continuous line); in the inset, zoom of far infrared region related to cations vibrations against experimental data (dots [142] and dashes [143]).

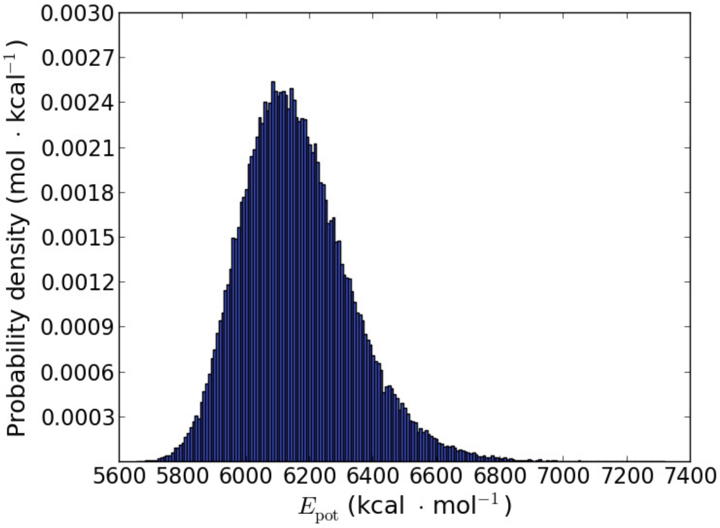


Figure B.4: Distribution of 6-per-cage constrained, see text, cations configurations according to total system potential energy.

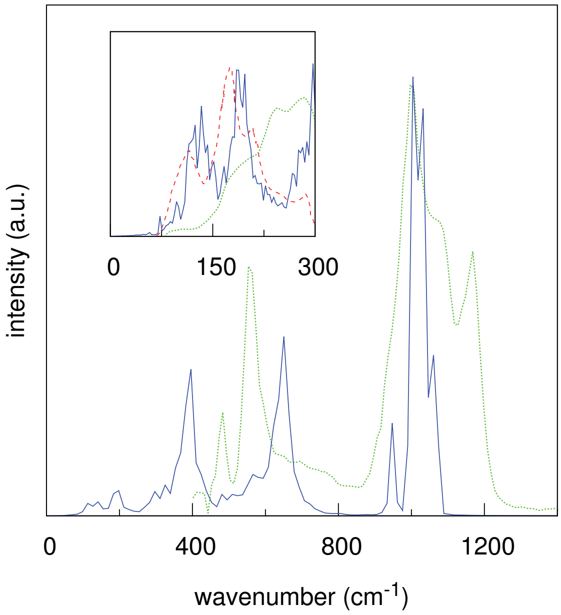


Figure B.5: IR spectra for Ca A, experimental (dots) [111] and 6-per-cage constrained configuration model (continuous line); in the inset, zoom of far infrared region related to cations vibrations against experimental data (dots [142] and dashes [143]).

As can be seen, the spectrum is very similar to the one computed in the fully random case, Figure B.3, but the framework structure is certainly more homogeneous and stable. This configuration is, from our point of view, the best choice and it is the one that has been used as the starting point for all the Ca A simulations presented in chapter 5.

Appendix C

Additional Force Matching data

Reported here:

- comparison between BOMD and experimental IR spectra (Figures C.1 to C.4);
- comparison of IR spectra obtained with a FF optimized only with bonded interactions and a FF optimized including also the nonbonded interactions, taken from the literature (Figures C.5 to C.8);
- comparison between IR spectra obtained from FFs optimized with and without Urey-Bradley term, for both CO_2 and ZIF-8 (Figures C.9 and C.10, respectively);
- dihedral angles distributions for C1-N-Zn-N and C2-N-Zn-N obtained from BOMD trajectory (Figures C.11 and C.12, respectively).

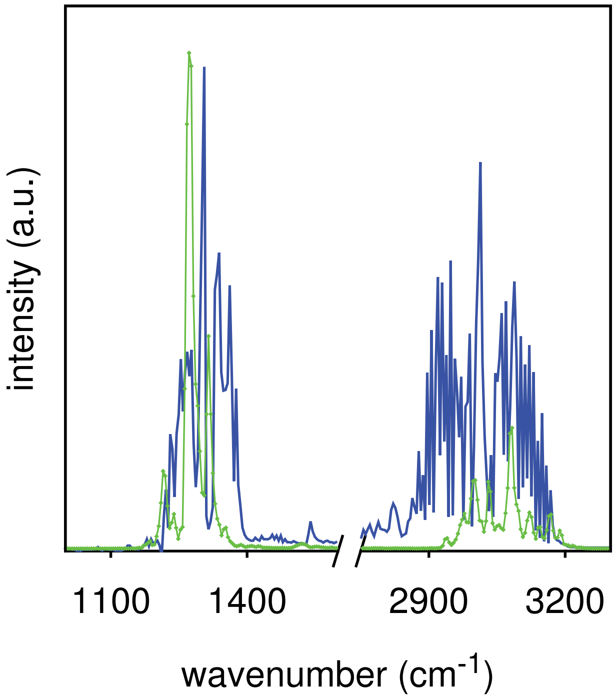


Figure C.1: Methane IR spectra, BOMD (dots) vs. experimental [198] (solid line).

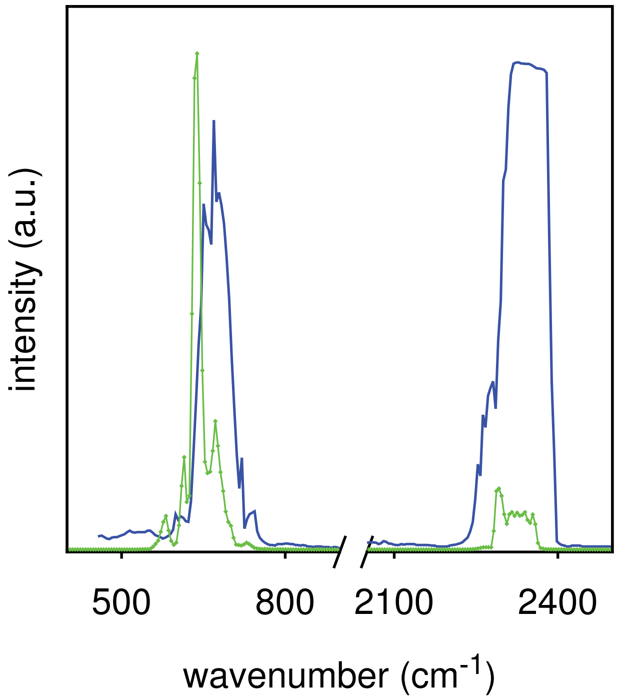


Figure C.2: CO_2 IR spectra, BOMD (dots) vs. experimental [198] (solid line).

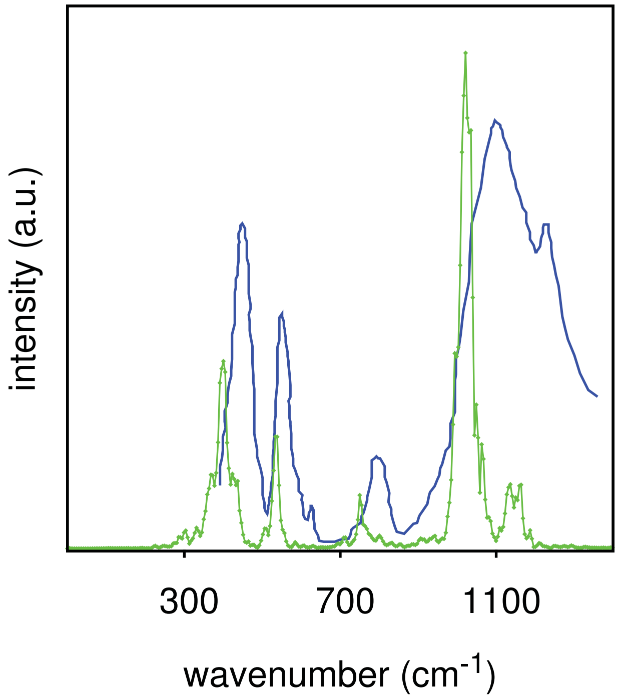


Figure C.3: Silicalite IR spectra, BOMD (dots) vs. experimental [111] (solid line).

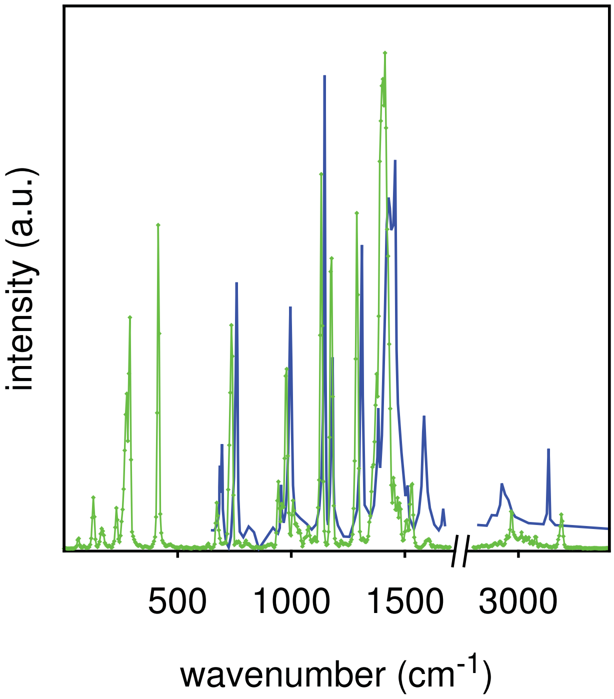


Figure C.4: ZIF-8 IR spectra, BOMD (dots) vs. experimental [193] (solid line).

In the following four figures, the FF optimization including also the nonbonded interactions has been performed holding fixed (i.e., cannot be changed by the matcher) the latter constants to the values taken from the literature. For each system, the nonbonded parameters are reported in the corresponding table.

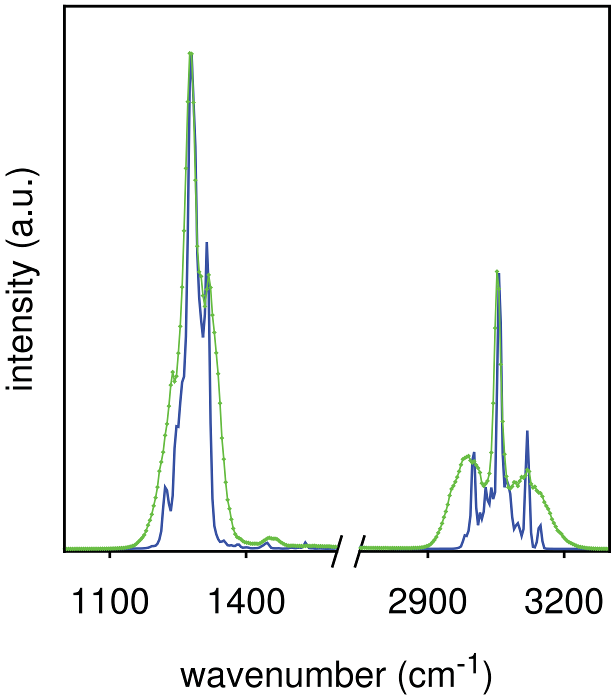


Figure C.5: Methane IR spectra obtained from a FF optimized only with bonded interactions (solid line) vs. a FF optimized including also the nonbonded interactions (dots).

Table C.1: Methane nonbonded parameters [183].

	$\epsilon(\mathrm{kcal \cdot mol^{-1}})$	$\sigma(\mathrm{Å})$	$R_{\mathrm{min}}(\mathrm{Å})$	$q(e)$
C	-0.051	3.344	3.754	-0.572
H	-0.055	2.641	2.964	0.143

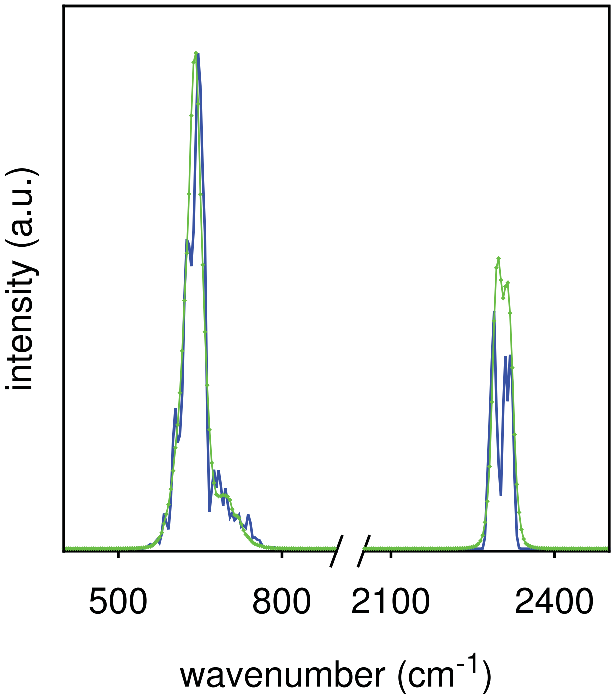


Figure C.6: CO_2 IR spectra obtained from a FF optimized only with bonded interactions (solid line) vs. a FF optimized including also the nonbonded interactions (dots).

Table C.2: Carbon dioxide nonbonded parameters [188].

	$\epsilon(\mathrm{kcal} \cdot \mathrm{mol}^{-1})$	$\sigma(\text{Å})$	$R_{\mathrm{min}}(\text{Å})$	$q(e)$
C	-0.0558	2.757	3.0946	0.6512
O	-0.1598	3.033	3.4044	-0.3256

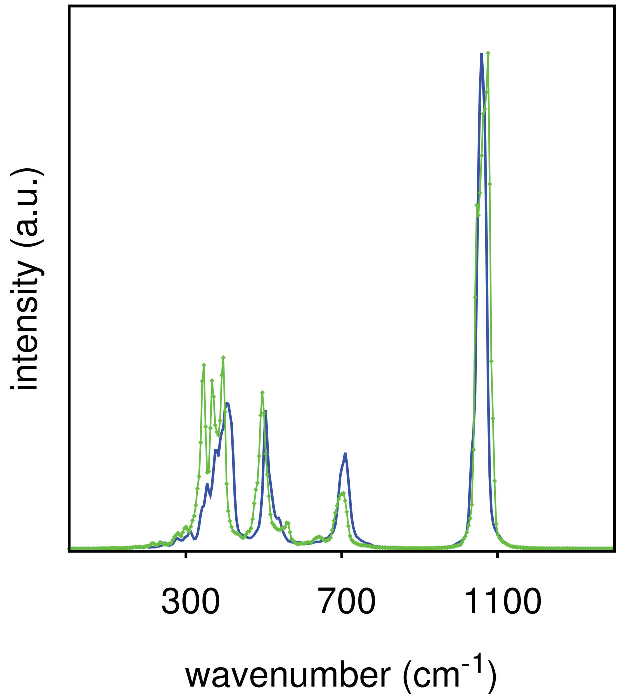


Figure C.7: Silicalite IR spectra obtained from a FF optimized only with bonded interactions (solid line) vs. a FF optimized including also the nonbonded interactions (dots).

Table C.3: Silicalite nonbonded parameters [128]. Exclusion policy $1-4$ rescaling factor is 1.0 for vdW and 0.5 for Coulomb.

	$\epsilon(\mathrm{kcal} \cdot \mathrm{mol}^{-1})$	$\sigma(\text{Å})$	$R_{\mathrm{min}}(\text{Å})$	$q(e)$
Si	-0.162	3.963	4.448	1.10
O	-0.058	3.063	3.438	-0.55

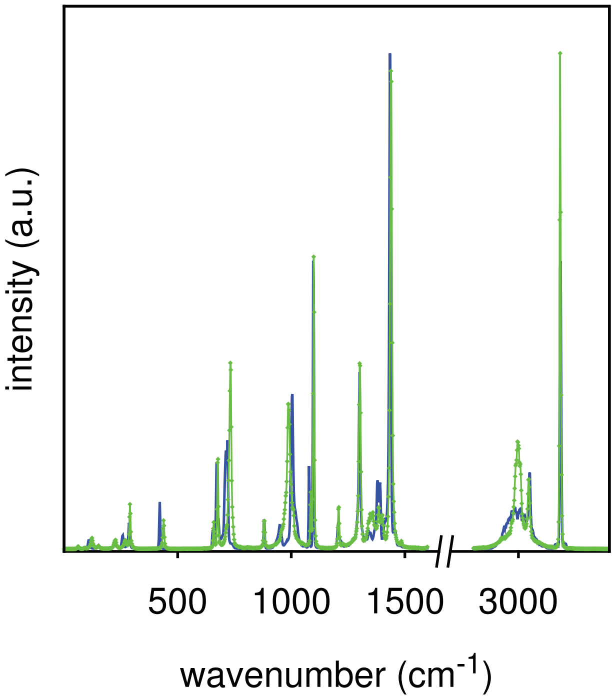


Figure C.8: ZIF-8 IR spectra obtained from a FF optimized only with bonded interactions (solid line) vs. a FF optimized including also the nonbonded interactions (dots).

Table C.4: ZIF-8 nonbonded parameters [147]. Exclusion policy $1-4$ rescaling factor is 0.5 for vdW and 0.833 for Coulomb. Note the presence of two typos in Ref. 147: the sigma of H3 and the partial charge of C3, the values here reported are the correct ones.

	$\epsilon(\mathrm{kcal}\cdot\mathrm{mol}^{-1})$	$\sigma(\text{Å})$	$R_{\mathrm{min}}(\text{Å})$	$q(e)$
C1	-0.0860	3.400	3.816	0.4339
C2	-0.0860	3.400	3.816	-0.1924
C3	-0.1094	3.400	3.816	-0.6042
H2	-0.0150	2.511	2.818	0.1585
H3	-0.0157	2.471	2.774	0.1572
N	-0.1700	3.250	3.648	-0.3008
Zn	-0.0125	1.960	2.200	0.7362

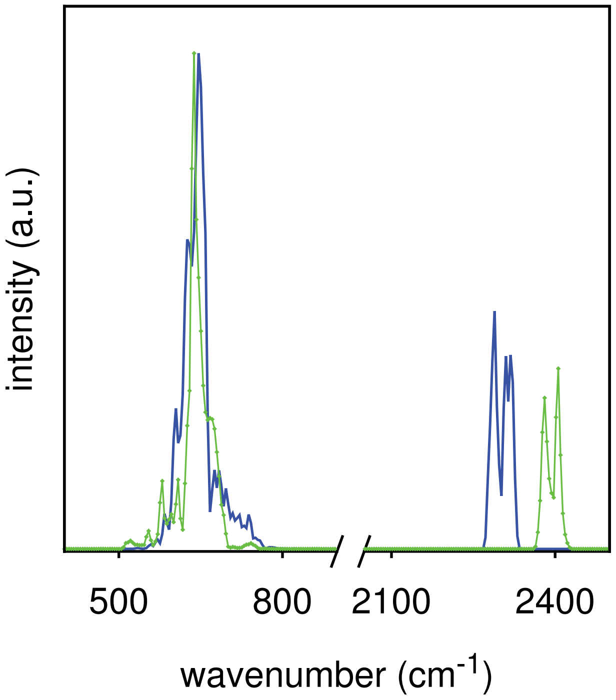


Figure C.9: CO_2 IR spectra obtained from a FF optimized including the UB term (solid line) vs. a FF optimized excluding it (dots).

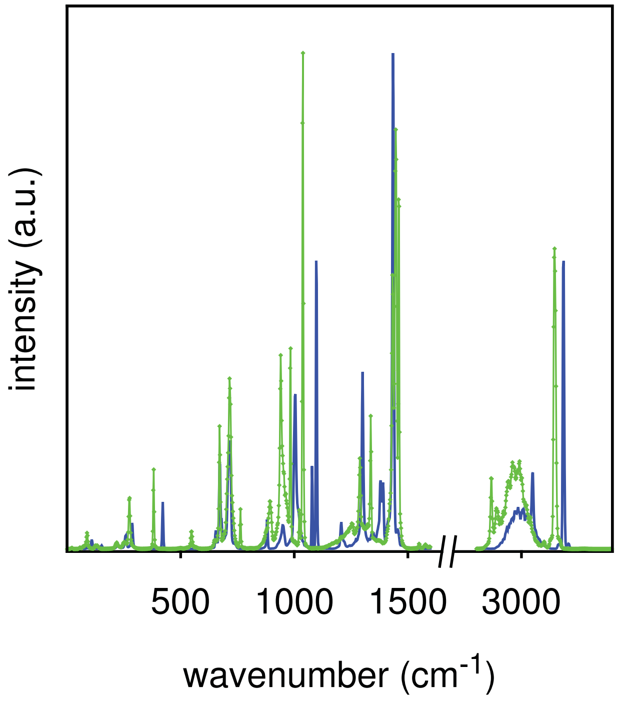


Figure C.10: ZIF-8 IR spectra obtained from a FF optimized including the UB term (solid line) vs. a FF optimized excluding it (dots).

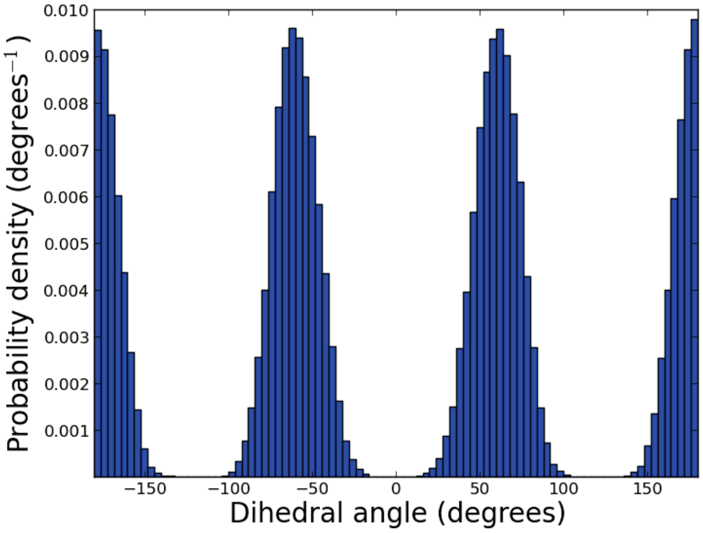


Figure C.11: C1-N-Zn-N dihedral angles distribution (phase $\delta = 0\,^{\circ}$).

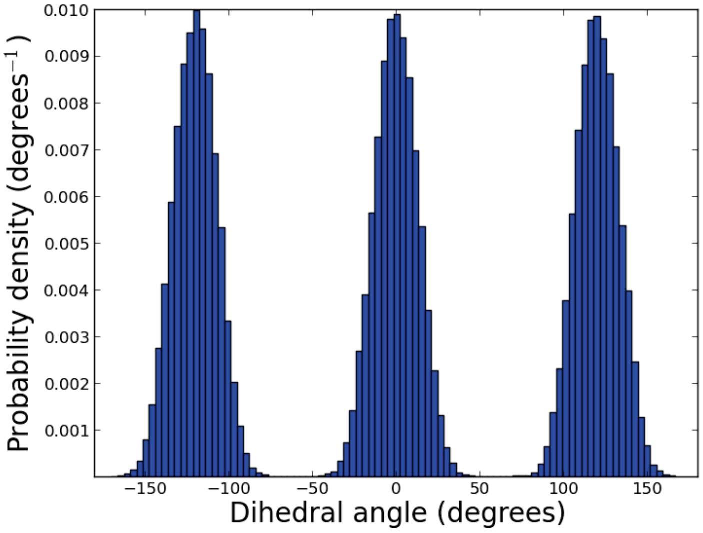


Figure C.12: C2-N-Zn-N dihedral angles distribution (phase $\delta = 180\,^{\circ}$).

Bibliography

[1] P. Atkins and R. Friedman, *Molecular Quantum Mechanics 4/E*, Oxford University Press, 2005.

[2] C. Cramer, *Essentials of Computational Chemistry: Theories and Models*, Wiley, 2002.

[3] L. H. Thomas, *Mathematical Proceedings of the Cambridge Philosophical Society* **23**, 542 (1927).

[4] E. Fermi, *Zeitschrift für Physik* **48**, 73 (1928).

[5] F. Bloch, *Zeitschrift für Physik* **57**, 545 (1929).

[6] P. A. M. Dirac, *Mathematical Proceedings of the Cambridge Philosophical Society* **26**, 376 (1930).

[7] J. C. Slater, *Physical Review* **81**, 385 (1951).

[8] P. Hohenberg and W. Kohn, *Physical Review* **136**, B864 (1964).

[9] W. Kohn and L. J. Sham, *Physical Review* **140**, A1133 (1965).

[10] R. O. Jones, Introduction to Density Functional Theory and Exchange-Correlation Energy Functionals, in *Computational Nanoscience: Do It Yourself!*, edited by J. Grotendorst, S. Blügel, and D. Marx, pp. 45–70, John von Neumann Institute for Computing, Jülich, 2006.

[11] K. Burke, *The Journal of Chemical Physics* **136**, 150901 (2012).

[12] K. Burke, J. P. Perdew, and M. Ernzerhof, *International Journal of Quantum Chemistry* **61**, 287 (1997).

[13] J. P. Perdew, K. Burke, and M. Ernzerhof, *Physical Review Letters* **77**, 3865 (1996).

[14] A. D. Becke, *Physical Review A* **38**, 3098 (1988).

[15] C. Lee, W. Yang, and R. G. Parr, *Physical Review B* **37**, 785 (1988).

[16] A. D. Becke, *The Journal of Chemical Physics* **98**, 5648 (1993).

[17] M. Guidon, F. Schiffmann, J. Hutter, and J. VandeVondele, *The Journal of Chemical Physics* **128**, (2008).

[18] J. Klimeš and A. Michaelides, *The Journal of Chemical Physics* **137**, (2012).

[19] B. J. Alder and T. E. Wainwright, *The Journal of Chemical Physics* **27**, 1208 (1957).

[20] B. J. Alder and T. E. Wainwright, *The Journal of Chemical Physics* **31**, 459 (1959).

[21] M. P. Allen and D. J. Tildesley, *Computer simulation of liquids*, Clarendon Press, New York, NY, USA, 1989.

[22] D. Frenkel and B. Smit, *Understanding Molecular Simulation*, Academic Press, Inc., Orlando, FL, USA, 2nd edition, 2001.

[23] G. Sutmann, Molecular Dynamics - Vision and Reality, in *Computational Nanoscience: Do It Yourself!*, edited by J. Grotendorst, S. Blügel, and D. Marx, pp. 159–194, John von Neumann Institute for Computing, Jülich, 2006.

[24] G. Sutmann, Molecular Dynamics - Extending the Scale from Microscopic to Mesoscopic, in *Multiscale Simulation Methods in Molecular Sciences*, edited by J. Grotendorst, N. Attig, S. Blügel, and D. Marx, pp. 1–49, John von Neumann Institute for Computing, Jülich, 2009.

[25] D. A. McQuarrie, *Statistical Mechanics*, Harper and Row, New York, first edition, 1976.

[26] D. Chandler, *Introduction to modern statistical mechanics*, Oxford university press, New York, Oxford, 1987.

[27] T. Hill, *An Introduction to Statistical Thermodynamics*, Addison-Wesley series in chemistry, Dover Publications, 1960.

[28] D. Dubbeldam, E. Beerdsen, T. J. H. Vlugt, and B. Smit, *The Journal of Chemical Physics* **122**, 224712 (2005).

[29] H. Goldstein, C. P. Poole, and J. L. Safko, *Classical Mechanics (3rd Edition)*, Addison-Wesley, 3 edition, 2001.

[30] M. E. Tuckerman and G. J. Martyna, *The Journal of Physical Chemistry B* **104**, 159 (2000).

[31] L. Verlet, *Physical Review* **159**, 98 (1967).

[32] D. Marx, An Introduction to Ab Initio Molecular Dynamics Simulations, in *Computational Nanoscience: Do It Yourself!*, edited by J. Grotendorst, S. Blügel, and D. Marx, pp. 195–244, John von Neumann Institute for Computing, Jülich, 2006.

[33] J. VandeVondele, M. Krack, F. Mohamed, M. Parrinello, T. Chassaing, and J. Hutter, *Computer Physics Communications* **167**, 103 (2005).

[34] T. Warnock, *Los Alamos Science* **15**, 137 (1987).

[35] R. Eckhardt, *Los Alamos Science* **15**, 131 (1987).

[36] N. Metropolis, *Los Alamos Science* **15**, 125 (1987).

[37] B. J. Alder, S. P. Frankel, and V. A. Lewinson, *The Journal of Chemical Physics* **23**, 417 (1955).

[38] K. Binder, *Reports on Progress in Physics* **60**, 487 (1997).

[39] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, *The Journal of Chemical Physics* **21**, 1087 (1953).

[40] I. Beichl and F. Sullivan, *Computing in Science & Engineering* **2**, 65 (2000).

[41] J. Kemeny and J. Snell, *Finite Markov Chains: With a New Appendix "Generalization of a Fundamental Matrix"*, Undergraduate Texts in Mathematics, Springer, 1976.

[42] N. V. Kampen, *Stochastic processes in physics and chemistry*, North Holland, 2007.

[43] P. Kratzer, Monte Carlo and Kinetic Monte Carlo Methods - A Tutorial, in *Multiscale Simulation Methods in Molecular Sciences*, edited by J. Grotendorst, N. Attig, S. Blügel, and D. Marx, pp. 51–76, John von Neumann Institute for Computing, Jülich, 2009.

[44] V. I. Manousiouthakis and M. W. Deem, *The Journal of Chemical Physics* **110**, 2753 (1999).

[45] A. Voter, Introduction to the Kinetic Monte Carlo Method, in *Radiation Effects in Solids*, edited by K. Sickafus, E. Kotomin, and B. Uberuaga, volume 235 of *NATO Science Series*, pp. 1–23, Springer Netherlands, 2007.

[46] K. A. Fichthorn and W. H. Weinberg, *The Journal of Chemical Physics* **95**, 1090 (1991).

[47] A. Chatterjee and D. G. Vlachos, *Journal of Computer-Aided Materials Design* **14**, 253 (2007).

[48] N. Margolus, T. Toffoli, and G. Vichniac, *Physical Review Letters* **56**, 1694 (1986).

[49] T. Toffoli, Occam, Turing, von Neumann, Jaynes: How much can you get for how little? (A conceptual introduction to cellular automata), 1994.

[50] T. Toffoli and N. H. Margolus, *Physica D: Nonlinear Phenomena* **45**, 229 (1990).

[51] B. Chopard and M. Droz, *Cellular Automata Modeling of Physical Systems*, Cambridge University Press, Cambridge, England, first edition, 1998.

[52] M. Gardner, *Scientific American* , 120 (1970).

[53] S. Wolfram, *A New Kind of Science*, Wolfram Media, 2002.

[54] T. Toffoli and N. Margolus, *Cellular Automata Machines: A New Environment for Modeling*, MIT Press series in scientific computation, Cambridge, 1987.

[55] P. Demontis, F. G. Pazzona, and G. B. Suffritti, *The Journal of Physical Chemistry B* **110**, 13554 (2006).

[56] P. Demontis, F. G. Pazzona, and G. B. Suffritti, *The Journal of Chemical Physics* **126**, 194709 (2007).

[57] P. Demontis, F. G. Pazzona, and G. B. Suffritti, *The Journal of Chemical Physics* **126**, 194710 (2007).

[58] F. G. Pazzona, P. Demontis, and G. B. Suffritti, *The Journal of Chemical Physics* **131**, 234703 (2009).

[59] P. Demontis, F. G. Pazzona, and G. B. Suffritti, *The Journal of Physical Chemistry B* **112**, 12444 (2008).

[60] F. G. Pazzona, P. Demontis, and G. B. Suffritti, *The Journal of Chemical Physics* **131**, 234704 (2009).

[61] F. G. Pazzona, *A cellular automata model for diffusion and adsorption in zeolites: Construction of a mesoscopic model*, LAP Lambert Academic Publishing, 66123 Saarbrücken, Germany, 2010.

[62] J. Kärger and D. M. Ruthven, *Diffusion in zeolites and other microporous solids*, John Wiley and Sons, New York, 1992.

[63] D. M. Ruthven, *Principles of Adsorption and Adsorption Processes*, John Wiley & Sons, 1984.

[64] G. Sastre and A. Corma, *Journal of Molecular Catalysis A: Chemical* **305**, 3 (2009).

[65] A. Huwe, F. Kremer, P. Behrens, and W. Schwieger, *Physical Review Letters* **82**, 2338 (1999).

[66] K. S. Park, Z. Ni, A. P. Côté, J. Y. Choi, R. Huang, F. J. Uribe-Romo, H. K. Chae, M. O'Keeffe, and O. M. Yaghi, *Proceedings of the National Academy of Sciences* **103**, 10186 (2006).

[67] A. Phan, C. J. Doonan, F. J. Uribe-Romo, C. B. Knobler, M. O'Keeffe, and O. M. Yaghi, *Accounts of Chemical Research* **43**, 58 (2010).

[68] B. Smit and T. L. M. Maesen, *Chemical Reviews* **108**, 4125 (2008).

[69] S. D. Collins, A. Chatterjee, and D. G. Vlachos, *The Journal of Chemical Physics* **129**, 184101 (2008).

[70] C. Colella, P. Aprea, B. de Gennaro, and B. Liguori, editors, *Proceedings of the 16th International Zeolite Conference*, A. De Frede, Naples, Italy, 2010.

[71] A. Bortz, M. Kalos, and J. Lebowitz, *Journal of Computational Physics* **17**, 10 (1975).

[72] B. D. Lubachevsky, *Journal of Computational Physics* **75**, 103 (1988).

[73] Y. Shim and J. G. Amar, *Physical Review B* **71**, 115436 (2005).

[74] E. Martínez, J. Marian, M. Kalos, and J. Perlado, *Journal of Computational Physics* **227**, 3804 (2008).

[75] Y. Shim and J. G. Amar, *Physical Review B* **71**, 125432 (2005).

[76] B. D. LUBACHEVSKY and A. WEISS, *CoRR* **cs.DC/0405053** (2004).

[77] M. MERRICK and K. A. FICHTHORN, *Physical Review E* **75**, 011606 (2007).

[78] S. NAMUANGRUK, P. PANTU, and J. LIMTRAKUL, *Journal of Catalysis* **225**, 523 (2004).

[79] P. DEMONTIS, S. YASHONATH, and M. L. KLEIN, *The Journal of Physical Chemistry* **93**, 5016 (1989).

[80] A. N. FITCH, H. JOBIC, and A. RENOUPREZ, *The Journal of Physical Chemistry* **90**, 1311 (1986).

[81] S. M. AUERBACH, N. J. HENSON, A. K. CHEETHAM, and H. I. METIU, *The Journal of Physical Chemistry* **99**, 10600 (1995).

[82] S. M. AUERBACH, L. M. BULL, N. J. HENSON, H. I. METIU, and A. K. CHEETHAM, *The Journal of Physical Chemistry* **100**, 5923 (1996).

[83] S. M. AUERBACH and H. I. METIU, *The Journal of Chemical Physics* **105**, 3753 (1996).

[84] C. SARAVANAN and S. M. AUERBACH, *The Journal of Chemical Physics* **107**, 8120 (1997).

[85] C. SARAVANAN and S. M. AUERBACH, *The Journal of Chemical Physics* **107**, 8132 (1997).

[86] C. SARAVANAN and S. M. AUERBACH, *The Journal of Chemical Physics* **110**, 11000 (1999).

[87] G. VITALE, C. F. MELLOT, L. M. BULL, and A. K. CHEETHAM, *The Journal of Physical Chemistry B* **101**, 4559 (1997).

[88] S. M. AUERBACH, *International Reviews in Physical Chemistry* **19**, 155 (2000).

[89] E. S. HOOD, B. H. TOBY, and W. H. WEINBERG, *Physical Review Letters* **55**, 2437 (1985).

[90] A. GERMANUS, J. KÄRGER, and H. PFEIFER, *Zeolites* **4**, 188 (1984).

[91] T. P. SCHULZE, *Physical Review E* **65**, 036704 (2002).

[92] C. BAERLOCHER, D. OLSON, and W. MEIER, *Atlas of Zeolite Framework Types (formerly: Atlas of Zeolite Structure Types): Atlas of Zeolite Structure Types)*, Elsevier Science, 2001.

[93] J. Klafter and J. M. Drake, editors, *Molecular Dynamics in Restricted Geometries*, John Wiley and Sons, New York, first edition, 1989.

[94] M.-O. Coppens, A. T. Bell, and A. K. Chakraborty, *Chemical Engineering Science* **53**, 2053 (1998).

[95] C. Saravanan, F. Jousse, and S. M. Auerbach, *Physical Review Letters* **80**, 5754 (1998).

[96] Z. Chvoj, H. Conrad, V. Cháb, M. Ondrejcek, and A. M. Bradshaw, *Surface Science* **329**, 121 (1995).

[97] J. van den Bergh, S. Ban, T. J. H. Vlugt, and F. Kapteijn, *The Journal of Physical Chemistry C* **113**, 17840 (2009).

[98] D. Frenkel and M. H. Ernst, *Physical Review Letters* **63**, 2165 (1989).

[99] D. F. M. A. van der Hoef, *Physical Review A* **41**, 4277 (1990).

[100] D. F. M. A. van der Hoef, *Physica D* **47**, 191 (1991).

[101] R. Gomer, *Reports on Progress in Physics* **53**, 917 (1990).

[102] P. Demontis, L. Fenu, and G. B. Suffritti, *The Journal of Physical Chemistry B* **109**, 18081 (2005).

[103] J. C. Phillips, R. Braun, W. Wang, J. Gumbart, E. Tajkhorshid, E. Villa, C. Chipot, R. D. Skeel, L. Kalé, and K. Schulten, *Journal of Computational Chemistry* **26**, 1781 (2005).

[104] B. R. Brooks, C. L. Brooks, A. D. MacKerell, L. Nilsson, R. J. Petrella, B. Roux, Y. Won, G. Archontis, C. Bartels, S. Boresch, A. Caflisch, L. Caves, Q. Cui, A. R. Dinner, M. Feig, S. Fischer, J. Gao, M. Hodoscek, W. Im, K. Kuczera, T. Lazaridis, J. Ma, V. Ovchinnikov, E. Paci, R. W. Pastor, C. B. Post, J. Z. Pu, M. Schaefer, B. Tidor, R. M. Venable, H. L. Woodcock, X. Wu, W. Yang, D. M. York, and M. Karplus, *Journal of Computational Chemistry* **30**, 1545 (2009).

[105] S. Plimpton, *Journal of Computational Physics* **117**, 1 (1995), `http://lammps.sandia.gov` (accessed November 2013).

[106] Y. Duan, C. Wu, S. Chowdhury, M. C. Lee, G. Xiong, W. Zhang, R. Yang, P. Cieplak, R. Luo, T. Lee, J. Caldwell, J. Wang, and P. Kollman, *Journal of Computational Chemistry* **24**, 1999 (2003).

[107] B. Hess, C. Kutzner, D. van der Spoel, and E. Lindahl, *Journal of Chemical Theory and Computation* **4**, 435 (2008).

[108] J. A. Anderson, C. D. Lorenz, and A. Travesset, *Journal of Computational Physics* **227**, 5342 (2008), `http://codeblue.umich.edu/hoomd-blue` (accessed November 2013).

[109] A. D. MacKerell, B. Brooks, C. L. Brooks, L. Nilsson, B. Roux, Y. Won, and M. Karplus, CHARMM: The Energy Function and Its Parameterization, in *Encyclopedia of Computational Chemistry*, chapter 1, pp. 271–277, John Wiley & Sons, 2002.

[110] P. E. M. Lopes, V. Murashov, M. Tazi, E. Demchuk, and A. D. MacKerell, *The Journal of Physical Chemistry B* **110**, 2782 (2006).

[111] P. Demontis, G. B. Suffritti, S. Bordiga, and R. Buzzoni, *Journal of the Chemical Society, Faraday Transactions* **91**, 525 (1995).

[112] P. Demontis, G. Stara, and G. B. Suffritti, *The Journal of Physical Chemistry B* **107**, 4426 (2003).

[113] P. Demontis, J. Gulín-González, H. Jobic, M. Masia, R. Sale, and G. B. Suffritti, *ACS Nano* **2**, 1603 (2008).

[114] P. Demontis, J. Gulín-González, H. Jobic, and G. B. Suffritti, *The Journal of Physical Chemistry C* **114**, 18612 (2010).

[115] P. Demontis, H. Jobic, M. A. Gonzalez, and G. B. Suffritti, *The Journal of Physical Chemistry C* **113**, 12373 (2009).

[116] P. Demontis, J. Gulín-González, M. Masia, and G. B. Suffritti, *Journal of Physics: Condensed Matter* **22**, 284106 (2010).

[117] P. Demontis, J. Gulín-González, and G. B. Suffritti, *The Journal of Physical Chemistry B* **110**, 7513 (2006).

[118] D. W. Breck, *Zeolite molecular sieves: structure, chemistry, and use*, John Wiley & Sons, New York, 1973.

[119] H. v. Bekkum, *Introduction to Zeolite Science and Practice*, Elsevier, 2001.

[120] W. Humphrey, A. Dalke, and K. Schulten, *Journal of Molecular Graphics* **14**, 33 (1996).

[121] P. Demontis and G. B. Suffritti, *Chemical Reviews* **97**, 2845 (1997).

[122] H. van Koningsveld, J. C. Jansen, and H. van Bekkum, *Zeolites* **10**, 235 (1990).

[123] H. van Koningsveld, *Acta Crystallographica Section B* **46**, 731 (1990).

[124] J. J. Pluth and J. V. Smith, *Journal of the American Chemical Society* **102**, 4704 (1980).

[125] R. L. Firor and K. Seff, *Journal of the American Chemical Society* **100**, 3091 (1978).

[126] F. Porcher, M. Souhassou, H. Graafsma, A. Puig-Molina, Y. Dusausoy, and C. Lecomte, *Acta Crystallographica Section B* **56**, 766 (2000).

[127] J. Hunger, I. A. Beta, H. Böhlig, C. Ling, H. Jobic, and B. Hunger, *The Journal of Physical Chemistry B* **110**, 342 (2006).

[128] J. B. Nicholas, A. J. Hopfinger, F. R. Trouw, and L. E. Iton, *Journal of the American Chemical Society* **113**, 4792 (1991).

[129] N. E. Ghermani, C. Lecomte, and Y. Dusausoy, *Physical Review B* **53**, 5231 (1996).

[130] E. Pantatosaki and G. K. Papadopoulos, *The Journal of Chemical Physics* **127**, 164723 (2007).

[131] `http://mackerell.umaryland.edu/CHARMM_ff_params.html` (accessed November 2013).

[132] D. Beglov and B. Roux, *The Journal of Chemical Physics* **100**, 9050 (1994).

[133] S. Marchand and B. Roux, *Proteins: Structure, Function, and Bioinformatics* **33**, 265 (1998).

[134] M. Fathizadeh and N. Ordou, *International Journal of Industrial Chemistry* **2**, 190 (2011).

[135] Dalai, Ajay K, Pradhan, Narayan C, Rao, Musti S, and Gokhale, K V G K, *Indian Journal of Engineering and Materials Sciences* **12**, 227 (2005).

[136] H. Karge and E. Geidel, Vibrational Spectroscopy, in *Characterization I*, edited by H. Karge and J. Weitkamp, volume 4 of *Molecular Sieves - Science and Technology*, pp. 1–200, Springer Berlin Heidelberg, 2004.

[137] P. H. Berens and K. R. Wilson, *The Journal of Chemical Physics* **74**, 4872 (1981).

[138] P. H. Berens, S. R. White, and K. R. Wilson, *The Journal of Chemical Physics* **75**, 515 (1981).

[139] S. H. Garofalini, *The Journal of Chemical Physics* **76**, 3189 (1982).

[140] F. Harris, *Proceedings of the IEEE* **66**, 51 (1978).

[141] P. Welch, *IEEE Transactions on Audio and Electroacoustics* **15**, 70 (1967).

[142] A. Boumiz, J. Cartigny, and E. Cohen de Lara, *The Journal of Physical Chemistry* **96**, 5419 (1992).

[143] M. D. Baker, J. Godber, and G. A. Ozin, FT–Far IR Spectroscopic Studies of Alkali and Alkaline Earth Linde Type A Zeolites, in *Perspectives in Molecular Sieve Science*, edited by W. H. Flank and T. E. Whyte, chapter 9, pp. 136–149, American Chemical Society, Washington, DC, 1988.

[144] M. Baertsch, P. Bornhauser, G. Calzaferri, and R. Imhof, *The Journal of Physical Chemistry* **98**, 2817 (1994).

[145] E. Geidel, K. Knut, H. Förster, and F. Bauer, *Journal of the Chemical Society, Faraday Transactions* **93**, 1439 (1997).

[146] P. Demontis and G. B. Suffritti, *Microporous and Mesoporous Materials* **125**, 160 (2009).

[147] B. Zheng, M. Sant, P. Demontis, and G. B. Suffritti, *The Journal of Physical Chemistry C* **116**, 933 (2012).

[148] A. Colantuono, S. Dal Vecchio, G. Mascolo, and M. Pansini, *Thermochimica acta* **296**, 59 (1997).

[149] G. Cruciani, *Journal of Physics and Chemistry of Solids* **67**, 1973 (2006).

[150] M. Noack, M. Schneider, A. Dittmar, G. Georgi, and J. Caro, *Microporous and Mesoporous Materials* **117**, 10 (2009).

[151] M. P. Attfield, *Chemical Communications* , 601 (1998).

[152] J. Gulín-González and G. Suffritti, *Microporous and Mesoporous Materials* **69**, 127 (2004).

[153] Y. Huang and E. A. Havenga, *Chemical Physics Letters* **345**, 65 (2001).

[154] F. Ercolessi and J. B. Adams, *EPL (Europhysics Letters)* **26**, 583 (1994).

[155] P. Maurer, A. Laio, H. W. Hugosson, M. C. Colombo, and U. Rothlisberger, *Journal of Chemical Theory and Computation* **3**, 628 (2007).

[156] P. Tangney and S. Scandolo, *The Journal of Chemical Physics* **117**, 8898 (2002).

[157] S. Izvekov, M. Parrinello, C. J. Burnham, and G. A. Voth, *The Journal of Chemical Physics* **120**, 10896 (2004).

[158] J. Sala, E. Guàrdia, J. Martí, D. Spångberg, and M. Masia, *The Journal of Chemical Physics* **136**, 054103 (2012).

[159] D. Spångberg, E. Guàrdia, and M. Masia, *Computational and Theoretical Chemistry* **982**, 58 (2012).

[160] J. Sala, E. Guàrdia, and M. Masia, *Computer Physics Communications* **182**, 1954 (2011).

[161] T. G. A. Youngs, M. G. Del Pópolo, and J. Kohanoff, *The Journal of Physical Chemistry B* **110**, 5697 (2006).

[162] Y. Umeno, T. Kitamura, K. Date, M. Hayashi, and T. Iwasaki, *Computational Materials Science* **25**, 447 (2002).

[163] T. J. Lenosky, J. D. Kress, I. Kwon, A. F. Voter, B. Edwards, D. F. Richards, S. Yang, and J. B. Adams, *Physical Review B* **55**, 1528 (1997).

[164] A. Gabrieli, M. Sant, P. Demontis, and G. B. Suffritti, *The Journal of Physical Chemistry C* **117**, 503 (2013).

[165] M. Frigo and S. Johnson, *Proceedings of the IEEE* **93**, 216 (2005).

[166] J. Kolafa, *Journal of Computational Chemistry* **25**, 335 (2004).

[167] J. VandeVondele and J. Hutter, *The Journal of Chemical Physics* **118**, 4365 (2003).

[168] G. Lippert, M. Parrinello, and J. Hutter, *Molecular Physics* **92**, 477 (1997).

[169] G. Bussi, D. Donadio, and M. Parrinello, *The Journal of Chemical Physics* **126**, 014101 (2007).

[170] M. Krack, *Theoretical Chemistry Accounts* **114**, 145 (2005).

[171] C. Hartwigsen, S. Goedecker, and J. Hutter, *Physical Review B* **58**, 3641 (1998).

[172] S. Goedecker, M. Teter, and J. Hutter, *Physical Review B* **54**, 1703 (1996).

[173] M. Kosa, J.-C. Tan, C. A. Merrill, M. Krack, A. K. Cheetham, and M. Parrinello, *ChemPhysChem* **11**, 2332 (2010).

[174] M. K. Rana, F. G. Pazzona, G. B. Suffritti, P. Demontis, and M. Masia, *Journal of Chemical Theory and Computation* **7**, 1575 (2011).

[175] S. Grimme, J. Antony, S. Ehrlich, and H. Krieg, *The Journal of Chemical Physics* **132**, 154104 (2010).

[176] G. van Rossum, Python tutorial, Report CS-R9526, Centrum voor Wiskunde en Informatica, P. O. Box 4079, 1009 AB Amsterdam, The Netherlands, 1995.

[177] The Python language reference, `http://docs.python.org/release/2.7` (accessed November 2013).

[178] R. Byrd, P. Lu, J. Nocedal, and C. Zhu, *SIAM Journal on Scientific Computing* **16**, 1190 (1995).

[179] C. Zhu, R. H. Byrd, P. Lu, and J. Nocedal, *ACM Transactions on Mathematical Software* **23**, 550 (1997).

[180] SciPy: Open source scientific tools for Python, v0.12, `http://www.scipy.org` (accessed November 2013).

[181] T. E. Oliphant, *Computing in Science & Engineering* **9**, 10 (2007).

[182] M. V. Berry, *Proceedings of the Royal Society of London A* **392**, 45 (1984).

[183] J. B. Nicholas, F. R. Trouw, J. E. Mertz, L. E. Iton, and A. J. Hopfinger, *The Journal of Physical Chemistry* **97**, 4149 (1993).

[184] P. Demontis, G. B. Suffritti, E. S. Fois, and S. Quartieri, *The Journal of Physical Chemistry* **96**, 1482 (1992).

[185] E. Pantatosaki, F. G. Pazzona, G. Megariotis, and G. K. Papadopoulos, *The Journal of Physical Chemistry B* **114**, 2493 (2010).

[186] L. N. Gergidis, D. N. Theodorou, and H. Jobic, *The Journal of Physical Chemistry B* **104**, 5541 (2000).

[187] A.-K. Pusch, T. Splith, L. Moschkowitz, S. Karmakar, R. Biniwale, M. Sant, G. Suffritti, P. Demontis, J. Cravillon, E. Pantatosaki, and F. Stallmach, *Adsorption* **18**, 359 (2012).

[188] J. G. Harris and K. H. Yung, *The Journal of Physical Chemistry* **99**, 12021 (1995).

[189] J.-R. Li, Y. Ma, M. C. McCarthy, J. Sculley, J. Yu, H.-K. Jeong, P. B. Balbuena, and H.-C. Zhou, *Coordination Chemistry Reviews* **255**, 1791 (2011).

[190] R. Banerjee, A. Phan, B. Wang, C. Knobler, H. Furukawa, M. O'Keeffe, and O. M. Yaghi, *Science* **319**, 939 (2008).

[191] C. Nieto-Draghi, T. de Bruin, J. Pérez-Pellitero, J. B. Avalos, and A. D. Mackie, *The Journal of Chemical Physics* **126**, 064509 (2007).

[192] T. Merker, C. Engin, J. Vrabec, and H. Hasse, *The Journal of Chemical Physics* **132**, 234512 (2010).

[193] Y. Hu, H. Kazemian, S. Rohani, Y. Huang, and Y. Song, *Chemical Communications* **47**, 12694 (2011).

[194] S. Moggach, T. Bennett, and A. Cheetham, *Angewandte Chemie International Edition* **48**, 7087 (2009).

[195] D. Fairen-Jimenez, S. A. Moggach, M. T. Wharmby, P. A. Wright, S. Parsons, and T. Düren, *Journal of the American Chemical Society* **133**, 8900 (2011).

[196] L. Zhang, Z. Hu, and J. Jiang, *Journal of the American Chemical Society* **135**, 3722 (2013).

[197] C. Beauvais, X. Guerrault, F.-X. Coudert, A. Boutin, and A. H. Fuchs, *The Journal of Physical Chemistry B* **108**, 399 (2004).

[198] A. Smith, The Coblentz Society Desk Book of Infrared Spectra, in *The Coblentz Society Desk Book of Infrared Spectra, Second Edition*, The Coblentz Society Kirkwood, MO, 1982.