# Dielectric Variations in Simulations of Charged Soft Matter

Von der Fakultät für Mathematik und Physik
der Universität Stuttgart zur Erlangung der
Würde eines Doktors der Naturwissenschaften
(Dr. rer. nat.) genehmigte Abhandlung

Vorgelegt von

Florian Fahrenberger
aus Erlangen

Hauptberichter: Prof. Dr. Christian Holm
Mitberichter: Priv.-Doz. Dr. Johannes Roth
Tag der mündlichen Prüfung: 08.01.2016

Institut für Computerphysik,
Universität Stuttgart
2016

# Contents

# Zusammenfassung

In den letzten Jahrzehnten haben sich Computersimulationen fest als Bestandteil der Physik etabliert. Sie stellen eine Brücke dar zwischen der Beobachtung von makroskopischen Observablen im Experiment und den detaillierten analytischen Berechnungen der Theorie, siehe Abbildung 0.1.

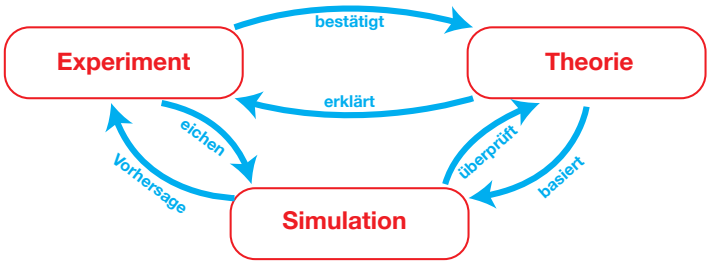



**Abbildung 0.1.:** Simulationen dienen als drittes Standbein der Physik. Sie basieren auf theoretischen Berechnungen und können diese gleichzeitig überprüfen. Experimentelle Messungen auf der anderen Seite können zur Eichung von Simulationen dienen und Simulationen wiederum zur Vorhersage neuer experimenteller Aufbauten.

Auf dem Gebiet der Simulation selbst gibt es mehrere Ansätze zur Darstellung physikalischer Systeme, die auf verschiedenen Größenskalen einsetzbar sind. Auf sehr kleinen Skalen, im Sub-Nanometerbereich, kann man mittels numerischer Störungsrechnung die quantenmechanischen Zustandsfunktionen eines Systems bestimmen. Das ist für die Dynamik weniger Atome oder Moleküle mit vertretbarem Rechenaufwand möglich und wird „Ab-Initio“-Simulation genannt. Am anderen

Ende der Skala, zur Simulation rein makroskopischer Phänomene, befinden sich Kontinuumsmethoden, beispielsweise die Familie der Finite-Differenzen-, Finite-Elemente- und Finite-Volumina-Methoden. Sie finden häufig Anwendung im Ingenieurwesen und zur Simulation von Flüssigkeitsdynamik oder starren Körpern.

In der Mitte dieser Extrema ist die Molekularsimulation angesiedelt. Im Gegensatz zu quantenmechanischen Ansätzen werden Atome und Moleküle hier nicht als Wellenfunktionen sondern als Partikel dargestellt, die über klassische Potentiale miteinander wechselwirken. Teilchensimulationen reichen bis hinunter in den atomistischen Bereich, wo Atomkerne und dazugehörig ein „Molekularfeld“, gemittelt aus der Elektronendichte, für die Dynamik einzelner Atome verwendet werden. Diese Methodik wird Dichtefunktionaltheorie genannt. Molekularsimulationen reichen aber auch bis in den mesoskopischen Skalenbereich, bis hin zu Mikrometern oder Millimetern. Hier werden dann oft mehrere Teilchen zu mesoskopischen Partikeln zusammengefasst, die entweder auf ein Gitter beschränkt – wie etwa bei Lattice-Boltzmann – oder mit adaptierten Wechselwirkungen die makroskopischen Gleichgewichtszustände asymptotisch reproduzieren. Anders als bei makroskopischen Ansätzen werden hierbei die thermischen Bewegungen der Teilchen noch berücksichtigt.

Die klassischen Wechselwirkungen zwischen diesen Teilchen werden entweder aus Ab-Initio-Simulationen oder aus experimentellen Aufbauten bestimmt. Die Dynamik der Teilchen kann man durch numerische Integration der Newtonschen Bewegungsgleichungen

$$m_i \frac{\partial^2}{\partial t^2} \mathbf{r}_i = -\nabla V(\mathbf{r}_1, \ldots, \mathbf{r}_n) \tag{0.1}$$

bestimmen, wobei hier $m_i$ die Massen und $\mathbf{r}_i$ die Positionen der Teilchen sind, sowie V das resultierende Potential aller Wechselwirkungen. Diese numerische Integration wird aufgrund der hohen Zahl an Freiheitsgraden oft explizit und nicht implizit vorgenommen. Hierbei werden aus den Potentialen die resultierenden Kräfte

$$\mathbf{F}_i = -\nabla V(\mathbf{r}_1, \ldots, \mathbf{r}_n)|_{\mathbf{r}_i} \tag{0.2}$$

berechnet. Daraus können dann die Teilchenpositionen um einen kleinen Zeitschritt $\delta t$ aktualisiert werden. Das hierbei in dieser Arbeit tatsächlich verwendete Integrationsschema, Velocity-Verlet, ist zweiter Ordnung in $\delta t$ und zeitreversibel. Letzteres ist wichtig, da hierbei das Phasenraumvolumen und damit die Gesamtenergie des Systems theoretisch erhalten bleibt.

Diese explizite Integration ist numerisch nicht besonders aufwendig, so dass die Berechnung der Kräfte auf jedes Teilchen mit großem Abstand den rechenintensivsten Teil der Simulation darstellt. Hierfür wird nochmals unterschieden zwischen kurzreichweitigen Wechselwirkungen, deren Potential schneller als $1/r^D$ abfällt, wobei $D$ die Dimension des Systems ist, und langreichweitigen Wechselwirkungen, die langsamer abfallen. Erstere sind numerisch leichter zu bewältigen, da man die Paarwechselwirkung mit Teilchen jenseits eines bestimmten Abstands, dem „cut-off"-Radius, vernachlässigen kann, weil sie nur noch Beiträge unterhalb der Maschinengenauigkeit des Computers liefern. Für langreichweitige Wechselwirkungen hingegen muss man alle Teilcheninteraktionen in die Kraftberechnung einbeziehen und noch dazu die periodisch replizierten Bilder des Systems in allen Dimensionen, die oft verwendet werden um Randeffekte an den Systemgrenzen zu vermeiden. Die so entstehende Summation ist nicht nur unendlich, sondern auch nur bedingt konvergent. Die beiden meistverwendeten langreichweitigen Wechselwirkungen sind die Hydrodynamik und die Elektrostatik, und obwohl beide in dieser Arbeit angewandt werden, konzentrieren wir uns auf die Elektrostatik.

Im Laufe der Jahre wurden viele Algorithmen präsentiert, um das Problem der Poisson-Gleichung für Elektrostatik,

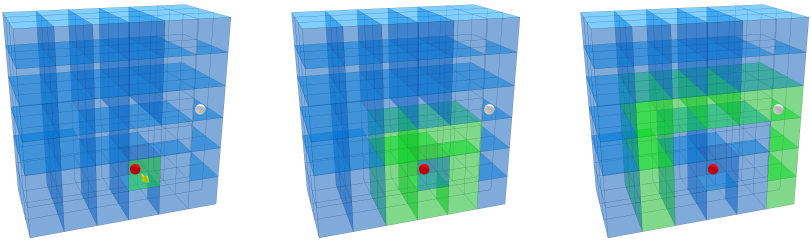
$$\nabla \varepsilon \nabla \Phi = -\rho \tag{0.3}$$

zu lösen. Hier bezeichnet $\Phi$ das gesuchte elektrostatische Potential der Ladungsverteilung $\rho$, und $\varepsilon$ die dielektrische Permittivität des Systems. Ein recht erfolgreicher Ansatz, dessen Variante P3M wir in dieser Arbeit auch verwenden, besteht darin, den langreichweitigen Anteil der Poisson-Gleichung (0.3) in den Fourier-Raum zu transformieren. In diesem ist nicht nur die Lösung der Gleichung numerisch deutlich einfa-

cher, sondern die Periodizität des Systems ist bereits eine Voraussetzung, und sogar die anschließende Differentiation $\mathbf{E} = -\nabla\Phi$ des Potentials stellt nur eine Multiplikation dar. Ein solches Vorgehen ermöglicht allerdings die Hinzunahme von lokal veränderlichen dielektrischen Konstanten $\varepsilon$ nur auf Umwegen, da räumliche Veränderungen der Gleichung nur schwer in den Fourier-Raum übersetzbar sind.

Eine recht junge Methode zur Berechnung elektrostatischer Wechselwirkungen wurde 2002 von A. Maggs vorgeschlagen. Sie basiert auf der Idee, eine Art Elektrodynamik durchzuführen statt instantan wechselwirkender Elektrostatik, die das Limit der Elektrodynamik für unendlich große Lichtgeschwindigkeit ist. Bereits 1985 hatten R. Car und M. Parrinello die Dynamik von Atomkernen simuliert, indem sie den dazugehörigen Elektronen eine unrealistisch hohe Masse gaben und ihre Geschwindigkeit damit drastisch drosselten. Dadurch rückten die Relaxationszeiten der Atomkerne und Elektronen in vergleichbare Regionen und die Simulationen konnten zu relevanten Zeitskalen geführt werden. Es wurde gezeigt, dass hierbei die Dynamik der Elektronen zwar unrealistisch ist, die Dynamik der Teilchen und vor allem der ausgelesenen Observablen des Systems aber intakt blieb. Der Algorithmus von A. Maggs behandelt nun elektrodynamische Wechselwirkungen auf eine ähnliche Weise, indem den Wechselwirkungsteilchen, den Photonen, ein masseartiger Vorfaktor zugewiesen wird, der die Dynamik der elektromagnetischen Felder bremst. Der originale Algorithmus wurde für Monte-Carlo-Simulationen mit gitterbasierten Teilchenbewegungen und das diffusive Ausbreiten der elektromagnetischen Felder aufgestellt, wurde aber parallel von I. Pasichnyk und B. Dünweg sowie J. Rottler und A. Maggs bereits 2004 auf eine wellenförmige Ausbreitung und kontinuierlich bewegliche Ladungsträger erweitert und damit für Molekulardynamik greifbar. Diese erweiterte Variante wurde „Maxwell Equations Molecular Dynamics“ (MEMD) genannt.

Hierbei folgen die Bewegungsgleichungen für Felder und Teilchen den bekannten Maxwell-Gleichungen und reduzieren das numerische Problem dadurch von einer elliptischen partiellen Differentialgleichung (der Poisson-Gleichung (0.3)) zu einem Satz von hyperbolischen Differentialgleichungen. Diese haben den großen Vorteil, vollständig lokal auswertbar zu sein. Die Lösung der Gleichungen erfordert also in jedem



**(a)** $\dot{\mathbf{D}} = -\mathbf{j}/\varepsilon$ **(b)** $\ddot{\mathbf{B}} = c^2 \nabla \times \nabla \times \mathbf{B}$ **(c)** $\dot{\mathbf{D}} = c^2 \nabla \times \mathbf{B}$

**Abbildung 0.2.:** Skizze der Idee hinter MEMD: **(a)** Eine bewegte Ladung erzeugt einen lokalen Stromfluss $\mathbf{j}$ und damit eine Änderung des $\mathbf{D}$-Feldes. Diese Änderung wird per Rotation $\dot{\mathbf{B}} = -\nabla \times \mathbf{D}/\varepsilon$ auf das $\mathbf{B}$-Feld übertragen und **(b)** wellenartig auf einem Gitter ausgebreitet. **(c)** Bei einem zweiten Teilchen wird das $\mathbf{B}$-Feld wieder zurück auf das $\mathbf{D}$-Feld rotiert und die Lorentz-Kraft $\mathbf{F}_L = q\mathbf{E} + \mathbf{v} \times \mathbf{B}$ auf das Teilchen berechnet.

Zeitschritt nur Informationen von den nächsten Nachbarzellen. Felder und damit auch Wechselwirkungen werden schrittweise über das Gitter ausgebreitet, wie in Abbildung 0.2 skizziert.

Das führt selbstverständlich zu retardierten Lösungen und die Information über die aktuelle Teilchenkonfiguration ist an jeder Stelle des Systems leicht zeitverzögert. Genau wie bei der Methode von R. Car und M. Parrinello mit Atomkernen und Elektronen führt dies nur zu einer unrealistischen Repräsentation der Felddynamik, die Teilchendynamik bleibt im thermodynamischen Limit davon unberührt solange die Geschwindigkeit der Ladungen deutlich, d. h. mehr als eine Größenordnung, unter der Ausbreitungsgeschwindigkeit der Magnetfelder liegt. Und da Teilchen sich in Molekulardynamiksimulationen nur sehr wenig pro Zeitschritt bewegen, ist diese Bedingung in allen Situationen gegeben, andernfalls gibt unsere Implementation des Algorithmus einen Fehler aus.

Ein bisher nicht genannter Aspekt innerhalb der Molekularsimulationen ist das stetige Vergröbern von Systemen. Ein DNA-Strang in wässriger Lösung kann mit heutigen Computern nicht mit der gleichen Auflösung und Genauigkeit simuliert werden oder die gleichen

relevanten Zeitskalen erreichen wie ein einzelnes Protein oder gar nur wenige Moleküle. Dementsprechend ist ein wichtiger Aspekt der skalenübergreifenden Teilchensimulation das Vergröbern oder Zusammenfassen mehrerer Teilchen in eine gröbere Repräsentation. Dabei ist darauf zu achten, dass man die für das System relevanten Eigenschaften der Teilchen nicht verliert, sondern in der vergröberten Darstellung nach wie vor beachtet. Dies geschieht normalerweise durch entsprechend angepasste klassische Wechselwirkungspotentiale zwischen den Teilchen, die entweder durch detailliertere Simulationen oder durch experimentelle Bestätigung entwickelt und geeicht werden.

Ein in der Biophysik allgegenwärtiges Beispiel ist Wasser. In atomistischen Simulationen wird ein Wassermolekül normalerweise durch drei oder mehr Punktladungen für die einzelnen ionisierten Atome sowie durch ein umschließendes kugelsymmetrisches Potential dargestellt, üblicherweise ein Lennard-Jones-Potential, das auf kurzen Entfernungen stark abstoßend wirkt und auf längeren leicht anziehend. In vergröberten Modellen können die einzelnen Wassermoleküle nicht mehr so detailreich betrachtet werden und werden zu größeren Teilchen zusammengefasst oder sogar durch teilchenfreie Gittermethoden repräsentiert. Dabei ist es wichtig, zwei zentrale Eigenschaften des Wassers unbedingt beizubehalten. Zum einen sind das die hydrodynamischen Wechselwirkungen, die in viskosen Flüssigkeiten zwischen zwei Teilchen ausgetauscht werden. Hierfür werden in unseren Simulationen mesoskopische Methoden wie „Dissipative Particle Dynamics“ (DPD) oder „Lattice-Boltzmann“ (LB) verwendet, die im Laufe dieser Arbeit kurz erklärt werden, deren Komplexität aber den Rahmen dieser Zusammenfassung sprengen würde. Zum anderen sind aus elektrostatischer Sicht Wassermoleküle an sich zwar neutral, tragen aber ein kleines Dipolmoment. Dieses Dipolmoment hat die Tendenz, sich entlang jeglicher elektrischer Felder auszurichten um seine elektrostatische Energie zu minimieren, wie in Abbildung 0.3 dargestellt.

In Simulationen wird diese Polarisierbarkeit eines Materials oft durch die dielektrische Permittivität $\varepsilon$ beschrieben. Sie kann tensoriell sein, eine Betrachtung als Skalar ist jedoch für biophysikalische Systeme normalerweise ausreichend. Dementsprechend tritt sie einfach als Vorfaktor bei elektrostatischen Wechselwirkungen auf. Reines Wasser hat

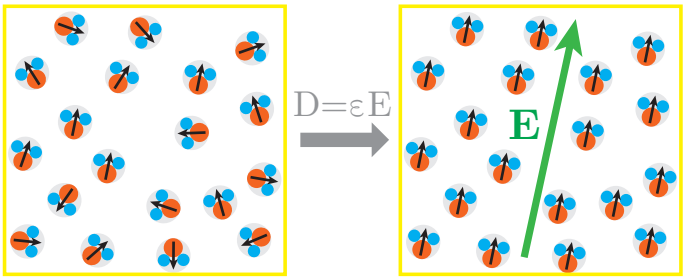



**Abbildung 0.3.:** Jedes Wassermolekül trägt ein kleines Dipolmoment. Bei Anlegen eines elektrischen Feldes werden sich diese Dipole entgegen des Feldes ausrichten um ihre elektrostatische Energie zu minimieren. Diese Polarisation spannt ein Feld in die Gegenrichtung auf und schwächt effektiv alle **E**-Felder und damit die elektrostatischen Wechselwirkungen des Systems ab. Diese Dämpfung kann durch die dielektrische Permittivität $\varepsilon$ als Vorfaktor vor dem Feld ausgedrückt werden. Diese Skizze vereinfacht die Zusammenhänge natürlich stark, die Dipole werden nicht derart strikt ausgerichtet, sondern zeigen nur eine leichte Tendenz zu dieser Ausrichtung auf, die durch thermisches Rauschen überdeckt wird.

üblicherweise eine dielektrische Permittivität von etwa $\varepsilon_W = 80$, was einer starken Dämpfung der elektrostatischen Wechselwirkung um einen Faktor 80 entspricht, während biologische Membrane, Kolloide oder Polymere eher ungefähr $\varepsilon_C = 2$ aufweisen. Dieser Unterschied stellt Elektrostatikalgorithmen, besonders die auf Fouriertransformation basierenden Methoden, oft vor Probleme, da sich die Ortsabhängigkeit der Permittivität nicht leicht in den **k**-Raum übertragen lässt. Um entsprechende Grenzflächen zwischen zwei Dielektrika darzustellen, werden oft virtuelle zusätzliche Ladungen eingeführt, etwa Spiegelladungen auf der anderen Seite der Grenzfläche zu jeder im System vorhandenen Ladung, oder Randladungen, die auf der Grenzfläche sitzen und sich iterativ so verändern dass die vorgegebenen Randbedingungen für elektrische Felder an der Grenze eingehalten werden. In biophysikalischen Systemen treten aber eben häufig Situationen auf, in denen sich die dielektrische Permittivität nicht schrittweise ändert sondern graduell. Gerade in der Nähe von stark geladenen Grenzflächen oder bei hohen Ladungskonzentrationen durch Salzionen sind die Wassermoleküle

in ihrem Rotationsfreiheitsgrad stark eingeschränkt und können nicht mehr die selbe Dämpfung durch Ausrichtung entgegen der Feldlinien aufbringen. Die lokale Permittivität sinkt in diesem Fall signifikant, bis zu einem Faktor zwei auf $\varepsilon = 40$. Unseres Wissens gab es bisher keinen Algorithmus, der solche Systeme dynamisch simulieren konnte. Der oben beschriebene MEMD-Algorithmus erscheint aber durch seine Lokalität gut dafür geeignet zu sein.

In dieser Dissertation erweitern wir den MEMD-Algorithmus, so dass er beliebig räumlich und zeitlich veränderliche dielektrische Permittivität enthalten kann. Wir beweisen, dass die thermodynamischen Observablen nach wie vor korrekt berechnet werden, dass der Algorithmus an sich energie- und impulserhaltend ist und dass unsere zeitliche und räumliche Diskretisierung diese Erhaltungssätze nicht bricht. Wir erweitern den Algorithmus zudem um einen Initiallöser und jeweils einen Korrekturterm für die auftretende Selbstenergiewechselwirkung und das Monopol- und Dipolmoment des Gesamtsystems. Wir stellen zwei Implementationen des Algorithmus vor, in dem Molekulardynamikpaket ESPResSo und in ScaFaCoS, einer Bibliothek mit numerischen Methoden für langreichweitige Wechselwirkungen. Für beide Implementationen liefern wir eine Verifikation des Algorithmus an einer Vielzahl von Systemen, im Vergleich sowohl mit analytischen als auch mit numerischen Lösungen. Abschließend vergleichen wir den Algorithmus und unsere Implementation im Bezug auf Möglichkeiten und serielle und parallele Geschwindigkeit mit etablierten Elektrostatikmethoden.

Im letzten Kapitel stellen wir einige Anwendungen vor, für die der erweiterte MEMD-Algorithmus geeignet ist und die Ergebnisse, die wir aus dazugehörigen Simulationen erhalten haben. Wir tasten uns hierbei langsam durch drei Systemtypen: Zum einen Elektrophorese in Nanoporen, die scharfe dielektrische Grenzflächen enthalten und auch mit alternativen Algorithmen in ESPResSo und auch außerhalb gelöst werden können. Zweitens ein Kolloid in wässriger Lösung, da wir mit einem Kollegen einen zweiten Algorithmus entwickeln konnten, mit dem sich kugelsymmetrische Verteilungsfunktionen für die Permittivität simulieren lassen. Und drittens ein flexibles Polyelektrolyt, bei dem die lokale dielektrische Konstante $\varepsilon(\mathbf{r}, t)$ sowohl räumlich als auch zeitlich flexibel

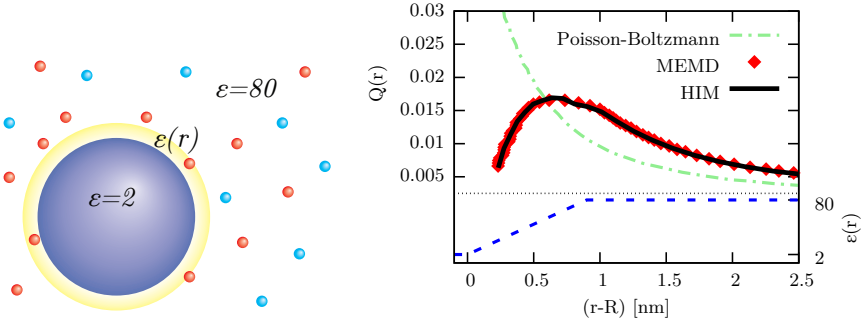
und während der Simulation berechnet wird. Auf diese Weise haben wir versucht, dem typischen „bottom-up“-Design von vergröberten Molekularsimulationen zu folgen, bei dem neue und bisher nicht zugängliche Ergebnisse gründlich verankert werden durch Vergleichssimulationen mit bekannten Größen oder alternativen Algorithmen.

Polyelektrolyte, wie zum Beispiel einzelsträngige DNA in nanoskaligen Schlitzporen, sind nicht nur experimentell zugänglich, sondern stellen ein sehr interessantes System dar. Die Wechselwirkung des Polymers mit den Wänden stellt eine effektive Reibung dar, die stark von der Länge des Polymers und der Porenhöhe abhängt. Auf diese Weite bietet sich die Möglichkeit, ein externes elektrisches Feld entlang einer Schlitzpore anzulegen, und aus der Geschwindigkeit, mit der sich die geladenen Polymere bewegen, auf deren Länge rückzuschließen. Dies ist beispielsweise für DNA-Separation wichtig, die ein sehr aktives Gebiet in der medizinischen Physik darstellt. In dieser Arbeit untersuchen wir zuerst ein Polyelektrolyt in einer unendlichen Schlitzpore, an das ein externes Feld angelegt wird. Wir zeigen, dass das Hinzuschalten von dielektrischen Kontrasten an den Wänden der Pore die Debye-Schicht verbreitert, in der sich Gegenionen ansammeln und entlang des Feldes bewegen. Dadurch wird das in die Gegenrichtung bewegte Polyelektrolyt weiter in die Porenmitte getrieben, was eine stärkere effektive Reibung mit der Wand impliziert. Die Ergebnisse wurden hierbei durch Verwendung eines alternativen Algorithmus in ESPResSo, ICMMM2D, bestätigt.

Mit dem zweiten Nanoporen-System stellten wir direkt einen experimentellen Aufbau nach, dessen überraschende Resultate bis heute nicht hinreichend in Simulationen erklärt werden konnten. 2007 beobachteten M. Krishnan und ihre Kollegen, dass negativ geladene Polyelektrolyte in sehr schmalen negativ geladenen Nanokanälen in die Ecke des Kanals gezogen werden und sich dort festsetzen. Das ist zunächst nicht intuitiv, da die Polymere sowohl durch diese Adsorption entropische Freiheitsgrade verlieren als auch von den gleichnamig geladenen Wänden abgestoßen werden. In Simulationen mit zwei verschiedenen statischen Lösern für partielle Differentialgleichungen sowie auch in molekulardynamischen Simulationen mit dem MEMD-Algorithmus konnten wir dieses Verhalten reproduzieren, wenn auch nur für eine

eingeschränkte Systemklasse.

2013 besuchte Zhenli Xu das Institut für Computerphysik und zeigte sich interessiert an Simulationen mit veränderlichem dielektrischem Hintergrund. Er entwickelte während seiner Zeit am Institut einen von ihm publizierten Algorithmus zur „Harmonic Interpolation Method“ (HIM) weiter, mit der sich fast beliebige kugelsymmetrische Funktionen für die dielektrische Konstante simulieren ließen. Er implementierte die Methode in seinen eigenen Monte-Carlo-Code und wir simulierten parallel ein exakt identisches System, das in Abbildung 0.4a skizziert ist.




**(a)** Skizze des Simulationsaufbaus

**(b)** Radiale Ladungsverteilung

**Abbildung 0.4.:** **(a)** Ein geladenes Kolloid (Ladung $Z = 60$, Radius $R = 30$) ist in Wasser mit $c = 50\,\mathrm{mM}$ Salzkonzentration gelöst. Wir beschreiben die dielektrische Permittivität mit dem MEMD- und dem HIM-Algorithmus durch einen linearen Anstieg von der Kolloidoberfläche zu freiem Wasser innerhalb von 2 Ionendurchmessern. Die resultierende radiale Ladungsverteilung **(b)** unterscheidet sich erheblich von Simulationen mit konstantem dielektrischem Hintergrund, eingezeichnet als Poisson-Boltzmann, oder einem schrittartigen Anstieg auf der Kolloidoberfläche, während sich die Ergebnisse der beiden verwendeten, völlig unterschiedlichen Algorithmen gegenseitig bestätigen.

Ein Kolloid mit starker Oberflächenladung befindet sich hierbei in einer wässrigen Lösung mit endlichem Salzgehalt. Innerhalb des Kolloids wird die Permittivität auf $\varepsilon_C = 2$ festgelegt, in freiem Wasser auf

$\varepsilon_W = 80$. In einer Übergangsregion simulieren wir aber verschiedene abstandsabhängige Funktionen $\varepsilon(r)$ und deren Einfluss auf die radiale Ladungsverteilung in der Umgebung des Kolloids. Ein einfaches, jedoch recht effektives Modell ist eine lineare Interpolation der Permittivität von der Kolloidoberfläche innerhalb von zwei Ionendurchmessern. Die hierbei entstehenden Ergebnisse (Abbildung 0.4b) weichen nicht nur deutlich von der Poisson-Boltzmann-Verteilung ab, sondern stimmen auch für beide Methoden sehr gut überein. Das schenkt uns Zuversicht, dass zumindest auf kurze Entfernungen die Variation der dielektrischen Konstante merklichen Einfluss auf physikalische Observablen haben kann, und dass der erweiterte MEMD-Algorithmus sowie unsere Implementation davon zuverlässige Ergebnisse liefern. Wir können auch zeigen, dass dieser kurzreichweitige Unterschied sich makroskopisch in der „effektiven Ladung“ des Kolloids bemerkbar macht, für die wir bis zu 7% quantitativen Unterschied feststellen.

Wie wir bereits erwähnt haben, ist die dielektrische Permittivität von Wasser nicht konstant, sondern reduziert sich in der Nähe von Ladungsträgern, auch von gelösten Salzionen. Nicht nur dass Hydrathüllen um die Ionen gebildet werden, in denen das Wasser faktisch gebunden und gar nicht mehr flexibel ist, sondern das starke Feld der Ionen beeinflusst auch den Rotationsfreiheitsgrad der Wassermoleküle in der näheren Umgebung. B. Hess et al. veröffentlichten dazu Messungen, die die Permittivität direkt mit der Salzkonzentration im Wasser in Verbindung bringen und lieferten dazu auch eine einfache Formel, die an die Messpunkte gelegt wurde. Wir können also für die letzte in dieser Arbeit präsentierte Anwendung direkt aus einer lokalen Salzkonzentration eine dielektrische Permittivität berechnen und mit MEMD anwenden. Ein dafür gut geeignetes und interessantes System sind flexible Polyelektrolyte in wässriger Salzlösung.

Da wir mit einem solchen Algorithmus Neuland betreten und es dazu keine Referenzsimulationen mehr gibt, gehen wir schrittweise vor. Im ersten Ansatz fixieren wir ein Polyelektrolyt auf einer geraden Linie im Raum. Innerhalb des Polymers, also bei einem Abstand kleiner seinem Radius, wird die Permittivität auf $\varepsilon_C = 2$ gesetzt, außerhalb soll sie flexibel entsprechend der lokalen Salzkonzentration berechnet werden. Wir beginnen mit einem flachen Profil und berechnen hieraus lokal die

Volumenkonzentration der Gegenionen an jeder Stelle um das Polymer herum. Das liefert uns über die semiempirische Formel von B. Hess et al. eine Permittivität, die wir in den MEMD-Algorithmus einsetzen. Damit wird dann die dynamische Simulation gestartet. Während der Simulation speichern wir die daraus resultierende Verteilung von Gegenionen als zylindersymmetrisches Profil. Im nächsten Schritt können wir dieses Ergebnis wieder auf eine neue Permittivitätsverteilung projizieren. So erhalten wir iterativ eine realistische Ionenverteilung um das Polyelektrolyt und eine dazugehörige Permittivität. Dieses Schema konvergiert und liefert ein Profil, das dem um das Kolloid (siehe Abbildung 0.4b) nicht unähnlich sieht, entsprechend dazu einen nahezu linearen Anstieg der Permittivität.

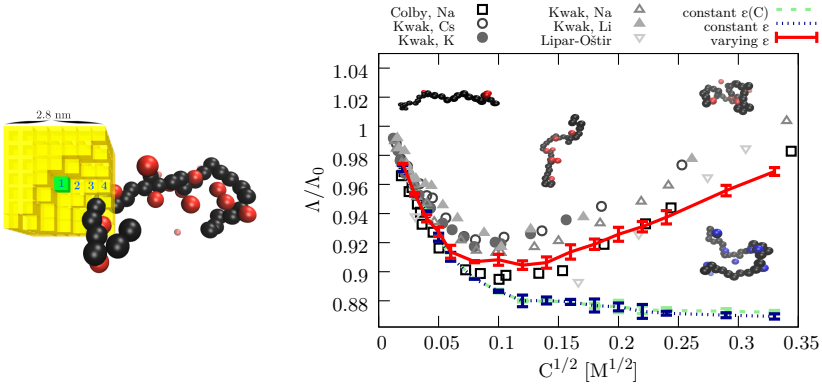
Im nächsten Schritt verändern wir dieses langsame iterative Schema zu einem adaptiven Schema, das während der Simulationslaufzeit lokal die Ladungskonzentration berechnet und die dielektrische Permittivität in jedem Zeitschritt dementsprechend anpasst. Wir wenden dieses adaptive Schema zunächst auch auf den linear eingefrorenen Polyelektrolyten mit Gegenionen an, es ist aber als Skizze links oben in Abbildung 0.5a mit einem flexiblen Polyelektrolyten zu sehen. Wir mitteln hierbei gewichtet über die Anzahl der Ladungen in einem Volumen mit $7^3$ Gitterzellen um jede Zelle, so dass die näher gelegenen Zellen mehr Einfluss ausüben. Hieraus erhalten wir eine lokale Salzkonzentration und damit die Permittivität, die wir in jedem Zeitschritt berechnen und einsetzen können. Man muss hierbei darauf achten, dass sich die dielektrische Konstante einzelner Links nicht sprunghaft ändert, da derartiges Verhalten Artefakte und unphysikalische Ergebnisse im Algorithmus erzeugen kann. Durch die Gewichtung und das relativ große Volumen entstehen aber im Normalfall keine Sprünge, wenn Ladungen in dieses Volumen hinein- oder aus ihm herauslaufen. In SI-Einheiten entspricht eine Gitterzelle etwa 0.4 nm, das Volumen des Würfels aus $7^3$ Zellen ist also etwa ($2.8\,\mathrm{nm}^3$, was einer Reichweite der Konzentrationsberechnung von zwei typischen Bjerrumlängen $l_B = 0.7\,\mathrm{nm}$ in Wasser entspricht, die Entfernung ab der zwei Teilchen mit Einheitsladung gebundene Paare bilden. Die Ergebnisse der Simulationen mit adaptivem Schema liefern ein fast identisches Ergebnis zur iterativen Herangehensweise, was die Korrektheit beider komplett unterschiedlicher Schemata

nahelegt.

Dieses adaptive Schema ist nun geeignet, um auch Simulationen mit flexiblen Polyelektrolyten durchzuführen. Unsere ersten Simulationen sind freie Polyelektrolyte in salzfreier wässriger Lösung, ausschließlich mit Gegenionen. Die hydrodynamischen Wechselwirkungen werden hierbei mit der Lattice-Boltzmann Implementation in ESPResSo berechnet. Wir untersuchen zu Beginn die zylindrischen Verteilungsfunktionen um das Rückgrat des Polymers, um sie mit den bisherigen Ergebnissen zu vergleichen. Hierfür entwickeln wir ein einfaches Verfahren, das an jedem Monomer entlang des Rückgrats die lokale Ausrichtung des Polymers bestimmt und darum herum zylindrisch die Ladungsdichte aufsummiert. Die Ergebnisse stimmen innerhalb der Fehlergrenzen mit den Verteilungsfunktionen für gestreckte Polymere überein. Wir betrachten außerdem einige typische Polymereigenschaften wie den Gyrationsradius und die Diffusionsgeschwindigkeit in Simulationen mit und ohne variierende dielektrische Permittivität. Die Unterschiede sind hier jedoch vernachlässigbar.

Die Struktur der elektrochemischen Doppelschicht weist allerdings deutliche Veränderungen auf. Gegenionen werden durch den Permittivitätsgradienten stärker vom Rückgrat des Polymers ferngehalten und die gesamte Doppelschicht ist merklich verbreitert. Ein experimenteller Aufbau, in dem die Struktur der elektrochemischen Doppelschicht eine entscheidende Rolle spielt, und den wir bereits in anderem Zusammenhang betrachtet haben, ist die Elektrophorese, also das Anlegen eines externen elektrischen Feldes. Die Bewegung des Polyelektrolyts in die eine und der Gegenionen in die andere Richtung ist bestimmt durch die hydrodynamische und elektrostatische Kopplung zwischen den beiden Spezies. Es ist zu erwarten, dass die elektrostatische Kopplung bei Variation der dielektrischen Permittivität merklich schwächer ausfällt, da die Gegenionen im Durchschnitt weiter vom Polyelektrolyten entfernt und damit weniger stark gebunden sind. Tatsächlich sehen wir quantitative Unterschiede um bis zu 20% bei der elektrophoretischen Mobilität, der effektiven Ladung und schließlich der elektrischen Leitfähigkeit des Systems. Wir führen hierzu Simulationen bei verschiedenen Polymerlängen und Salzkonzentrationen durch und kommen konsistent zu ähnlichen Ergebnissen.

Die Leitfähigkeit des Polyelektrolytsystems hängt neben der Länge des Polymers vor allem auch von der Monomerkonzentration im System ab. Wir variieren diese indem wir bei gleichbleibender Systemgröße mehr Polyelektrolyte gleichen Typs hinzufügen. Hier sehen wir zum ersten Mal einen qualitativen Unterschied zwischen den Simulationen mit konstanter und jenen mit lokal veränderlicher dielektrischer Permittivität, wie an den farbigen Linien in Abbildung 0.5b dargestellt ist.




**(a)** Systemskizze **(b)** Verlauf der Leitfähigkeit

**Abbildung 0.5.:** **(a)** Wir simulieren flexible Polyelektrolyte in salzfreier wässriger Lösung, wobei während der Simulation die lokale Ladungskonzentration in jeder Gitterzelle aus den umgebenden $7^3$ Zellen berechnet wird und daraus die dielektrische Permittivität bestimmt und gesetzt wird. **(b)** Das experimentelle Verhalten (graue Symbole), wonach die Leitfähigkeit pro Monomer $\Lambda$ des Systems bei steigender Monomerkonzentration zuerst ab- und dann wieder zunimmt, kann bei veränderlicher Permittivität (rote Linie) reproduziert werden. Die Simulationen mit konstanter Permittivität (blaue Linie) zeigen ein anderes qualitatives Verhalten. Der Grund dafür ist das Zusammenrollen der Polymere (siehe Bildeinschübe), das bei hohen Salzkonzentrationen zu höherer Ladungskonzentration und damit niedrigerer lokaler Permittivität führt. Das Ergebnis sind weniger kondensierte Gegenionen (rot respektive blau) und mehr freie Ionen, was zur höheren Leitfähigkeit führt. Die Grafiken wurden in [6] veröffentlicht.

Für Monomerkonzentrationen bis etwa 0.01 M sinkt die Leitfähigkeit des Systems stetig, da die höhere Salzkonzentration zu mehr gebundenen Gegenionen führt, die am Rückgrat des Polyelektrolyten kondensiert sind. Für noch höhere Salzkonzentrationen allerdings fällt die Leitfähigkeit der Simulationen mit konstantem dielektrischem Hintergrund weiter stetig, während die für variierende Permittivität wieder ansteigt. Grund dafür ist, dass der Polyelektrolyt sich bei hohen Salzkonzentrationen einrollt, wie im kleinen Bild rechts oben in Abbildung 0.5b zu sehen. Dadurch nimmt die Ladungskonzentration in dieser Region deutlich zu und es entsteht ein Permittivitätsminimum, das die Gegenionen zu meiden versuchen. Bei näherem Betrachten sehen wir, dass die Anzahl der kondensierten Gegenionen bei veränderlicher dielektrischer Konstante ab etwa dem Punkt des Leitfähigkeitsminimums wieder kleiner wird. Es gibt somit mehr freie Gegenionen und zudem weniger starke Kopplung zwischen den beiden Spezies, was direkt zu einer höher werdenden Leitfähigkeit führt.

Abschließend vergleichen wir unsere Ergebnisse für die Leitfähigkeit flexibler Polyelektrolyte noch mit experimentellen Daten von Polymeren in salzfreier wässriger Lösung, siehe dazu die grauen Symbole in Abbildung 0.5b. Die Leitfähigkeiten sind hier auf den extrapolierten Wert bei Monomerkonzentration $C = 0$ normiert, da der absolute Wert sehr stark vom hydrodynamischen Radius der verwendeten Gegenionentypen abhängt. Die Simulationen mit veränderlicher Permittivität reproduzieren auf beeindruckende Weise das experimentell beobachtete Verhalten, während die gleichen Simulationen mit konstantem dielektrischen Hintergrund ein qualitativ anderes Verhalten aufweisen.

**Zusammenfassung** Wir haben in dieser Dissertation den MEMD-Algorithmus um veränderliche Permittivität erweitert, implementiert, parallelisiert, verifiziert und im Hinblick auf Ergebnisse und Leistungsfähigkeit mit anderen Algorithmen für Elektrostatik verglichen. Ferner haben wir mit Hilfe dieser Implementation verschiedene Klassen von biophysikalischen Systemen untersucht und dabei sowohl kleine als auch signifikante Einflüsse gefunden, die das Berücksichtigen von lokal veränderlichen Dielektrika auf physikalische Observablen hat.

# 1. Introduction

## Computer simulations

In the course of several decades, computer simulations have become a well established and important tool to gain insight into physical processes [200]. This includes the study of experimentally inaccessible systems as well as theoretical predictions for complex setups that are not solvable analytically. The advancements made in this field and its applications are vast and keep on growing. The development of suitable software tools for the study of more and more complex systems aims to extend simulation methods into larger scales and more detailed representations. The inclusion of previously disregarded physical properties can lead to more accurate results, while the ability to simulate larger systems and longer time scales opens up the research area to new and industrially interesting physical problems.

A straight forward way to extend the reach of computer simulations to larger scales is the development and production of more powerful computing devices, which has proven to be very effective. Following Moore's law [25], the computational abilities of accessible computers have been growing at an exponential rate and with them the system sizes accessible to numerical simulations. In the last two decades, the focus has shifted from increasing the performance of single computer processors to an increase of the number of cores coupled in one supercomputer cluster. Simulations can then run on many cores at the same time, and the parallelization of computer software has greatly expanded the limits of computer simulations. At the same time, research in applied mathematics has significantly increased the efficiency of time consuming calculations by means of steadily improved numerical algorithms. It has even been suggested that the algorithmic speedups since the beginning of numerical simulations are at a similar level as

their computational counterparts [248, p. 71]. Given the shift towards many-core computing, the parallelizability of algorithmic approaches has become an ever increasing focus of current research.

Another way of expanding simulations to larger scales is by means of coarse-graining. The idea is to reduce the amount of computations by combining several particles into a bigger particle or cell and decreasing the number of interactions and therefore the complexity of the system. This is a very powerful tool, but one has to be careful to include all relevant physical properties of the system in order to obtain reasonable results. In algorithmic development, one hopes to find methods that will retain important physical interactions while operating at a coarser scale. One particular property that could be neglected in coarse-grained simulations is the polarizability of a material, and its corresponding dielectric permittivity when calculating electrostatic interactions within the system.

### Biophysical systems

Since the beginning of computer simulations, biophysics and particularly soft matter have been a very active research area for several reasons. One reason is that this field of research includes investigations of the building blocks of life. Ever since the book "What is life?" [69] by Erwin Schrödinger was published in 1944, physicists have shown more and more interest in this field, studying systems such as colloidal suspensions, polyelectrolytes e. g. DNA or RNA, cell nuclei, or even small bacteria with great success. The multidisciplinarity of this research area has spawned many workgroups and collaborations at universities all over the world. Another reason is that soft matter is very well suited for computer simulations and especially Molecular Dynamics (MD) simulations. This is due to the fact that in soft matter processes, the time scales on which equilibration and interesting phenomena take place are often in the same order of magnitude. This thesis mostly deals with MD simulations and the algorithms used therein, so it will be explained in more detail in section 2.1. For now, the key aspect to know about MD simulations is that the Newtonian equations of motion are numerically integrated for systems containing

discrete particles that interact with each other via pair forces or classical potentials. The majority of the computational effort is spent on the calculation of forces acting on each particle.

This force calculation can usually, for numerical purposes, be split into the calculation of short-range interactions and long-range interactions. Short-range interactions are defined as forces which in three dimensions decay faster than $1/r^3$ with distance $r$ and their shell-wise sum converges fairly quickly in real space and the interaction can be cut off at finite distances without introducing an additional error in the numerical integration. Long-range interactions on the other hand, which decay slower than $1/r^3$ in three dimensional space, can not simply be cut off without introducing a significant error in most systems and therefore often require much more computational effort. In addition, computer simulations of biophysical systems will often be carried out with the use of periodic boundary conditions, assuming periodic replication of the system in all three dimensions to avoid boundary artifacts. This introduces an infinite sum that is merely conditionally convergent in general cases, a problem that several mathematical approaches have solved in different ways, particularly for the often encountered $1/r$ Coulomb potential [2].

In biophysical MD simulations, there are two types of long-range interactions that need to be considered. The first are hydrodynamic interactions in systems including implicit fluids, but only a short introduction to different approaches to hydrodynamic interactions will be given in this thesis, since they will only play a minor role in chapter 6. The second are electrostatic interactions resulting from charged objects within the system, and they will be the main focus of this thesis.

One of the key components of life and therefore biology is water. The systems examined in simulational biophysics usually contain water that is either considered insignificant for the underlying physical process or treated in an explicit or implicit manner. In addition, this water usually has a finite salt concentration, allowing for important biological effects such as osmosis. Another important feature of water, being a polar solvent, is that if a biomolecule is put in aqueous solution, some of its dissolvable groups will most likely be solvated, leading to a charged biomolecule and additional salt ions within the

solution. This means that most biophysical simulations will have to include a model for water as well as long-ranged electrostatic interactions between charged objects. Accordingly, a pivotal aspect of algorithmic research in biophysical simulations is the improvement of speed and accuracy for hydrodynamic and electrostatic interactions, both of which are long-range in nature.

## Suitable algorithmic approach

In computer simulations, there are several successful approaches to deal with electrostatic interactions in periodic systems [2]. Some of the most important ones account for the periodicity of the system by means of mathematical tricks. Most commonly used algorithms in Molecular Dynamics simulations are mesh-based advancements of the so-called Ewald summation [12]. Typically, the interaction is split into a short-range part which can be calculated locally pairwise and a long-range part which will be Fourier transformed and solved in Fourier space. This reduces the necessary integration and differentiation to a multiplication, and after back transformation will guarantee a periodic potential. When combined with a lattice interpolation and Fast Fourier Transform, the algorithm is very efficient and scales as $\mathcal{O}(N \log N)$, where $N$ is the number of charges in the system. One major disadvantage of these Ewald based methods is that they do not allow for any spatial variations of the dielectric properties in the Poisson equation, since this does not easily translate to Fourier space. Such variations have to be dealt with by means of added artificial charges in the system [250] or other additional calculations. This is why, for this work, a different algorithm was chosen as a foundation to build on.

In 2002, A. Maggs proposed an on-lattice Monte Carlo scheme to calculate electrostatic interactions in computer simulations of periodic systems [133]. His idea was to introduce an artificial field that propagates the interaction and any changes throughout the system. The displacement of a charge leads to a change in the artificial field which is then propagated in a diffusive manner with specific Monte Carlo moves and couples to the electric field acting on all other charges. This sounds, and in fact is, very close to a realistic electrodynamic

(and not -static) treatment of the system, and has been referred to as "Maggswellian Dynamics". The original algorithm is hereby restricted to charges moving in discrete Monte Carlo steps on a lattice. It scales linearly $\mathcal{O}(N)$ with the number of lattice sites $N$, and for constant density with the number of charges, and is intrinsically local, potentially allowing for arbitrary changes in the system's dielectric properties.

Because of its locality, this algorithm was developed further and in 2004, J. Rottler et al. [155] and I. Pasichnyk et al. [151] concurrently extended the algorithm to include off-lattice charges and therefore made it suitable for Molecular Dynamics simulations. The version by J. Rottler kept the diffusive Monte Carlo propagation of the artificial field, whereas I. Pasichnyk's implementation propagated the field in a wave-like manner, and we will follow the latter approach which has been coined "Maxwell Equations Molecular Dynamics" (MEMD).

The original implementation by I. Pasichnyk had been included in the ESPResSo ("Extensible Simulation Package for Research on Soft matter") framework [1, 315]. This Open Source software is in active development mostly at the Institute for Computational Physics in Stuttgart and all improvements presented in this thesis were also implemented in this simulation package. ESPResSo also includes several alternative algorithms that can include dielectric boundaries and they serve very well for comparison and verification of the MEMD implementation.

## The work within this thesis

The goal of this thesis was to extend the MEMD algorithm as presented by I. Pasichnyk to include spatially varying dielectric permittivity. This would open the possibility to include details in coarse-grained simulations that had been ignored before. Starting from the implementation found in the ESPResSo package, several problems and errors of numerical and mathematical nature were encountered. The implementation was also planned to be included in ScaFaCoS, a highly scalable electrostatics library for Molecular Simulation. Therefore and because of the extended interface for varying permittivities, the entire code base was rewritten to better suit the purpose of providing a variable dielec-

tric background and get rid of the underlying problems. The code has been optimized, parallelized, and modularized within ESPResSo, and it is included in the ScaFaCoS library [2].

With the inclusion of spatially varying dielectric permittivity to the algorithm, several aspects had to be adapted and all proofs for correctness and mathematical conclusiveness had to be redone. This is presented in a new and comprehensible way in chapter 3. The addition of varying dielectric permittivity has been implemented in the ESPResSo package, including functionality to easily set and adapt it locally and as a whole in some typical geometries.

In chapter 4, the implementation is verified via analytically solvable setups and against several other available algorithms. The algorithm consists of two entirely separate parts: The dynamic algorithm we have talked about thus far, and an initial static solution from which the system is propagated. Both parts are tested against suitable electrostatic systems. The newly implemented algorithm is also compared against other methods in terms of feature set and performance. This has been done using the newly written library ScaFaCoS, a collaborative BMBF ("Bundesministerium für Bildung und Forschung") project including research groups from all over Germany. Especially for such a young and versatile algorithm, MEMD performs quite competitively when compared to other algorithms, as shown in chapter 5.

With this new and verified algorithm at hand, it is of course of high interest to apply its unique features to relevant biophysical systems. In chapter 6, some physical systems are presented in which dielectric boundaries and/or highly charged objects are key components, and the influence and importance of locally varying dielectric properties is examined. It turns out that some physical systems are quite sensible to small changes in electrostatic interactions and under those circumstances, the inclusion of varying dielectric bulk permittivity can actually have a significant influence even on larger scale statistic observables.

Finally, the findings and relevance of this research are summarized in chapter 7, and some ideas for possible future applications and development are presented.

## Published work


1. Arnold, A., Lenz, O., Kesselheim, S., Weeber, R., Fahrenberger, F., Röhm, D., Košovan, P. & Holm, C. *ESPResSo 3.1 — Molecular Dynamics Software for Coarse-Grained Models* in *Meshfree Methods for Partial Differential Equations VI* (eds Griebel, M. & Schweitzer, M. A.) **89** (Springer, 2013), 1–23.
2. Arnold, A., Fahrenberger, F., Holm, C., Lenz, O., Bolten, M., Dachsel, H., Halver, R., Kabadshow, I., Gähler, F., Heber, F., Iseringhausen, J., Hofmann, M., Pippig, M., Potts, D. & Sutmann, G. Comparison of scalable fast methods for long-range interactions. *Phys. Rev. E* **88,** 063308 (6 2013).
3. Arnold, A., Breitsprecher, K., Fahrenberger, F., Kesselheim, S., Lenz, O. & Holm, C. Efficient Algorithms for Electrostatic Interactions Including Dielectric Contrasts. *Entropy* **15,** 4569–4588 (2013).
4. Fahrenberger, F. & Holm, C. Computing the Coulomb interaction in inhomogeneous dielectric media via a local electrostatics lattice algorithm. *Phys. Rev. E* **90,** 063304 (6 2014).
5. Fahrenberger, F., Xu, Z. & Holm, C. Simulation of electric double layers around charged colloids in aqueous solution of variable permittivity. *J. Chem. Phys.* **141** (2014).
6. Fahrenberger, F., Hickey, O. A., Smiatek, J. & Holm, C. Importance of varying permittivity on the conductivity of polyelectrolyte solutions. *Phys. Rev. Lett.* **115,** 118301 (11 2015).
7. Fahrenberger, F., Hickey, O. A., Smiatek, J. & Holm, C. The influence of charged-induced variations in the local permittivity on the static and dynamic properties of polyelectrolyte solutions. *J. Chem. Phys.* **143,** 243140 (2015).

# 2. Theoretical framework

In this chapter, we will present the theoretical background used for the simulations and proofs presented in this thesis. We will give a brief introduction to the methodology of Molecular Dynamics simulations and its most important features. In addition, we will line out some background information on coarse-graining in simulations and the dielectric properties of physical systems. Finally, we will discuss some key considerations to run computer simulations in general and what they mean for algorithmic improvements.

## 2.1. Molecular Simulations

The simulations in this thesis consist entirely of classical particles and potential-based interactions between those particles. Simulations with these properties are generally referred to as *Molecular Simulations* and can be separated into two different conceptual ideas [127], Monte-Carlo and Molecular Dynamics.

### 2.1.1. Monte-Carlo simulations

The concept of Monte-Carlo simulations is very much based on the idea that a system will strive to reach sample states which follow a Boltzmann distribution. The simulations start with a random configuration of the system. The following propagation of the system does not posses any physical meaning as it does in Molecular Dynamics simulations, but is entirely chosen for convenience and efficiency. Thus, Monte-Carlo simulations can be very useful to gain insight into static properties of systems in thermal equilibrium, but are not suited for the study of dynamic observables.

The most famous algorithm for the propagation of a Monte-Carlo system, and the only one that is used in this thesis, is the Metropolis algorithm [19]. In this algorithm, a new system configuration is created by randomly generating a small displacement for one or more particles in the system. The potential energy of this new configuration $V(\tau + \delta\tau)$ is calculated and compared to the potential energy before the displacement $V(\tau)$. The move is accepted with the probability

$$p = \begin{cases} 1 & \text{if} \quad V(\tau + \delta\tau) < V(\tau) \\ \exp\left(-\frac{V(\tau+\delta\tau)-V(\tau)}{k_B T}\right) & \text{if} \quad V(\tau + \delta\tau) > V(\tau), \end{cases} \tag{2.1}$$

where $k_B$ is the Boltzmann constant and $T$ is the assigned temperature of the system.

In such an algorithm, the dynamics are intrinsically unphysical and can be chosen in accordance with the problem at hand, which offers a lot of algorithmic freedom. However, if one is interested in the dynamical properties of a system, Monte-Carlo simulations are not suitable. In addition, they are not easy to efficiently parallelize on several compute cores, since "detailed balance" must be stricly obeyed, for further reading see [127]. For these reasons, this once very popular method had been on the decline in recent years, when parallelization and scalability are of key interest to the scientific community.

### 2.1.2. Molecular Dynamics simulations

The second conceptual idea for Molecular Simulations is called Molecular Dynamics. Again, we simulate classical particles interacting with pair potentials, meaning that the interactions conserve energy. But as the name suggests, the dynamic propagation of the system is much more physical. The idea is indeed very simple: From the pair potentials, one calculates the combined force acting on each particle. Using these forces, the Newtonian equations of motion for an NVE ensemble,

$$m_i \frac{\partial^2}{\partial t^2} \mathbf{r}_i = -\nabla V(\mathbf{r}_1, \ldots, \mathbf{r}_n) \tag{2.2}$$

are numerically integrated, and the system is propagated. Here, $m_i$ are the particle masses, $\mathbf{r}_i$ the particle positions, and $V$ the potential

resulting from the predefined model for particle interactions. For the numerical integration, explicit integration schemes are often preferred over implicit schemes, since the number of degrees of freedom is typically very high. In our simulations, we exclusively used the popular Velocity-Verlet integration scheme. Basically, the velocities $\mathbf{v}_i$ of the particles are updated in two half time steps $\delta t/2$, in the beginning and the end of the scheme, while the particle coordinates and forces are updated between those two steps. This leaves us with a second order time-reversible integration scheme. Time-reversibility is hereby very important, since the phase space volume and therefore energy is conserved. A more detailed information on the Velocity-Verlet integration scheme can be found in [40, 127].

It has to be noted that in the ESPResSo simulation package, as is often the case, the propagation of the system is additionally influenced by a thermostat to create an NVT ensemble. This extends the equation of motion gained from Newton's equation (2.2) to a Langevin equation which includes an additional friction factor and noise term on the right hand side,

$$m\frac{\partial^2 \mathbf{r}}{\partial t^2} = -\lambda\frac{\partial \mathbf{r}}{\partial t} + F_{\text{noise}} \quad , \tag{2.3}$$

where $\lambda$ is a linear rescaling of the velocity due to a frictional or viscous force, and $F_{\text{noise}}$ is a random noise term representing the effect of random particle collisions in a dense system. This equation in and of itself is not time-reversible anymore, in fact the noise term renders the particle trajectories non-differentiable at all points. This does not ruin our integration scheme, since the standard Newtonian equations of motion are recovered in the limit of a small friction term, in which our systems generally operate. But it is worth noting that even in the current implementation of ESPResSo, the dissipative forces are calculated from the velocities at half time steps, and the conservative forces at full time steps. We want to point this out since for the developed algorithm, we will later extend the existing integration scheme and time reversibility will play a role for our approach although it is still partially broken by the Langevin coupling of the thermostat.

Thermostats in general will alter the equations of motion directly by adding a randomly distributed noise to the particle velocities, keep-

ing the kinetic energy of the entire system around a constant value. This can be seen as coupling the entire system to a heat bath, effectively creating a canonical ($NVT$) or grandcanonical ($\mu VT$) ensemble. In our simulations, we used two different kinds of thermostats: The Langevin thermostat is essentially described by the Langevin equation (2.3), where a thermal noise coupling can enter directly. The Lattice-Boltzmann thermostat essentially thermalizes the velocities and even higher modes of an implicit hydrodynamics solver directly, which couples to the particles and keeps the system at a given temperature.

Although Molecular Dynamics simulations can be used to simulate non-equilibrium systems, in this thesis we only examine systems that are in thermodynamic equilibrium. A big advantage of Molecular Dynamics simulations is that we can study specific details of a system via measuring all statistic observables that can be expressed by particle positions and momenta.

## 2.2. Force calculation

Computationally, the propagation of the described Molecular Dynamics simulations mostly consists of two algorithms: The numeric integration scheme and the calculation of forces at the heart of this integration scheme. With a second order time reversible integration scheme like the Velocity-Verlet algorithm, the calculation of forces is the most crucial part of the simulation. Not only does it determine the accuracy with which observables are reproduced, it is also computationally far more costly than the numerical integration.

It is common to split the force calculation into two parts: The *short-range* and the *long-range* interactions. *Long-range* interactions are defined as all interactions whose classic potential decays slower than $r^{-D}$, where $r$ is the distance between a particle pair and $D$ is the number of dimension in the system. For our three-dimensional simulations, $1/r$- and $1/r^2$-interactions are long-ranged, although the latter don't occur very commonly. This distinction is made because short- and long-range interactions can be computationally handled in a different way.

### 2.2.1. Short-range interactions

The gradient of short-range interaction potentials reaches the limits of machine precision so quickly that it can usually be cut off at fairly close distances without introducing any artifacts in the force calculation. Even the energy calculation can be corrected if the potential is shifted by the value it displays at cut-off distance. In other words, the sum over all particle interactions with a spherical summation scheme converges after a finite radius $r_{\text{cut}}$ and all other particles can be neglected.

Technically, this can be achieved by use of so-called Verlet lists. Each particles carries with it a list of particles that is within a sphere of radius $r_{\text{cut}} + r_{\text{skin}}$, where $r_{\text{skin}}$ is the maximum amount particles can move before these lists have to be updated. Then each particle can simply loop over the particles in its Verlet list and calculate the interactions directly.

This also allows for an easy parallelization of the short-range interactions. As depicted in figure 2.1, the system is subdivided into

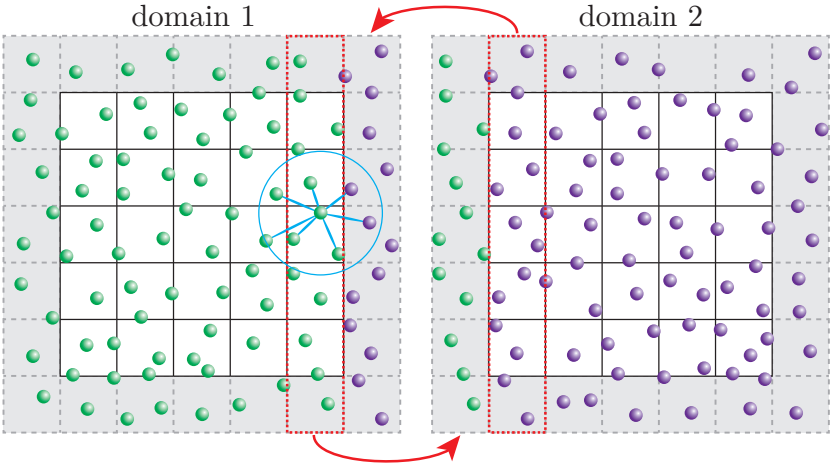



**Figure 2.1.:** Domain decomposition scheme for parallel execution: The system is divided into cuboid spatial domains. Each domain (white) is surrounded by a layer of ghost cells (gray) to directly calculate the short-range interactions of particles close to the domain boundaries. The properties (particle positions, fields, ...) of the outermost cells are communicated to the adjacent domain every time step (red).

spatial domains. Each domain is again subdivided into cells and contains a number of particles and their positions. The domain itself is surrounded by a layer of *ghost cells* that contain information about the directly adjacent cell layer of the neighboring domain. The cells are at least $r_{\text{cut}} + r_{\text{skin}}$ wide, and the particle positions in the ghost cells are communicated every time step, so that the short-range interactions can easily be calculated, even for particles near the boundaries of a domain.

To avoid any boundary effects at the edge of the simulation box, we often apply periodic boundary conditions. This means that the system is replicated infinitely in each direction and is topologically equivalent to a 3D-torus geometry, where the oppositely placed boundaries are connected. This provides a finite system without any physical boundaries, by simply connecting the very top domain boundary to the very bottom domain boundary and so forth.

Long-range interactions on the other hand can not be cut off without

introducing noticeable numeric errors. In fact, a spherical summation in an infinitely big or periodic system only converges conditionally [55, 127], and the results depend on the order of summation. And even if it converges, mathematically, a sum over an infinite amount of particles is required to obtain the correct solution. This is of course impossible, and many algorithmic approaches have been developed over the years to deal with the numerical evaluation of long-range interactions. We will briefly describe a selection of them within this thesis.

It is also noteworthy that long-range interactions are computationally much more costly, and even modern algorithms and computers will use up to 80% of the CPU load for the evaluation. Improving those methods is of high practical relevance and a very active field of research in the simulating community.

### 2.2.2. Electrostatics

A prime example of a long-range interaction is the electrostatic potential. And since most biophysical systems contain charged objects, it is omnipresent in simulations. For any long-range potential, it is computationally not feasible to calculate direct pair interactions between particles, since without a cut-off and with a periodically replicated system, the summation would not terminate. Therefore the force on each particle is calculated via a global electrostatic potential $\Phi$. It is defined via the Poisson equation

$$\nabla\varepsilon(\mathbf{r})\nabla\Phi = -\rho \quad , \tag{2.4}$$

where $\varepsilon(\mathbf{r})$ is the dielectric permittivity we will explain later, and $\rho$ denotes the distribution of charges in the system. If we regard two point-like particles $p_1$ and $p_2$ with charges $q_1$ and $q_2$ at positions $\mathbf{r}_1$ and $\mathbf{r}_2$, the spherical charge distribution seen by each particle is a delta-distribution $\delta(|\mathbf{r}_2 - \mathbf{r}_1|)$ and the Poisson equation for $\varepsilon(\mathbf{r}) = \varepsilon$ can be solved via Green's function. The electrostatic force acting on $p_1$ is then

$$\mathbf{F}_1 = \frac{q_1 q_2}{\varepsilon(\mathbf{r}_2 - \mathbf{r}_1)^2}\hat{\mathbf{r}} \quad , \tag{2.5}$$

where

$$\hat{\mathbf{r}} = \frac{(\mathbf{r}_2 - \mathbf{r}_1)}{|\mathbf{r}_2 - \mathbf{r}_1|} \tag{2.6}$$

is the connecting vector between the two particles. This force provides two numerical challenges: First, it is a long-range interaction and introducing a cut-off would result in errors that decay slowly. Therefore, we are not merely dealing with pair interactions but have to consider each particles periodic images in periodic boundary conditions. And second, it diverges at short distances $(\mathbf{r}_2 - \mathbf{r}_1) \to 0$. Because of this, it is very common for algorithmic approaches to the solution of the Poisson equation to separate the interaction into a short-range and a long-range part and to tackle both problems individually.

There is a plethora of different methods to solve the partial differential equation for the electrostatic potential, and we will present and compare the most common algorithms in chapter 5. For now, we would only like to mention two more things. First, let us discuss the calculation of actual forces from the resulting electrostatic potential.

The electrostatic Lorentz force on a charged particle is defined as

$$\mathbf{F}_L = q\mathbf{E} \quad , \tag{2.7}$$

where

$$\mathbf{E} = -\nabla\Phi \tag{2.8}$$

is the electric field at the position of the particle, and it is calculated with the gradient of the electrostatic potential, as depicted in figure 2.2

A very convenient way to tackle the differentiation and the periodicity of the potential at the same time is to use Fourier transformations such as

$$\hat{f}(\mathbf{k}) = \frac{1}{(2\pi)^(3/2)} \int e^{-i\mathbf{kx}} f(\mathbf{x}) d^3\mathbf{x} \tag{2.9}$$

and the back-transformation

$$f(\mathbf{x}) = \frac{1}{(2\pi)^(3/2)} \int e^{i\mathbf{kx}} \hat{f}(\mathbf{k}) d^3\mathbf{k} \quad , \tag{2.10}$$

where $f(\mathbf{x})$ is any arbitrary function and $\hat{f}(\mathbf{k})$ is called its *Fourier-transform* in $\mathbf{k}$-space. Without proof here, this transformation has

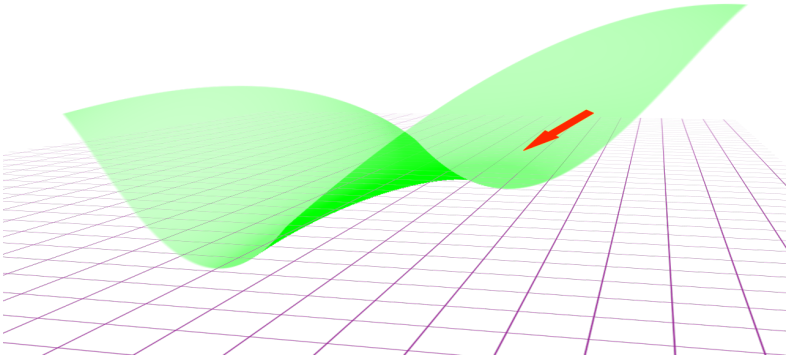


**Figure 2.2.:** The electric field **E** (red arrow) is calculated locally as the gradient of the global electrostatic potential $\Phi$ (green surface). From a visual perspective, it is obvious that only local values of the potential are actually needed to get the field and the resulting electrostatic forces, there is no intrinsic need for the global potential $\Phi$.

two advantages: It assumes a periodicity of the signal automatically, solving the problem of periodic replications. And a differentiation in $\mathbf{k}$-space is merely a multiplication with $i\mathbf{k}$. The P3M algorithm and the Ewald summation both make use of this Fourier transform and will be described later in this thesis.

The second issue we wanted to mention is the difference between a torus geometry and infinite periodic replications of the simulation box. It becomes very apparent when simply placing one single charge into the box, giving the system a net charge in total. In a torus geometry, this is still just one charge in a system bent in space. But if the system is replicated periodically, an infinite system with a constant charge density is created. This results in an infinite amount of energy for the creation of such a system.

In other words: If there is just one single positive charge in the system, where would the field lines originating from this charge lead to? This problem appears only for the monopole- and dipole-moments of the system, since the interactions of higher multipole moments will de-

cay faster and can be considered short-range interactions, their integral does not diverge. Both monopole and dipole moment are therefore corrected in all electrostatics methods and manually forced to be zero. For the monopole moment, one can imagine this in three different ways: The net charge of the system should be zero, so if it is not, an equal amount of opposite charge is homogeneously smeared out over the volume. This will not affect the resulting forces, since it does not create an additional gradient in the potential. It also means that the electrostatic potential is shifted so it yields a boundary condition of $\Phi \equiv 0$ outside the simulation box. This can then be seen at setting *metallic boundary conditions* outside the summation sphere.

We mention this here with only a few handwaving arguments because the above terms, especially "metallic boundary conditions" are mentioned very regularly in literature and are easy to misunderstand. And for our algorithmic extension in chapter 3, it is helpful to have an idea where they originated.

### 2.2.3. Hydrodynamics

The second long-range force present in systems containing a solvent for macromolecules are hydrodynamic interactions, transferred between particles by a solvent such as water. Especially in the limit of an incompressible fluid, they show many similarities to electrostatics, including they same $1/r^2$ decay of the particle pair forces. A key distinction, however, is that hydrodynamic particles do not act as monopoles, or sources, but rather resemble a computational point dipole, with an additional direction determined by their velocity.

Just like electrostatic interactions, hydrodynamic interactions are a computational challenge in coarse-grained Molecular simulations in general. If one simulated explicit water, hydrodynamic interactions would be included, but this is not a viable approach for most macromolecules, since the necessary time scales are too long. A very first attempt at solving hydrodynamic interactions in coarse-grained fashion was *Brownian Dynamics*, following an overdamped Langevin equation [32]. Although further advancements in the field using Fast Fourier transforms improved the asymptotic scaling of the algorithm for $N$ par-

ticles from $\mathcal{O}(N^3)$ to $\mathcal{O}(N \log N)$, it is still computationally very costly, and it is not used for simulations within this thesis.

One major reason for the high computational effort is that the simulation of an explicit solvent contains a very large amount of particles that need to be stored and integrated even though they are not needed for the analysis of most observables and are merely used to transfer hydrodynamic forces. This can be overcome by using one of the more modern hydrodynamic "mesoscopic" algorithms. Here, the solvent is simulated implicitly via a medium that resembles a Newtonian fluid. The two best known methods for such mesoscopic hydrodynamic algorithms are *Dissipative Particle Dynamics* (DPD) and *Lattice-Boltzmann* (LB), both of which are used within this thesis.

The idea of DPD, suggested by Hoogerbrugge and Koelman [64, 82], is to represent several water molecules by one larger soft particle, and to thermalize not individual particles but random particle pairs with opposite velocities, perfectly conserving the total momentum. The limit of an isothermal Newtonian fluid is automatically obtained asymptotically. With this method, the amount of solvent particles is typically reduced by more than an order of magnitude, and it is computationally very effective. A DPD thermostat is implemented in ESPResSo, and it can be used to simulate DPD hydrodynamic interactions by simply adding big soft particles to the system.

The Lattice-Boltzmann method follows a different approach. The idea was originally introduced by McNamara and Zanetti [56]. They followed the approach that, instead of solving the nonlinear Navier-Stokes equation that describes macroscopic flow fields of liquids, the Boltzmann transport equation is used to average the collision between microscopic particles. Unlike the continuum Navier-Stokes equation, discretizing this equation does not lead to the breaking of physical conservation laws, and the solution can be obtained much easier. The fluid is represented by a discrete set of distribution functions living on a regular lattice. Exchange between the lattice sites is done in two steps, a streaming and a collision step. In the streaming step, each particle of the population moves to a neighbor lattice site, according to its velocity direction. And in the collision step, the particle velocities are altered according to a collision model, for example the Bhatna-

gar–Gross–Krook operator [20]. The algorithm has some advantages over conventional methods for Computational Fluid Dynamics, since additional particles and complex boundary conditions can be coupled to it more easily, and the parallelization of the code is very straight forward. We use the GPU-implementation of the Lattice-Boltzmann method in ESPResSo for most of our simulations including hydrodynamic interactions.

## 2.3. Coarse-graining

As mentioned already in the last paragraphs, a key aspect to make Molecular Dynamics simulations computationally feasible is the inclusion of coarse-graining. Depending on the physical problem at hand, some of the interactions can be simplified and several molecules can be merged to allow for larger systems and time scales. The Lattice-Boltzmann method is a very good example where a large and computationally demanding number of explicit water molecules are replaced by a coarser, grid-based method that replicates the important properties of the fluid yet simplifies the numerical solution significantly.

Nowadays, besides the ever increasing power of supercomputer clusters, coarse-graining is successfully used to reach new simulation scales that were not accessible to computer simulations before. Since computer simulations are a very welcome tool to bridge the gap between theoretical prediction and experimental setups, an expansion to reach further into experimental scales is of high interest to the simulational community.

When coarse-graining from quantum mechanical ab-initio simulations to all-atom simulations to Molecular Dynamics simulations to even mesoscopic methods, as depicted in figure 2.3, one has to be particularly careful. Along this way, certain properties of the original system will certainly get lost and the coarse-grained model will always be a simplified and less accurate version. It is therefore an important part of this bottom-up modeling that physicists need to make sure that all relevant interactions and processes to the underlying problem are included in the coarse-grained model. On the next pages, we will give a

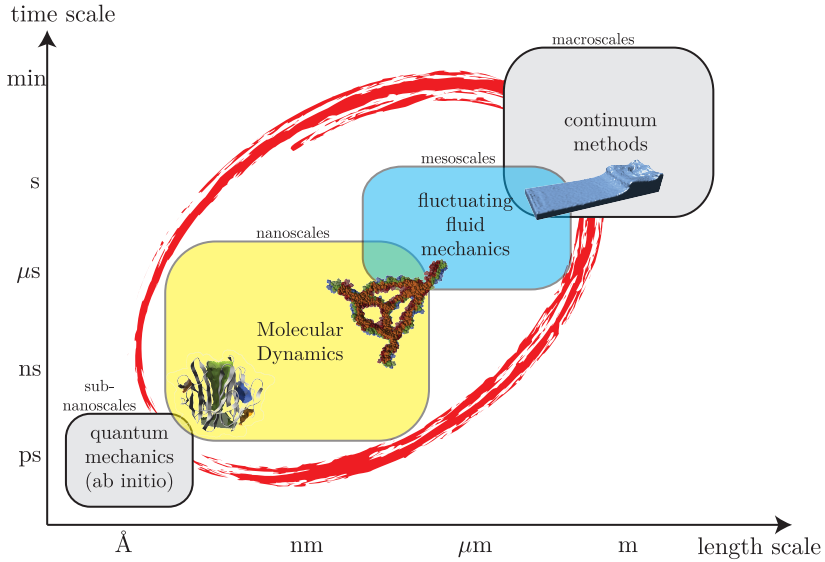



**Figure 2.3.:** Computer simulations span multiple time- and length-scales. Molecular Dynamics simulations are situated in the middle, between engineering approaches with continuum methods and quantum mechanical ab-initio simulations.

short overview on the length scales that different simulational methods are working with today.

### 2.3.1. Atomistic

At the shortest lengths scales of just one or a few atoms, so called *ab-initio* simulations can be executed, actually solving the quantum mechanical equations via perturbation theories to the highest detail possible with our current physical knowledge [22]. Quantum mechanical simulations can be increased to some degree by applying mean field methods like the *Density Functional Theory*, where electrons are represented by a spatially dependent density function [23, 24, 60].

The next step towards coarse-graining are *all-atom* simulations, which do not calculate quantum mechanical wave functions anymore but treat

atoms classically and define force fields to incorporate their interactions on an effective level [51]. This is already considered a Molecular Dynamics simulation, since classical particles interact via potentials and are propagated following the Newtonian equations of motion using an integration scheme.

A good example of a system that has been studied extensively is water [38, 310]. An example for an atomistic model to simulate water would be the TIP4P model [42, 146]. Here, a water molecule is represented by a sphere with a repelling Lennard-Jones potential [14] and 4 point charges inside this sphere, one representing the O-atom, two representing the two H-atoms, and an additional charge to improve the model's reproduction of physical observables. This means that for every water molecule, one short-range interactions and four charges are needed for a proper representation, which is a vast improvement over ab-initio calculations from a computational viewpoint.

### 2.3.2. Macroscopic

At the other end of the spectrum are macroscopic methods (see figure 2.3) that are used for engineering purposes. Numerical methods to tackle those challenges are mostly continuum methods like finite-difference-, finite-element-, or finite-volume-simulations [147]. Again taking water as example, the macroscopic dynamics can be solved with Computational Fluid Dynamics methods on Finite Volume discretization with Reynolds-averaged Navier-Stokes. This will give the desired results for the macroscopic dynamics of flowing water in a realistic time frame, but will entirely disregard any single particles properties of the water, as well as thermal fluctuations.

### 2.3.3. coarse-grained MD

coarse-grained Molecular Dynamics simulations are at the core of this thesis, and are somewhat of a middle ground between quantum theory and macroscopic physics. With some exceptions like the LB method, they are still particle-based and include thermal fluctuations by means of a thermostat. But "coarse-grained" already implies that individual

particles often do not represent single atoms or even molecules, but several atomic groups or molecules are combined into one bead.

The interactions between those beads are based on smaller scaled all-atom simulations, empirical derivations or experiments. Most beads still contain a Lennard-Jones-like spherical repulsion, and very commonly FENE- or harmonic potentials are employed to connect particles with each other. A polymer for example is represented by a chain of beads, each containing an entire section of the polymer representation, connected via bond potentials and Lennard-Jones interactions to maintain an average distance.

If we again look at water as an example, the individual charges of the molecule are discarded and the molecule is often seen as a single neutral particle. To calculate hydrodynamic interactions, we can either simulate the system using explicit water molecules without charges, or we can apply a method like Lattice-Boltzmann that further coarse-grains the particles to a lattice representation. This will still contain all information on viscous Navier-Stokes flow, which is all one needs for many simulations of physical problems.

## 2.4. Dielectric properties

One thing we have not discussed in the last paragraph is that while water molecules do not carry a net charge, they still contribute to the electrostatic interactions because they carry a dipole moment. This is often referred to as the dielectric properties of a material. During the coarse-graining of water via mesoscopic methods like Lattice-Boltzmann of DPD, we have included the hydrodynamic interaction that arise. What has been missing so far is the influence that water has on electrostatic interactions.

### 2.4.1. Permittivity through polarization

As stated before, each water molecule, although electrically neutral overall, resembles a point dipole from a distance. If an electric field is applied to a collection of these water molecules, they are restricted in their rotational degree of freedom, as depicted in figure 2.4.

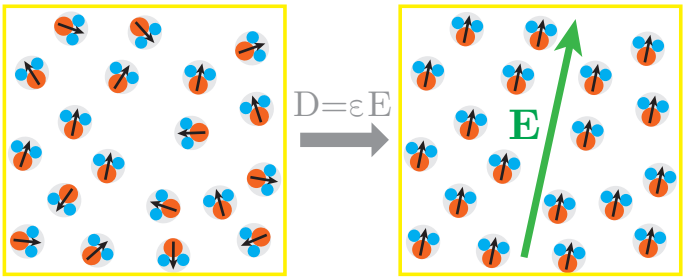



**Figure 2.4.:** Each water molecule carries a small dipole moment. If an electric field is applied, the dipole will orient the opposite direction to minimize their electrostatic energy. This polarization will create a field in the opposite direction and weaken the **E**-field and therefore the electrostatic interactions in the system. This weakening of the field can be expressed via a prefactor $\varepsilon$. Of course, this sketch is oversimplified and the dipoles will not strictly line up as shown here.

The sketch is obviously oversimplified, since the dipoles will have thermal energy and will not align like depicted here but rather exhibit a biased rotational degree of freedom with a tendency to orient in a way opposing the underlying electric field to minimize their electrostatic energy. Averaged over many water molecules, this will create a polarization density **P** created by the applied electric field **E**. In a linear, homogenous, and isotropic dielectric medium, that instantly responds to any changes, this relation can be considered linear,

$$\mathbf{P} = \chi \mathbf{E}. \tag{2.11}$$

We thus define a *displacement field*

$$\mathbf{D} = \mathbf{E} + \mathbf{P} = (1 + \chi)\mathbf{E} := \varepsilon \mathbf{E} \quad , \tag{2.12}$$

where $\varepsilon = (1 + \chi)$ is called the relative *dielectric permittivity*, a dimensionless value assigned to the material.

With the assumptions of instantaneous, linear response and homogeneity, this bulk permittivity can be treated as nothing but a prefactor which linearly weakens the electrostatic field and therefore any electrostatic interactions. This is typically how aqueous solutions in

biophysical systems are treated and how the dipole moment of water molecules can be carried through the coarse-graining process of Molecular Dynamics simulations.

### 2.4.2. Influences

The assumption that the electrostatic dipoles of the water molecules can react instantly is mostly true for our systems, but a strong frequency dependence of the permittivity can be observed in water if high frequency alternating fields are applied, as shown in figure 2.5a.

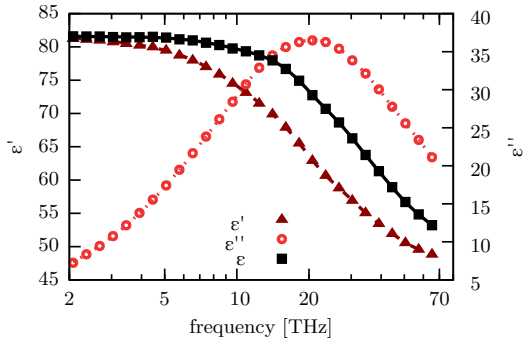
Here, the real and imaginary party of the permittivity $\varepsilon = \varepsilon' + i\varepsilon''$ are related to the stored energy and the dissipation of energy within the medium respectively. The peak in $\varepsilon''$ in figure 2.5a thus marks the frequency at which the most energy is absorbed into the medium, which in the case of water is used by commercial microwaves for heating purposes.

The dielectric permittivity of water is also dependent on the temperature, as shown in figure 2.5b. Other than that, we have made two more assumptions about the bulk dielectric permittivity: (a) The polarization of the material is linear to the applied field, which we will discuss in more detail in the next section, and (b) it is homogenous. This is obviously not true for many biophysical systems. They can not only include dielectric enclosures like large colloids, nanopores, membranes, or walls. The salt concentration also has a major influence on the dielectric permittivity, and it can change spatially and with time.

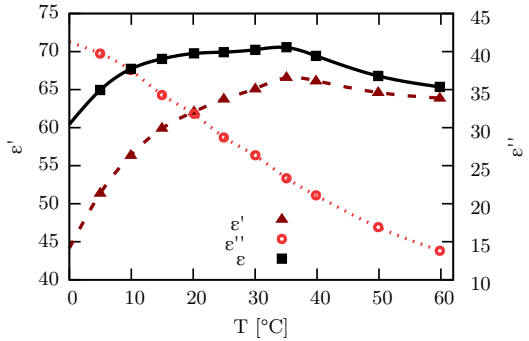
There exist a few methods to include changes in the dielectric constant within a Molecular Dynamics simulation. We will present some of them in section 2.5, and then introduce an entirely new way in chapter 3.

## 2.5. Numerical methods for dielectric enclosures

The results in this section have been published in [3], and some of the formulations are identical.




**(a)** frequency dependence




**(b)** temperature dependence

**Figure 2.5.:** The data has been measured experimentally by U. Kaatze [59]. (a) Dependence of the complex dielectric permittivity of water on the frequency of the applied field. (b) Dependence of the complex permittivity of water on the temperature. The absolute value, or relative permittivity, is added in black for convenience.

### 2.5.1. ICMMM2D

We start by discussing the simplest case of dielectric interfaces, namely two planar, parallel interfaces that enclose a set of charges. We assume a vertical orientation of the interfaces, and refer to a left and a right (l/r) interface. The electric field between the two interfaces can be computed from these charges plus additional image charges outside

of the dielectric boundaries [110]. The positions and magnitude of these images charges are chosen to satisfy the boundary conditions for the electric field. If only one interface, the left or the right interface, were present, image charges would appear reflected at the respective interface, with the charge scaled down by a factor of

$$\Delta_l = \frac{\varepsilon_m - \varepsilon_l}{\varepsilon_m + \varepsilon_l}, \qquad \text{and} \qquad \Delta_r = \frac{\varepsilon_m - \varepsilon_r}{\varepsilon_m + \varepsilon_r} \tag{2.13}$$

where $\varepsilon_m$ is the background permittivity in the main cell, and $\varepsilon_l$ and $\varepsilon_r$ are the permittivities in the adjacent left and right media, respectively. Note that $\Delta_l$ and $\Delta_r$ can be positive as well as negative, so that a charge can be attracted to the image charge or repelled by it.

To construct the image charges in a situation with two interfaces, every image charge created by reflection on one interface also needs to be reflected again onto the other interface. This leads to an infinite set of images: A charge $q$ is reflected at the right (left) interface and yields an image of magnitude $q\Delta_r$ (resp. $q\Delta_l$). The next reflection gives rise to another image charge $q\Delta_l\Delta_r$ (resp. $q\Delta_r\Delta_l$) in the opposite dielectric domain, and so on. The infinite array of image charges is depicted in figure 2.6: a charge $q$ at position $z$ will produce a series of mirror charges in the right dielectric domain (with $\varepsilon_r$) with charges

$$\begin{aligned} q\Delta^{n+1} &\text{ at positions } -2(n+1)L_z + z, \quad \text{and} \\ q\Delta_r\Delta^n &\text{ at positions } -2nL_z - z,\ n \geq 0, \end{aligned} \tag{2.14}$$

where $L_z$ denotes the distance between the two interfaces and $\Delta := \Delta_r\Delta_l$. In the left dielectric domain (with $\varepsilon_l$), the charges are

$$\begin{aligned} q\Delta^{n+1} &\text{ at positions } 2(n+1)L_z + z, \quad \text{and} \\ q\Delta_l\Delta^n &\text{ at positions } 2(n+1)L_z - z,\ n \geq 0. \end{aligned} \tag{2.15}$$

When computing the electrostatic interaction in a computer simulation with such parallel, infinite planar walls, it is often desired to employ periodic boundary conditions in the two directions parallel to the walls to minimize surface effects. The direct summation of periodic replicas is very costly as the sum is only slowly convergent. Thus special techniques to compute the electrostatic interactions are required.

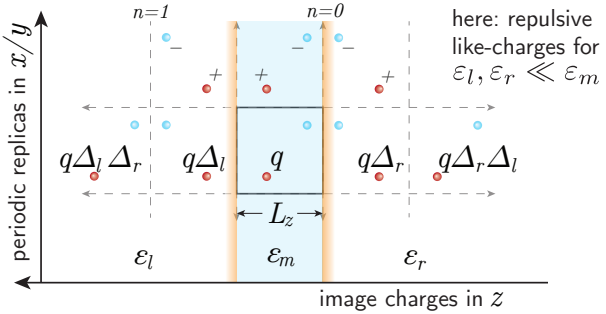



**Figure 2.6.:** Schematic summation scheme for ICMMM2D: in order to take into account dielectric boundaries, image charges are introduced outside the dielectric boundaries, to the left and right of the original box. The dielectric contrasts $\Delta_l$ and $\Delta_r$ are computed from the dielectric jump at the left and right boundaries, respectively. Depending on the dielectric contrasts, charges will either be repelled by the surface (as in this sketch) or attracted to it. Note that in usual computer simulations, the system is periodically replicated in the dimensions parallel to the interfaces, as indicated in the figure. The sketch has been published in [3].

MMM2D is such an algorithm, that computes the electrostatic interaction with two periodic dimensions [125, 126], and is well suited to compute the interactions with the image charges. The key idea of the MMM2D algorithm is to use two different formulas for the interaction of two charges. One of the formulas, the near formula, is only used if the two particles are sufficiently close. In combination with image charges it is only used for the closest of all images and its discussion is beyond the scope of this thesis.

The second, far formula is accurate if a certain distance between the two charges is exceeded. We assume a simulation box of $L \times L \times L_z$ that is periodically replicated in the $x$ and $y$ dimensions. Then the Coulomb potential of a unit charge placed at the origin evaluated at position $(x, y, z)$ with $|z| > 0$ including periodic replicas in $x$ and $y$ direction can be written as:

$$\Phi(x,y,z) = \frac{1}{4\pi\varepsilon L^2} \sum_{p,q\neq 0} \frac{e^{-2\pi f_{pq}|z|}}{f_{pq}} e^{(2\pi px + 2\pi qy)/L} + \frac{1}{2\varepsilon L^2}|z|, \quad (2.16)$$

where $f_{pq} := \sqrt{p^2+q^2}/L$. This formula follows from a Fourier transformation of the Poisson equation in $x$ and $y$ direction, and can be factorized into contributions. This makes it possible to compute the interactions between $N$ separated charges in $\mathcal{O}(N)$ operations. Due to the unfavorable scaling of the near formula MMM2D scales only like $\mathcal{O}(N^{5/3})$ overall. It however is still superior to Ewald based methods [99] in partially periodic boundary conditions.

Coming back to our original problem of taking into account the infinite array of image charges, we note that all these charges, with the exception of those directly adjacent, i. e. with index $n \geq 1$, are far from the slab containing the real charges. Therefore, we can apply the fast far formula in order to compute the interaction with these charges. The infinite sums of image charges lead to geometric series of the form

$$\sum_{n=0}^{\infty} \Delta^n \frac{1}{4\pi\varepsilon_m L^2} \sum_{p,q\neq 0} \frac{e^{-2\pi f_{pq}(2nL_z \pm z)}}{f_{pq}} e^{(2\pi px + 2\pi qy)/L} \tag{2.17}$$

which is a geometric series that can be easily simplified to

$$\frac{1}{4\pi\varepsilon_m L^2} \sum_{p,q\neq 0} \frac{e^{\pm 2\pi f_{pq} z}}{f_{pq}(1-\Delta e^{4\pi f_{pq} L_z})} e^{(2\pi px + 2\pi qy)/L}. \tag{2.18}$$

In other words, an existing implementation of MMM2D can be easily enhanced in order to include dielectric interfaces, simply by modifying the prefactors of the $p,q$-Fourier sum. For the detailed expressions, see [201]. In the following, we denote such an MMM2D implementation for dielectric interfaces as ICMMM2D (image charge MMM2D).

Note that (2.16) contains an additional term $|z|/(2\varepsilon_m L^2)$, which represents the constant Fourier mode. The summation over the image charges of this term leads to expressions of the form

$$\frac{1}{2\varepsilon_m L^2} \sum_{n\geq 0} \Delta^n \left(2nL_z \pm z\right), \tag{2.19}$$

since $2nL_z$ is larger than any possible particle distance $z$. Unlike (2.17), this is not a simple geometric series. However, when computing the total potential in a charge neutral system, terms that do not depend on

the positions cancel out, in particular the $2nL_z$ terms. What remains from the four image charge sums are again geometric series, that lead to a contribution of

$$\frac{1}{2\varepsilon_m L^2}\frac{\Delta_r - \Delta_l}{1-\Delta}z. \tag{2.20}$$

If the two planar walls have the same dielectric contrast, this contribution to the potential vanishes. Also, if one is only interested in energy or forces, this contribution vanishes due to charge neutrality.

An alternative method for planar dielectric interfaces is based on an extension of the ELC method. It also uses the technique to sum up the image charges with the ICMM2D far formula, and is termed ELCIC (ELC with image charges), see [224] for details of the method.

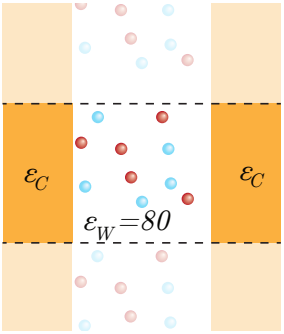
Note that it is also possible to consider systems that are not charge neutral. Formally, one assumes two equally charged plates at the positions of the dielectric interfaces that cancel the total charge [225]. The field of a charged plate is, however, exactly what the $|z|/(2\varepsilon_m L^2)$ term represents, so that the above considerations for this term still hold. Therefore, one can safely ignore this contribution.

### 2.5.2. Example: electrolyte between dielectric walls

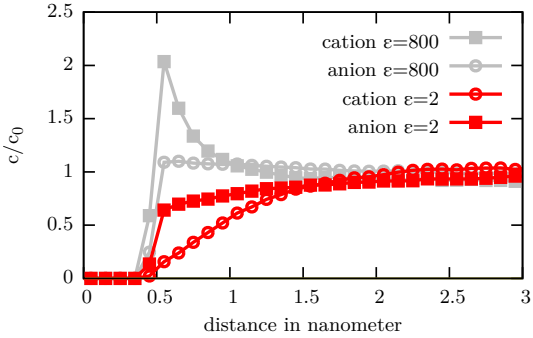
As an example application of the ICMMM2D algorithm, we simulated an 3:1 electrolyte (e. g. $AlCl_3$) with a concentration of 0.01 mol/l confined by uncharged planar walls to a slab as depicted in figure 2.8a. The size difference between positive and negative ions has been neglected here. This is a good model for the narrow slit between the electrodes of a capacitor as well as two biological membranes or glass plates. However, the dielectric properties of a metal electrode (here approximated with $\varepsilon_C \approx 800$) and a biological membrane ($\varepsilon_C \approx 2$) are very different, and in both cases also differ strongly from the permittivity of the solvent, here water ($\varepsilon_W = 80$). The resulting dielectric jumps at the surfaces have a pronounced effect on the distribution of ions near the walls.

Figure 2.8b shows this strong influence of the dielectric interfaces. Both cations and anions are attracted to the walls of of high permittivity ($\varepsilon_C = 800$), but repelled by the low permittivity ($\varepsilon_C = 2$) walls.

**Figure 2.7.:** (a) Sketch of a simulation setup, with a 3:1 electrolyte, e. g. $AlCl_3$, at 100 mM concentration between two uncharged walls with a dielectric constant different from that of water. The size difference between the ion types is neglected in this simulation. The slab is periodically replicated parallel to the walls, vertical in the sketch.
(b) Density distribution of anions and cations of the trivalent electrolyte, near the dielectric interface. The dielectric interfaces is placed at $x = 0$, and a repulsive potential maintains a minimum distance of 0.5 nanometers of all ions. A good dielectric $\varepsilon_C = 800$, representing conducting material, strongly attracts cations, while anions are less attracted. A bad dielectric $\varepsilon_C = 2$, representing a typical biological membrane, repels cations. The univalent anions are much less repelled. These graphs have been published in [3].




**(a)** Electrolyte schematic




**(b)** Ion density distribution

On a microscopic level, the electric field of a charge will give rise to a dielectric displacement in the wall. This displacement will weaken or pronounce the field within the dielectric medium, compared to the other side of the interface. It can be accounted for by imagining virtual mirror charges or surface charges directly on the interface. To correctly reproduce the field discontinuities at the boundary, these charges will be of attractive nature in the region of lower permittivity and of repulsive nature in the region of higher permittivity. Note also that the effect is more pronounced for multivalent ions. Ignoring the dielectric jumps would lead to a much more homogeneous charge distribution, so that one would strongly underestimate the effect of including multivalent

ions.

### 2.5.3. Induced Charge Calculation

The concept of induced charges rather than image charges is a direct route to take into account dielectric interfaces of arbitrary shape. Conventionally charge induction is considered to be the origin of Faraday's cage effect: applying an electric field to a conductor will trigger the mobile charges inside to move until the electric field vanishes inside and field lines end orthogonally to the surface of induced charges. The same concept, however, can also be applied to dielectrics, hence materials with immobile charges. This can be seen from the following mathematical consideration. The Poisson equation in an inhomogenous dielectric medium (3.13) can be rewritten as:

$$\Delta\Phi = -\frac{1}{\varepsilon}\rho - \frac{1}{\varepsilon}\nabla\varepsilon \cdot \nabla\Phi \tag{2.21}$$

The term $\nabla\varepsilon \cdot \nabla\Phi$ is identified as the *induced surface charge density* $\sigma$. It is nonzero only on the dielectric interfaces since $\nabla\varepsilon$ vanishes everywhere else.

Let us introduce the Green's function $G$ for the Laplace operator. It can e. g. be just $\frac{1}{4\pi|r-r'|}$, but may also include the desired periodicity. Then it is possible to eliminate the Laplace operator from the equation above and express the potential by means of two integrals:

$$\Phi = \int_V G\left(r, r'\right) \rho\left(r'\right) / \varepsilon \, dV' + \int_A G\left(r, r'\right) \sigma\left(r'\right) / \varepsilon \, dA'. \tag{2.22}$$

The volume integral extends over medium 1 and the surface integral extends over all dielectric interfaces. The potential is now expressed in terms of the Green's function of a homogeneous dielectric, yet the induced charge density $\sigma$ is still unknown. We now assume that the charges are embedded in a medium with permittivity $\varepsilon_1$ and for simplicity only a second permittivity $\varepsilon_2$. By taking the gradient and inserting this expression in the definition of the induced charge density,

we obtain the following integral equation:

$$\sigma = 2\frac{\varepsilon_1 - \varepsilon_2}{\varepsilon_1 + \varepsilon_2}\left(\int_V \nabla_{\mathbf{r}} G\left(r, r'\right)\rho\left(r'\right)/\varepsilon_1 V' + \int_A \nabla_{\mathbf{r}} G\left(r, r'\right)\sigma\left(r'\right)/\varepsilon_1 A'\right). \tag{2.23}$$

This result is easily generalized to multiple regions with different permittivities The idea of the ICC$\star$ algorithm [255] is to determine this charge density self-consistently, after discretizing the surface.

In principle this approach is a boundary element method, an approach that is very widely used, e. g. for low Reynold number flow[130]. Different from other approaches is, however, that the efficient evaluation of the Green's function can be borrowed from standard Coulomb solvers. This can be seen from the following: assuming a discretized surface of $m$ point charges on the dielectric interface the equation above for discretization point $k$ can be written as

$$q_k = A_k\frac{\varepsilon_1 - \varepsilon_2}{\varepsilon_1 + \varepsilon_2}\mathbf{n}_k \cdot \left[\sum_{i=1}^{n} q_i \nabla_{r_k} G\left(\mathbf{r}_k, \mathbf{r}_i\right) + \sum_{j=1, j\neq k}^{m} q_j \nabla_{r_k} G\left(\mathbf{r}_k, \mathbf{r}_j\right)\right],$$

where $A_k$ is the surface area of the surface element $k$. The term in square brackets is just the electric field acting at the position of point $k$ assuming a homogeneous dielectric constant $\varepsilon_1$ in the system, created by conventional (not induced) charges. Any standard Coulomb solver can thus be used to perform the calculation. The desired solution of all $q_k$ is the fix point of the following iteration, which is a successive overrelaxation scheme:

$$q_k^{l+1} = (1-\omega)\, q_k^l + \omega A_k \frac{\varepsilon_1 - \varepsilon_2}{\varepsilon_1 + \varepsilon_2}\mathbf{n}_k \cdot \mathbf{E}\left(\left[q_i\right], \left[q_j^l\right]\right).$$

It turned out that this iteration is very stable and with a choice of $\omega \approx 0.7$ no stability issues occur. In every MD step only 1-3 iterations are necessary as the particle positions change only slightly.

An important advantage of this algorithm is that the computationally most costly part, the evaluation of the electric field, can be done with *any* usual electrostatics solver without modifications. Thus not only the computational efficiency, but also the periodicity is inherited from the Coulomb solver. The complexity of the algorithm remains unchanged by the presence of induced surface charges. However, the number of particles can increase considerably. We found it sufficient to discretize the surface with mutual particle distances equal to the distance of closest approach. For the system shown in section 2.5.1 this would mean in total 1600 surface charges per wall, compared to less than 100 ions in the system.

In the research performed at the Institute for Computational Physics, this algorithm was applied to investigate if dielectric effects can change the electrolytic conductance of very narrow pores, nanopores, through membranes, or have an influence on the translocation of charged macromolecules through nanopores [265, 283]. Here, dielectric boundary forces lead to a repulsion of (unpaired) ions. Taking into account dielectric effects in small pores will decrease the number of available ions, and thus decrease the conductance.

The error in the obtained electric field depends on the applied resolution with which the surface is resolved. The method has been tested for planar and curved surfaces [255], and it was found that from a distance larger than one lattice spacing the relative error remains smaller than 1%. Since the permissible error depends often on the desired application we advise to determine the necessary accuracy specifically for each case. If interfaces to media with a high dielectric constant, or even metallic boundaries are considered, charges are attracted to the surface and can come quite close to the interface, depending on the ion size. In this case, the necessary accuracy is clearly higher than for interfaces to lower dielectric media, from which particles are repelled.

### 2.5.4. Example: ion distribution in a pore

As an example we investigate the ion distribution in a cylindrical pore of radius 5nm in a 40nm thick membrane in an aqueous electrolyte. This geometry resembles the so-called solid state nanopores [170, 185,

191] in silicon wafers. In several experiments (e. g. [178, 238]), it was shown that single DNA molecules present in such a pore can be detected by the change of the electric conductance of such a system. To make the dielectric effect more pronounced we again used a 3:1 electrolyte with a concentration of 10mmol/l. Our setup is sketched in figure 2.9a: the surface charges of the ICC$\star$ algorithm are displayed along with the ions, each as spheres. We assume the dielectric constant of the membrane to be $\varepsilon_P = 2$. In figure 2.9b the equilibrium distribution of ions near the center of the pore is shown. Ions, especially the trivalent ones, are repelled from dielectric interface. This leads to an overall decrease of ions in the pore by around 20%. Thus the conductance of the system can be expected to be reduced similarly, compared to a model which does not consider the dielectric contrast.

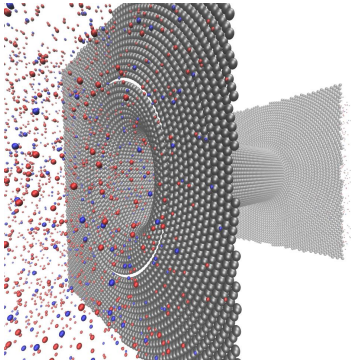
### 2.5.5. Iterative Poisson-Boltzmann Solver

In addition to the above explained algorithms to calculate electrostatic forces in systems with dielectric variations, we want to briefly present an entirely different approach, since it is used in a small section of our simulational part.

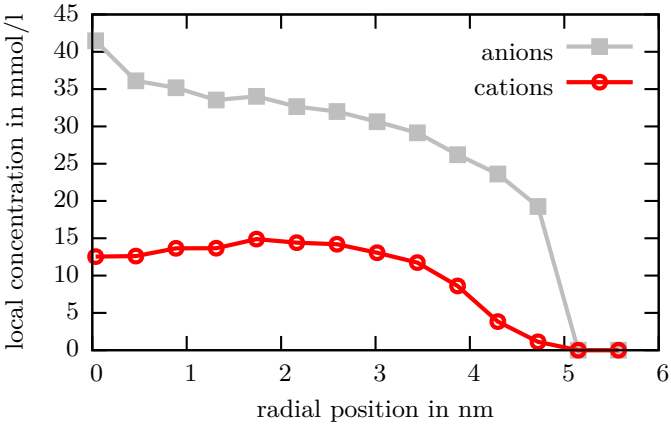
Most systems present in biophysics contain water that is not pure but has a finite salt concentration. If a charged object is then placed in aqueous solution, its solvated counterions as well as the salt ions of opposite sign will be electrostatically attracted to the object and assemble around it. The field created by the charged object is significantly weakened, and the counterions provide a "screening effect". At high salt concentrations, as they are sometimes present in experimental systems, this can lead to the charged object being entirely neutralized at just a few nanometers distance. In such systems, the long-range nature of electrostatic interactions is essentially completely screened and can be neglected.

The assembly of ions close to a charged object is determined by the attractive electrostatic energy and the entropic tendency of the ions to spread homogeneously in the system. This interplay was first theoretically described by Gouy and Chapman [9, 10], treated in a linearized fashion by Debye and Hückel [13] and is usually called *Poisson-*

**Figure 2.8.:** (a) The ICC⋆ example system: positive and negative ions are displayed as red and blue spheres, the ICC⋆ discretization points by grey spheres. (b) Ion density of both species in the pore measured close to the center of the pore. These graphs have been published in [3].



**(a)** ICC⋆ example nanopore




**(b)** Radial distribution of ions in the pore

*Boltzmann* (PB) theory. The theory assumes that the ions exhibit an ideal gas chemical potential, and that the electrostatic interaction can be described by a mean density field, not explicit charges. For multivalent ions, this latter assumption often breaks down, but in systems involving only monovalent ions in an aqueous environment, the Poisson-Boltzmann theory is a very good approximation and has been successfully used for many years. Conveniently, the theoretical description can be solved numerically for many systems, and if the systems display a symmetry so that the solution of the ion distribution can be regarded as a one-dimensional problem, it can even be solved analytically.

In his diploma thesis [270], A. Schlaich extended the PB algorithm even further to solve the equilibrium ion distribution and electrostatic potential for sharp dielectric interfaces of arbitrary shape. The numerical solution was hereby obtained using the "Distributed and Unified Numerics Environment" (DUNE) and the mesh generating interface PDELab [172, 205, 313].

Note that this algorithm is not meant for dynamic simulations but rather to calculate static equilibrium properties of a system, using mean-field theory approximation.

## 2.6. Computational Considerations

Numerical integration and derivation, as well as many fast numerical algorithms do not work in continuous space and time. In many computer simulations, space as well as time is therefore often discretized. This of course leads to numerical errors that need to be considered and kept at a minimum. Here, we will present the discretization used within this thesis, and give some thoughts on parallelization of different algorithms.

### 2.6.1. Discretization

In Molecular Dynamics simulations, time discretization is fairly self-explanatory. The Newtonian equations of motion are integrated numerically, and instead of a continuous time propagation, we apply a

second order numerical integration scheme. The particles, forces, velocities, and fields are updates step-wise at a time interval $\delta t$, and the numerical scheme provides a solution that is correct to the order $\delta t^2$, which is sufficient for Molecular Dynamics simulations. The numerical error is far smaller than the heat bath coupling via thermalization.

Spatial discretization, however, is not required from an algorithmic viewpoint. The particles in Molecular Dynamic simulations, apart from the finite machine precision of any computer program, move in continuous space. But many algorithms, particularly for the solution of long-range interactions, are designed to work on a lattice, in discretized space. We have already presented the Lattice-Boltzmann method, which solves an approximation of Navier-Stokes flow on a lattice and can couple to off-lattice particles. We have also mentioned the P3M algorithm frequently. It makes use of the Fast Fourier Transform (FFT) for transformation of the charge distribution into **k**-space, and this $\mathcal{O}(N \log N)$ algorithm will only work with equidistant supporting points, therefore discretized in space. Finite-Difference-, Finite-Element-, and Finite-Volume-methods are also implemented on a lattice, as are almost all other electrostatics methods we will introduce in this thesis, except for our slow reference implementation of the standard Ewald summation.

This means that some of the off-lattice particle properties, namely the charge for electrostatics and the velocity for hydrodynamics, have to be interpolated onto a regular lattice. While there are many different interpolation schemes available for this task, two main possibilities stick out. The first and simplest method is a linear interpolation to the eight nearest lattice points. It has the important advantages of being very fast and being completely local. This is not only computationally advantageous, but plays a key role if local variation of system properties have to be applied, as we will see in chapter 3. The second method is a B-spline interpolation, with piece-wise polynomial functions to interpolate the charge distribution onto supporting points. The degree of these polynomials, and with it the spread of the interpolation, is chosen to match the circumstances of the numerical application. The advantages of B-splines are that they are differentiable in all points, that is has – in contrast to a Gaussian interpolation – finite limits in space,

and that their Fourier transform is again finite. The disadvantages are that it requires a lot of computational effort – in typical Molecular Dynamics simulations with P3M the interpolation uses about half of the resources for the entire solution of the Poisson equation – and that it does not allow for local property changes of the underlying system.

### 2.6.2. Parallelization

In the last decade, the clock performance of single cores did not follow the famous Moore law anymore, in fact, it has almost stagnated. The overall performance of processors, however, has kept increasing with the use of multiple cores for each CPU (Central Processing Unit). The same trend is visible in High Performance Computing (HPC), where record-breaking supercomputers as well as smaller clusters are steadily increasing the number of compute nodes and improving the network structure between those nodes. Today, to reach relevant system sizes and simulation times for the study of biophysical systems, the parallelization of a simulation package has become more important than ever.

In the field of Molecular Dynamics, the system is usually split between several processors by means of domain decomposition, as depicted in figure 2.1. This is very well suited for short-range interactions and other local algorithms, where information about particles from other processors is only needed at the edges of a domain. For long-range interactions, there are three different approaches: Either the system is not split into spatial domain but follows a different decomposition scheme, as it is done for tree-methods like the Fast Multipole Methods, but this is only applicable if there are no or very few other interactions in the system that would greatly benefit from a domain decomposition. Or second, the system has to globally exchange information on all particle positions at every time step. While at first glance, this seems a lot of effort, we saw that with a performant implementation, it is still feasible up to a very high number of processors, and we will present such benchmarks in section 5.3.4. Or third, a *local* algorithm is used for the long-range interactions. A very good example of this is Lattice-Boltzmann, which recovers the complete long-range

hydrodynamic interaction with purely local numerical updates. Such an algorithm for electrostatic interactions will be presented in chapter 3.

It is obvious that if the communication between compute nodes in a massively parallel simulation takes a constant amount of time, the simulation will scale much better up to high CPU numbers if the calculations between communications steps take a long time. It is therefore beneficial in Molecular Dynamics simulations to decompose only to a level where there are still around $10^3$ particles left on each node. If this number becomes too small, the scaling of the implementation will suffer.

# 3. Maxwell Equations Molecular Dynamics (MEMD)

In this chapter, we present a new mathematical approach to the MEMD algorithm which is not based on the Maxwell equations but originates from the Poisson equation (3.13) and can easily include arbitrary dielectric properties in all proofs. We show that with very few physically motivated assumptions, the Maxwell equations of electrodynamics follow naturally from the Poisson equation.

We will then explain some of the problems that arise from the algorithm in general, the numerical discretization, and the extension to spatially varying dielectric permittivity in particular, and how we dealt with those problems. We present an error estimate and examine the strengths and weaknesses of the algorithm and possible applications.

Finally, a large portion of this chapter features proofs of the mathematical integrity of the algorithm and demonstrations of the fact that it results in correct static observables.

The content of this chapter was reviewed and published in [4], and similarities in the wording can occur.

## 3.1. Extension of the original algorithm

The idea for the original "Maggswellian Dynamics" algorithm is strongly based on Car-Parrinello Molecular Dynamics (CPMD), which was introduced in 1985 [47]. The authors had to face the challenge of simulating metallic systems, in which the relevant processes of nuclei motion, calculated with Molecular Dynamics (MD) simulations, and electron wave function variation, calculated via Density-Functional Theory (DFT), occur on vastly different time scales, which is very demanding regarding the required computational effort.

The approach of Car and Parrinello was to target the rate at which the DFT wave functions are updated in comparison to the much slower occurring MD steps. They were able to prove that one can enhance the update speed of the wave function to a highly unrealistic degree, as long as the time scale remains separated by an order of magnitude in comparison to the MD steps. The forces, and therefore the motion of the nuclei, will still be correct. This essentially leads to a system where the dynamics of the electron wave functions is not realistic and can in fact be considered an artificial dynamics, but the dynamics of the nuclei, and therefore the statistic observables relevant within the system, will be kept intact.

In 2002, A. Maggs introduced a new algorithm [133], where the same concept is applied to the numerical evaluation of the electrostatic interactions in a periodic system. As stated in section 2.2, long-range interactions like electrostatic interactions are a key component in Molecular Dynamics simulations, and to this day are one of the calculations requiring the most computational effort. And since electrostatics is nothing but the limit of electrodynamics where the speed of light approaches infinity, this concept is interesting and applicable to the force calculation in Molecular Dynamics simulations. A comparison of the various components of the two algorithms is shown in table 3.1. For readability, we refer to the "Maggswellian Dynamics" algorithm as MEMD for the remainder of this section, since the original Monte Carlo algorithm was not given an official name by A. Maggs and the same arguments and concepts apply to the MEMD algorithm which we will mostly use throughout this thesis.

The equivalent of nuclei and wave functions in such an algorithm are the charges moving through the system in small MD steps and the electromagnetic fields carrying the electrodynamic interactions, as presented in table 3.1. The same way the update frequency of the electron wave functions is reduced in CPMD, the update frequency and with it the propagation speed of electromagnetic fields is reduced in MEMD, leading to artificial dynamics for the wave functions, respectively the electromagnetic fields. In CPMD, the update frequency of the wave functions is reduced by artificially increasing the electron mass, and in MEMD a prefactor $f_{\text{mass}}$ is introduced in a similar manner. This

| CPMD | | MEMD |
|---|---|---|
| nuclei | $\leftrightarrow$ | charges |
| electron wave function | $\leftrightarrow$ | fields |
| artificial dynamics for $\Psi$ | $\leftrightarrow$ | artificial dynamics for $\mathbf{D}$ and $\mathbf{B}$ |
| electron mass | $\leftrightarrow$ | $f_{\text{mass}} = 1/c^2$ |

**Table 3.1.:** Parallels between Car-Parrinello Molecular Dynamics (CPMD) and the Maxwell Equations Molecular Dynamics (MEMD) algorithm. The concepts and proofs are quite similar: The dynamics of charges in MEMD and ion cores in CPMD remain correct and lead to valid results for the relevant statistical observables. But the dynamics of the interactions, wave functions and electromagnetic fields, are entirely artificial. Here, $\Psi$ denotes the electron wave function and $\mathbf{D}$ and $\mathbf{B}$ are the electric displacement field and the magnetic field, respectively. $f_{\text{mass}}$ is a dampening factor that is directly related to the field propagation speed $c$ as explained later in section 3.1.2.

prefactor can be directly linked to a reduction of the propagation speed $c$ of the electromagnetic fields, which for realistic field dynamics would be the speed of light.

Some specialties that differ from the CPMD formulation are: For numerical reasons, the motion of charges and especially the propagation of electromagnetic fields is calculated on a lattice via space discretization. This yields some additional problems as will be shown in section 3.2. The algorithm can become unstable and its numerical stability is coupled to the lattice spacing and wave propagation speed as also presented in section 3.2. And finally, the periodicity of the simulation box has to be addressed. In a straight forward way, the propagation of particles and electromagnetic fields alike can attach one side of the simulation box to another and create a three-dimensional torus geometry, or true periodic boundary conditions.

In the following sections, we shortly describe the original Monte Carlo algorithm as introduced by A. Maggs in 2002, and its adapta-

tion to a system with wave-like field propagation, now coined "Maxwell Equations Molecular Dynamics". We will then point out and solve the problems that arise from its application. The algorithm is then extended to feature spatially and temporally varying dielectric properties. We will prove the integrity of the algorithm in a novel and very basic way, and finally we will present some additional considerations concerning our implementation within the ESPResSo and ScaFaCoS software packages.

### 3.1.1. Monte Carlo algorithm

The original algorithm as introduced by A. Maggs, occasionally referred to as "Maggswellian Dynamics" is based on Monte Carlo moves on a lattice. A more thorough introduction to the Monte Carlo (MC) update scheme can be found in [127], and we will only sketch the general idea here. In this scheme, a system is propagated step-wise in the following manner: in every step, a particle is picked at random and moved in a randomized way. The total energy $\epsilon_{\text{new}}$ of this new configuration is calculated and compared to the energy $\epsilon_{\text{old}}$ before the move. If the energy difference $\Delta\epsilon = \epsilon_{\text{new}} - \epsilon_{\text{old}}$ is smaller or equal to zero, i.e. if the new position is energetically advantageous, the move is accepted and the system is further propagated from its new state. If the energy difference $\Delta\epsilon$ is positive, the move is only accepted with a probability $p_A$, the most famous and common of which is the Metropolis criterion with

$$p_A = \exp\left(-\frac{\Delta\epsilon}{k_B T}\right), \tag{3.1}$$

where $k_B$ is the Boltzmann constant and $T$ is the temperature of the system.

A. Maggs applied this general concept to the particle motion within the system. In his first publications on the topic [132, 133], charged particles were restricted to move individually on a lattice in a classical Monte Carlo manner. In addition, as depicted in this chapter's introduction, the electrostatic interactions between charges were calculated

with a modified electrodynamics algorithm, which was also propagated on the aforementioned lattice.

It was shown by A. Maggs in his publications that the system, including the electric fields, would reach the equilibrium state and therefore correct retarded solutions to the Maxwell equations if the electromagnetic fields satisfy the boundary condition

$$\nabla \mathbf{E} = \frac{\rho}{\varepsilon_0}, \tag{3.2}$$

where $\mathbf{E}$ is the electric field, $\rho$ is the total electric charge density, and $\varepsilon_0$ is the electric constant. We will give a different proof for this relation in section 3.2 that is better suited to our adaptation.

For the algorithms to produce correct results, two things were needed: A way to couple the moving charges to magnetic fields and a way to propagate those fields within the system while obeying the Gauss law (3.2). The former can be achieved by simply adjusting the electric field on the connecting lattice site in case a particle moves between two adjacent lattice vortices via

$$E_{\text{new}} = E_{\text{old}} - \frac{q_p}{\varepsilon_0}, \tag{3.3}$$

where $q_p$ is the particle charge. With this adjustment, the constraints on the electric fields are still satisfied.

The latter can be achieved in different ways, and in his original publication, A. Maggs opted for a Monte Carlo propagation of the electric fields in the system. This means, he came up with a rotational plaquette move on the lattice that would exactly satisfy the Gauss law boundary condition 3.2. The plaquettes and moves are again picked at random and performed according to the Metropolis criterion, resulting in a diffusive propagation of the electric fields. The particle and field updates are depicted in Figure 3.1.

This will, of course, result in retarded solutions of the Maxwell equations, but the dynamics of the particles remain intact, as has been shown in parallel to Car-Parrinello Molecular Dynamics, since the Gauss law is precisely kept by the allowed particle and Monte Carlo moves.

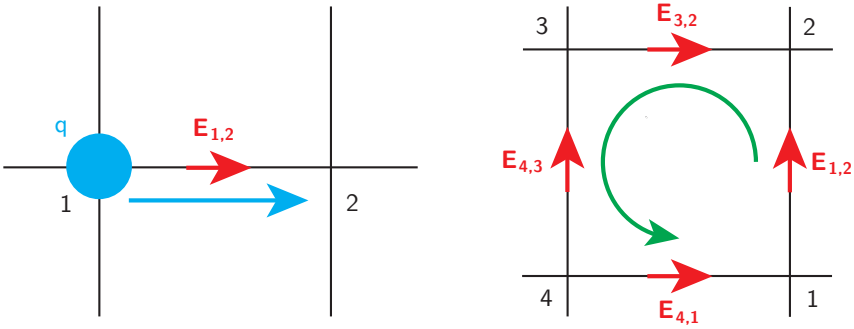



**Figure 3.1.:** Maggs update scheme for particles and plaquettes. Particles can only move via on-lattice Monte Carlo moves and will result in a change in the local electric field when doing so. Those field changes are propagated in a diffusive Monte Carlo manner, using a rotational move that obeys the boundary condition of the Gauss law.

In later publications, J. Rottler and A. Maggs [155, 156] , parallel to I. Pasichnyk and B. Dünweg [151] extended this algorithm to particles that can perform off-lattice moves in non-discretized space. This is achieved by interpolating the motion of a particle to the nearby lattice sites. The particle motion is then interpreted as an electric current density $\mathbf{j}$ with the charge conservation law

$$\nabla \mathbf{j} = -\dot{\rho}, \quad \text{or} \tag{3.4}$$

$$\mathbf{j} = q_p \mathbf{v}_p, \tag{3.5}$$

where $q_p$ and $\mathbf{v}_p$ are a particle's charge and displacement vector, respectively. This electric current then leads to the generation of an electromagnetic field via the Maxwell equations

$$\nabla \times \mathbf{B} = \varepsilon_0 \dot{\mathbf{E}} + \mathbf{j} \tag{3.6}$$

$$\nabla \times \mathbf{E} = -\dot{\mathbf{B}}. \tag{3.7}$$

The particle and field propagation was again done with the same Monte Carlo scheme by J. Rottler and A. Maggs, while I. Pasichnyk

and B. Dünweg aimed for a different approach that will be presented in the next section.

The original algorithm as introduced by A. Maggs is not widely used today for a few reasons. First, it is fairly complex to implement with the certainty that the correct physical behavior of the system is reproduced, since in this Monte Carlo algorithm, very close attention needs to be paid to detailed balance. And second, like most Monte Carlo schemes it can not be easily parallelized. Attempts have been made to introduce bigger cluster moves for the field propagation [182, 183, 198] that will speed up the simulation and allow for parallelization on smaller compute clusters. But the algorithm's intrinsic locality theoretically allows for larger scale parallelization, and we therefore prefer its adaptation to Molecular Dynamics for our extensions.

A key difference between "Maggswellian Dynamics" and more traditional electrostatics algorithms is the aforementioned intrinsic locality, which no other approach can claim so far. On one hand, this can be translated to a very straight forward numerical parallelization, but on the other hand, and most importantly for us, it theoretically allows for arbitrary local changes in some parameters such as the dielectric permittivity. While the possibility for the inclusion of local changes in the dielectric parameters has been mentioned in a few publications, most prominently by A. Maggs [183], and by J. Rottler [198, 269], it had not been entirely thought through and implemented prior to our work.

A group around S. Pasquali and A. Maggs has adapted the algorithm to feature variations of dielectric contrasts in time [219, 220, 232]. This has been used to study Casimir forces and Lifshitz interactions between dielectric boundaries. But this implementation does not allow for charges in the system and is not suited for our purposes. It needs to be added that claims have been made by A. Duncan, R. D. Sedgewick, and R. D. Coalson that the interactions measured here are not realistic and rise from missing corrections to the algorithm itself [165, 173], but the problems were not reproducible in all existing implementations of the algorithm, and in fact the agreement with theoretical predictions as presented 2 years later by S. Pasquali has rendered them moot. Even if the claims were justified, the corrections would be negligible in com-

parison to all other numerical errors we had to face and would not be noticeable in our results. This is why we have chosen not to include their proposed correction scheme via multi-bosons in our implementation.

### 3.1.2. Adaptation for Molecular Dynamics and similarity to hydrodynamics

So far, the "Maggswellian Dynamics" algorithm has been formulated for Monte Carlo simulations with a diffusive Monte Carlo field propagation. In 2004, both I. Pasichnyk and B. Dünweg [151] as well as J. Rottler and A. Maggs [155] simultaneously published a translation to Molecular Dynamics simulations. This means that particles were able to move freely and off-lattice, and that not just single Monte Carlo particle moves were allowed but all particles were being displaced at once. While J. Rottler kept the original diffusive field propagation and only changed it a few years later [198], I. Pasichnyk and B. Dünweg suggested a wave-like propagation of the fields, very much like a real electrodynamics simulation, but with an artificially reduced speed of light. Our presentation here uses to an identical algorithm, but is more concise and suited for the inclusion of a varying dielectric permittivity.

Electrodynamics bares close resemblance to hydrodynamics, conceptually and in terms of the underlying mathematics. Hydrodynamics is based on Navier-Stokes equation for incompressible Newtonian fluids without external force

$$\dot{\mathbf{u}} + \mathbf{u} \cdot \nabla \mathbf{u} - \nu \Delta \mathbf{u} = -\nabla w, \tag{3.8}$$

where $\mathbf{u}$ is the local fluid velocity, $\nu$ is the kinematic viscosity, and $w$ is the thermodynamic work. If the steady state of this equation is linearized, it simplifies to the Stokes equations including mass conservation

$$\mu \Delta \mathbf{u} + \mathbf{f} = 0 \tag{3.9}$$

$$\frac{d\rho_{\mathrm{fl}}}{dt} + \rho_{\mathrm{fl}} \nabla \mathbf{u} = 0, \tag{3.10}$$

where $\mu$ is the dynamic viscosity, $\mathbf{f}$ is an applied body force, and $\rho_{\text{fl}}$ is the fluid density. This can be directly compared to the Poisson equation and charge conservation for electrostatics,

$$\varepsilon \Delta \Phi + \rho_f = 0 \tag{3.11}$$

$$\frac{\partial \rho}{\partial t} + \nabla \mathbf{J} = 0, \tag{3.12}$$

where $\varepsilon$ is the dielectric permittivity, $\Phi$ is the electrostatic potential, $\rho_f$ is the free charge density, and $\mathbf{J}$ is the electric current. While the Stokes equations are solved for vector values $\mathbf{u}$ and not scalars $\Phi$, the resemblance can clearly be seen. Notably, the sources $\mathbf{f}$ in hydrodynamics are of vector value, or dipoles, giving them not only a magnitude but a direction, which corresponds to moving particles, whereas the sources $\rho_f$ in electrostatics are of scalar form, or monopoles.

The parallels go so far that the implementation of the adapted algorithm that I. Pasichnyk and B Dünweg presented is closely related to a Lattice-Boltzmann implementation, featuring propagating waves that travel one cell width per time step. In this section, we will prove in a novel way that starting with the Poisson equation for electrostatics and some straight forward assumptions, we necessarily arrive at a full electromagnetic formalism and all four Maxwell equations. We will also use this Ansatz to show that for this particular algorithmic approach, the wave propagation speed or speed of light $c$ can be adjusted as a parameter and that the static observables obtained in a simulation will not be affected by it and remain intact.

All calculations from here on will not assume a constant background permittivity $\varepsilon$, but a space dependent $\varepsilon(\mathbf{r})$. We will also start to use the abbreviation MEMD (Maxwell Equations Molecular Dynamics) that has been introduced by I. Pasichnyk and B. Dünweg, since the approach presented here is quite similar to theirs.

### 3.1.3. From electrostatics to the Maxwell equations

In a system containing a non-zero charge distribution, the electrostatic potential $\Phi$ is often described via the Poisson equation in SI units

$$\nabla\varepsilon\nabla\Phi = -\rho, \tag{3.13}$$

where $\varepsilon$ is the dielectric permittivity, $\rho$ is the charge distribution, and $\Phi$ marks the electrostatic potential. In this form of the equation, the dielectric permittivity $\varepsilon$ can vary in space and time. To simulate a system containing such a charge distribution, a numerical approach to solve the Poisson equation must be found. Algorithms for the solution of this equation, especially in periodic boundary conditions, are a challenge, since the problem is merely conditionally convergent and the trivial approach requires $\mathcal{O}(N^2)$ operations for $N$ particles. This can be reduced using a number of methods, including multigrid and multipole approaches, and by means of Fourier transformation with the Ewald summation or the Particle-Particle-Particle-Mesh (P3M) algorithm, although most of these algorithms can only deal with a constant dielectric permittivity $\varepsilon$, simplifying the Poisson equation. A short description and comparison of these methods can be found in chapter 5, but they all have one thing in common: They solve the electrostatic potential following the Poisson equation (3.13). To get the resulting forces acting on particles, one still needs to calculate the electric field via the derivative of said potential.

The MEMD algorithm follows a different approach. Instead of solving the Poisson equation (3.13) for the electrostatic potential, it is reformulated using the definition $\mathbf{D} = \varepsilon\mathbf{E}$ and the electric field $\mathbf{E} = -\nabla\Phi$, resulting in the first Maxwell equation or *Gauss law*,

$$\nabla\mathbf{D} = \rho. \tag{3.14}$$

There are several reasons why solving the Gauss law instead of the original Poisson equation is exciting: First and most obvious, you will immediately get the electric field (locally dividing $\mathbf{D}$ by $\varepsilon$) without need for a derivative. But second, this equation poses a very different numerical challenge. Where the original Poisson equation is an elliptic partial differential equation, requiring information on the global charge distribution, the Gauss law can be classified as a hyperbolic partial differential equation and therefore is intrinsically local in nature. This is very interesting from a numerical viewpoint, because not only would

such an algorithm most likely scale linearly $\mathcal{O}(N)$ for a constant average charge density in the system, but there would be no restrictions on local changes of the dielectric permittivity $\varepsilon$.

If we solve this equation once in the beginning and afterwards only apply valid updates, we should end up with the correct Gauss law (3.14) at all times. Using the charge conservation law $\dot{\rho} + \nabla \mathbf{j} = 0$, we arrive at the constraint

$$\nabla \cdot \left( \dot{\mathbf{D}} + \mathbf{j} \right) = 0$$

for the incremental update of our **D**-field. The simplistic solution for this constraint

$$\mathbf{D} = - \int_0^\tau \mathbf{j}(\tau) d\tau \tag{3.15}$$

produces physically incorrect results. As an example, we may look at a ring current in a metal, where $\mathbf{j} \neq 0$, $\rho = 0$, and $\dot{\rho} = 0$. The Gauss law (3.14) immediately shows that the displacement field for this problem is zero, but solving equation (3.15) would give a non-zero field which linearly increases in time. Therefore, it is clear that the general solution $\mathbf{D} + \mathbf{D}'$ of our incremental update field (obeying $\nabla \mathbf{D}' = 0$) must contain a transversal component we want to call $\mathbf{\Theta}$. This yields the most general form of constraint to obey the Gauss law:

$$\dot{\mathbf{D}} + \mathbf{j} - \nabla \times \dot{\mathbf{\Theta}} = 0 \tag{3.16}$$

To examine the propagation of such an algorithm, we will take a look at the phenomenological Lagrangian of the constrained system,

$$\begin{aligned} L = & \sum_i \frac{m_i}{2} \mathbf{v}_i^2 - U \\ & + \frac{f_{\text{mass}}}{2} \int \varepsilon \dot{\mathbf{\Theta}}^2 d\mathbf{r} - \frac{1}{2} \int \frac{\mathbf{D}^2}{\varepsilon} d\mathbf{r} \\ & + \int \mathbf{A} \left( \dot{\mathbf{D}} - \nabla \times \dot{\mathbf{\Theta}} + \mathbf{j} \right) d\mathbf{r}, \end{aligned} \tag{3.17}$$

where we use the Lagrange multiplier $\mathbf{A}$ to impose the kinematic constraint in the last term. The kinetic energy is described by the particle masses $m_i$ and velocities $\mathbf{v}_i$, and an external potential $U$ is introduced. The prefactor $f_{\text{mass}}$ simply denotes the mass of our photon-like particles, analog to the mass of electrons in CPMD. The two second line terms make up the energy of the $\dot{\mathbf{\Theta}}$- and $\mathbf{D}$-fields in the system. We can now apply the well known Lagrangian formalism to determine the equations of motion with the use of variational calculus. Variation with respect to $\dot{\mathbf{r}}_i$ gives us

$$\begin{aligned} \frac{\partial L}{\partial \dot{r}_i^\alpha} &= m_i \dot{r}_i^\alpha + q_i A^\alpha(\mathbf{r}_i) \\ \frac{d}{dt}\frac{\partial L}{\partial \dot{r}_i^\alpha} &= m_i \ddot{r}_i^\alpha + q_i \dot{A}^\alpha(\mathbf{r}_i) + q_i \frac{\partial A^\alpha}{\partial r_i^\beta} \dot{r}_i^\beta, \end{aligned} \tag{3.18}$$

where for the second line we just take the time derivative. Variation with respect to $\mathbf{r}_i$ yields

$$\frac{\partial L}{\partial r_i^\alpha} = -\frac{\partial U}{\partial r_i^\alpha} + q_i \dot{r}_i^\beta \frac{\partial A^\beta}{\partial r_i^\alpha} \tag{3.19}$$

Combining these two results (3.18) and (3.19) to

$$m_i \ddot{r}_i^\alpha = -\frac{\partial U}{\partial r_i^\alpha} - q_i \dot{A}^\alpha + q_i \dot{r}_i^\beta \left( \frac{\partial A^\beta}{\partial r_i^\alpha} - \frac{\partial A^\alpha}{\partial r_i^\beta} \right), \tag{3.20}$$

and introducing the vector

$$\mathbf{B} = \nabla \times \mathbf{A}, \tag{3.21}$$

we get the equations of motion for the particles:

$$m_i \ddot{\mathbf{r}}_i = -\frac{\partial U}{\partial \mathbf{r}_i} - q_i \dot{\mathbf{A}} + q_i \mathbf{v}_i \times \mathbf{B}, \tag{3.22}$$

The equations of motion for the electromagnetic fields can be found by varying the Lagrangian density $\mathcal{L}$, which by definition satisfies $L = \int \mathcal{L} d^3\mathbf{r}$. Variation in $\dot{\mathbf{\Theta}}$ and in time gives us

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\Theta}}} &= f_{\text{mass}}\varepsilon_0 \dot{\boldsymbol{\Theta}} - \varepsilon_0 \nabla \times \mathbf{A} = f_{\text{mass}}\varepsilon_0 \dot{\boldsymbol{\Theta}} - \varepsilon_0 \mathbf{B} \\
\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\Theta}}} &= f_{\text{mass}}\varepsilon_0 \ddot{\boldsymbol{\Theta}} - \varepsilon_0 \dot{\mathbf{B}} = 0 \\
f_{\text{mass}}\ddot{\boldsymbol{\Theta}} &= \dot{\mathbf{B}} \qquad (3.23) \\
\frac{1}{c^2}\dot{\boldsymbol{\Theta}} &= \mathbf{B} \qquad (3.24)
\end{aligned}$$

where we used the natural initial condition $\dot{\boldsymbol{\Theta}}(t = 0) = 0$ in the last step and replaced $f_{\text{mass}}$ with $1/c^2$, where $c$ later turns out to be the speed of light. The next variation in $\mathbf{E}$ gives us

$$\dot{\mathbf{A}} = -\mathbf{E} \qquad (3.25)$$

which leads to the more commonly know expression for equation (3.22). With these two last results (3.24) and (3.25), we can obtain two more Maxwell equations by inserting them into the constraint equation (3.16), namely Ampère's and Faraday's law:

$$\begin{aligned}
\dot{\mathbf{D}} &= c^2 \nabla \times \mathbf{B} - \mathbf{j} \qquad (3.26) \\
\dot{\mathbf{B}} &= \nabla \times \dot{\mathbf{A}} = -\nabla \times \mathbf{E} \qquad (3.27)
\end{aligned}$$

Furthermore, combining the time derivative of equation (3.25) with equations (3.26) and (3.27) yields the equations of motion for the artificial vector field $\mathbf{B}$

$$\frac{\partial^2}{\partial t^2}\mathbf{B} = c^2 \nabla \times \nabla \times \mathbf{B}. \qquad (3.28)$$

This is a standard wave equation, implying a wave-like propagation of the artificial field without any additional assumptions.

Remarkably, this means that only by applying the above mentioned constraint (3.16) and by a standard Lagrangian treatment, we naturally reproduce the complete electromagnetic formalism, including magnetic waves and the Lorentz force. It should be noted that the

equations (3.21) and (3.25) represent nothing but the so-called temporal gauge or Weyl gauge in electromagnetism, in which the scalar potential $\Phi$ is identically zero at infinity. This turns out to be the most appropriate gauge for our purposes.

### 3.1.4. Conservation laws

Since the Lagrangian we used is constrained, we can not simply construct a Hamiltonian from it. This would be nice to simplify further proofs for the conservation of phase-space volume, energy and momentum. However, we can construct a Lagrangian that is not constrained and produces the exact same equations of motion, as it is common practice in electromagnetism [110]. This new Lagrangian reads

$$\begin{aligned} L = &\sum_i \frac{m_i}{2}\mathbf{v}_i^2 - U + \frac{\varepsilon}{2}\int \dot{\mathbf{A}}^2 d^3\mathbf{r} \\ &- \frac{c^2}{2}\int \varepsilon(\nabla\times\mathbf{A})^2 d^3\mathbf{r} + \int \mathbf{A}\cdot\mathbf{j} d^3\mathbf{r} \end{aligned} \tag{3.29}$$

The dynamics for particles and fields can be derived via the same variational calculus formalism from equation (3.29) and the Lagrangian density like before and the new equations of motion, without giving the detailed derivation here, hence read

$$m_i\ddot{\mathbf{r}}_i = -\frac{\partial U}{\partial \mathbf{r}_i} - q_i\mathbf{E} + q_i\mathbf{v}_i\times\mathbf{B} \tag{3.30}$$

$$\mathbf{B} = \frac{1}{c^2}\dot{\mathbf{\Theta}} \tag{3.31}$$

$$\dot{\mathbf{D}} = c^2\,\nabla\times\mathbf{B} - \mathbf{j} \tag{3.32}$$

$$\dot{\mathbf{B}} = -\nabla\times\mathbf{D}. \tag{3.33}$$

In our implementation, the magnetic part of the Lorentz force, $\mathbf{v}\times\mathbf{B}$ from equation (3.30), is omitted. This increases the speed significantly, but makes it impossible to construct an unconstrained Lagrangian from eq. (3.29), and therefore the Hamiltonian nature of the algorithm does

not hold. Momentum conservation is violated by the amount of momentum that the virtual photons carry. This is a negligible percentage and perfect momentum conservation is not important in most simulated systems since many contain a thermostat.

If we introduce the new variables

$$\mathbf{P}_i = m_i \dot{\mathbf{r}}_i + q_i \mathbf{A}(\mathbf{r}_i) \tag{3.34}$$

$$\mathbf{p}_A = \varepsilon \dot{\mathbf{A}} \tag{3.35}$$

where $\mathbf{P}_i$ are the canonically conjugate momenta of the particles, and $\mathbf{p}_A$ describe the field momenta. By means of Legendre transformation, we can construct the Hamiltonian

$$\begin{aligned}\mathcal{H} = &\sum_i \frac{1}{2m_i} \left[\mathbf{P}_i - q_i \mathbf{A}(\mathbf{r}_i)\right]^2 + U \\ &+ \int \frac{1}{2\varepsilon} \mathbf{p}_A^2 d^3\mathbf{r} + \int \frac{\varepsilon c^2}{2} \left(\nabla \times \mathbf{A}\right)^2 d^3\mathbf{r}.\end{aligned} \tag{3.36}$$

The equations of motion for our new variables, as derived from this Hamiltonian, are

$$\frac{d}{dt}\mathbf{r}_i = \frac{1}{m_i}\left(\mathbf{P}_i - q_i \mathbf{A}(\mathbf{r}_i)\right) \tag{3.37}$$

$$\frac{d}{dt}P_i^\alpha = -\frac{\partial U}{\partial r_i^\alpha} + \frac{q_i}{m_i}\left[P_i^\beta - q_i A^\beta(\mathbf{r}_i)\right]\frac{\partial A^\beta}{\partial r_i^\alpha} \tag{3.38}$$

$$\frac{d}{dt}\mathbf{A} = \frac{\mathbf{p}_A}{\varepsilon} \tag{3.39}$$

$$\frac{d}{dt}\mathbf{p}_A = \sum_i \frac{q_i}{m_i}\left(\mathbf{P}_i - q_i \mathbf{A}(\mathbf{r}_i)\right)\delta(\mathbf{r} - \mathbf{r}_i) - \varepsilon c^2 \nabla \times \nabla \times \mathbf{A} \tag{3.40}$$

A very welcome property of Hamiltonian dynamics is that momentum and energy is intrinsically conserved. But if we change the resulting equations of motion for the particles (3.37) and (3.38) to omit the magnetic field part of the Lorentz force, we destroy the Hamiltonian structure of the system. As stated before, momentum conservation is

violated, but we can show that the far more crucial energy conservation still holds. For this, we take a look at the Hamiltonian energy

$$\begin{aligned}\mathcal{H} = &\sum_i \frac{m_i}{2}\dot{\mathbf{r}}_i^2 + U \\ &+ \frac{1}{2}\int \frac{\mathbf{D}^2}{\varepsilon} d^3\mathbf{r} + \frac{c^2}{2}\int \varepsilon \mathbf{B}^2 d^3\mathbf{r}\end{aligned} \tag{3.41}$$

The conservation of energy is equivalent to a conservation of phase space volume, better known as the Liouville theorem for electrodynamics, and we can apply the same strategy to our system. Generally, if we are given a system of $n$ ordinary differential equations

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}), \qquad \mathbf{x} = (x_1, \ldots, x_n), \tag{3.42}$$

and their solution may be extended to the whole time axis. Let $g^t$ be the the corresponding group of time transformations

$$g^t(\mathbf{x}) = \mathbf{x} + \mathbf{f}(\mathbf{x})t + \mathcal{O}(t^2), \qquad (t \to 0). \tag{3.43}$$

Then if

$$\nabla \mathbf{f} = \frac{\partial f_i}{\partial x_i} \equiv 0, \tag{3.44}$$

the map $g^t$ conserves the volume in $\mathbf{x}$-space [58]. For our Hamiltonian system this implies

$$\nabla \mathbf{f} = \frac{\partial}{\partial p}\left(-\frac{\partial \mathcal{H}}{\partial q}\right) + \frac{\partial}{\partial q}\left(\frac{\partial \mathcal{H}}{\partial p}\right) \equiv 0. \tag{3.45}$$

If we apply this to our system, the flow divergence is

$$\begin{aligned}\nabla \cdot \mathbf{f}_{\text{coul}} = &-\frac{\partial}{\partial \mathbf{A}}\mathbf{E} + \frac{\partial}{\partial \mathbf{E}}\left(c^2 \nabla \times (\nabla \times \mathbf{A}) - \mathbf{j}\right) \\ &+ \frac{\partial}{\partial \mathbf{r}_i}\frac{1}{m_i}\mathbf{p}_i + \frac{\partial}{\partial \mathbf{p}_i}\left(q_i \mathbf{E}(\mathbf{r}_i) - \frac{\partial U}{\partial \mathbf{r}_i}\right),\end{aligned} \tag{3.46}$$

which is trivially identically to zero. This means that the phase space volume and with it the energy is conserved by the algorithm. We will

not show the complete derivation here, since it is equivalent to the derivation by I. Pasichnyk [152]. However, if we apply a similar ansatz to obtain the momentum conservation, equation (3.40) contributes an additional term and the total momentum follows the relation

$$\sum_i \mathbf{p}_i = \text{const.} + \mathcal{O}(c^{-2}). \tag{3.47}$$

### 3.1.5. Reproduction of observables

Like in the algorithm for constant background permittivity, the thermodynamic observables are perfectly reproduced, since they are not dependent on the speed of light nor the magnetic field component. In contrast to the original version, the partition function in this extended algorithm contains an extra term for the varying permittivity. The particle momenta and the vector field $\mathbf{A}$ can still be integrated out in a straightforward way. If we split up the integration of the electric field in a longitudinal and a transversal component, we end up with

$$\begin{aligned}\mathcal{Z} = \int \prod_{i=1}^{N} d\mathbf{r}_i \prod_{\mathbf{r}} \mathcal{D}\mathbf{D}^{\parallel}(\mathbf{r}) \mathcal{D}\mathbf{D}^{\perp}(\mathbf{r})\, \delta\left(\nabla \cdot \mathbf{D} - \rho(\mathbf{r})\right) \\ \times \exp\left(-\frac{\beta}{2} \int d\mathbf{r} \frac{\mathbf{D}^{\parallel}(\mathbf{r})^2}{\varepsilon(\mathbf{r})}\right) \\ \times \exp\left(-\frac{\beta}{2} \int d\mathbf{r} \frac{\mathbf{D}^{\perp}(\mathbf{r})^2}{\varepsilon(\mathbf{r})}\right).\end{aligned} \tag{3.48}$$

The integration over the transversal component also only contributes a factor, and the longitudinal component cancels with the delta function. This contribution of the transversal component is constant for a static dielectric background, but can vary if the dielectric interfaces are mobile. This gives rise to thermal Casimir/Lifshitz interactions as discussed by Pasquali and Maggs [219, 220, 232], but the effect will not be discussed further in this thesis since we focus on moving charges and only allow slow and smooth temporal variations of the dielectric

background. The only degrees of freedom now left are the particle coordinates, which finally leads to

$$\mathcal{Z} = \int \prod_{i=1}^{N} d\mathbf{r}_i \exp\left(-\frac{\beta}{2}\int d\mathbf{r}\frac{\mathbf{D}(\mathbf{r})^2}{\varepsilon(\mathbf{r})}\right). \tag{3.49}$$

This is what is expected from the static case of electromagnetic interactions. Here, $\mathbf{D}$ is nothing but the solution of the standard electrostatic problem

$$\nabla \cdot \mathbf{D} = \rho \tag{3.50}$$

$$\nabla \times \frac{\mathbf{D}}{\varepsilon} = 0. \tag{3.51}$$

Inserting this field into equation (3.49), we find the standard Coulomb Hamiltonian. This means that even with our retarded solutions, the particles behave statistically exactly as if they would interact directly via electrostatics.

## 3.2. Completing the original algorithm

The algorithm as described in this chapter so far is a very good starting point, since it provides the option for spatially and temporally varying dielectric properties due to its intrinsic locality. But there are still some enhancements and corrections that arise from the algorithm itself and from its discretization in space and time. In this section, we will describe the problems we encountered and one or more solutions to each of them.

Some of the problems – namely the self energy correction and initial solution – had been noticed and resolved in the implementation by I. Pasichnyk but have to be addressed differently if locally varying permittivity is involved. The other problems – error estimates, a drift in the initial solution, and the dipole term correction – arose while we thoroughly tested the validity of the algorithm against analytical solutions and comparing simulations with other algorithms.

### 3.2.1. Initial solution

The algorithm we have described so far is used each time step at the core of a Molecular Dynamics simulation. But to arrive at the constraint equation (3.16) earlier, we have taken a time derivative. Thus, satisfying the constraint will only apply updates to an already correct initial solution of the Gauss law (3.14) which needs to be determined in another way. This is the starting point of our time propagation and any error can increase dramatically following a positive Lyapunov exponent [66]. It is therefore crucial that this solution follows the Gauss law of electrodynamics as precise as numerically possible.

In general, the dynamic analogue of the Poisson equation for electrostatics (with the speed of light $c \to \infty$) consists of the two equations (3.50) and (3.51). For our purpose, we will use the fact that the latter of these equations, $\nabla \times \mathbf{D}/\varepsilon = 0$ is equivalent to a minimization of the electric field energy

$$\mathcal{H}_{\mathrm{EF}} = \frac{1}{2} \int \frac{\mathbf{D}^2}{\varepsilon} d^3\mathbf{r}. \tag{3.52}$$

If we find a scheme that minimizes this energy and follows the Gauss law exactly, we can apply incremental updates as described before and stay on the constraint surface.

Of course, it is possible to use an Ewald based method for this, but the possibility of dealing with different dielectric properties within the system is then lost. So we introduce a very simple (but slow) recursive scheme that is graphically described in Figure 3.2.

First, the charges on each plane are averaged to $q_{\mathrm{plane}}$, scaled by the lattice size and local permittivity, and added to the field on each node

$$E_z^{(n+1)} = E_z^{(n)} + \frac{q_{\mathrm{plane}}}{\varepsilon_z^{x,y,n} a^2}, \tag{3.53}$$

and then the charge $q_{\mathrm{plane}}$ is subtracted from each vertex in the $z_n$-plane. The charges in $y$- and $x$-direction are averaged and updated

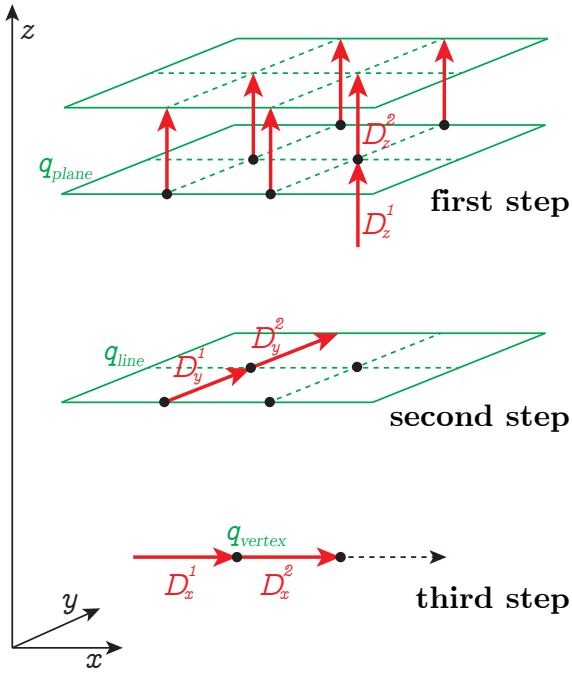



**Figure 3.2.:** Recursive scheme for the initial solution of the **D**-field. The average charge in the z-plane $q_{\text{plane}}$ is added to the charge and field at each node, following $E_z^{n+1} = E_z^n + \frac{q_{\text{plane}}}{\varepsilon a^2}$. Then we again subtract the charge $q_{\text{plane}}$ from each charge in the $z$-plane and the average field of the system in $z$-direction from each node. We proceed exactly the same with the average charge $q_{\text{line}}$ in $y$-direction and the point charges $q_{\text{vertex}}$ in $x$-direction. At the end, we place the original charge on each vertex. The sketch has been published in [4].

accordingly on lines and vertices, following

$$E_y^{(n+1)} = E_y^{(n)} + \frac{q_{\text{line}}}{\varepsilon_y^{x,z,n} a^2} \tag{3.54}$$

$$E_x^{(n+1)} = E_x^{(n)} + \frac{q_{\text{vertex}}}{\varepsilon_y^{y,z,n} a^2}. \tag{3.55}$$

Summation of the total charge in one cell is given by

$$q_{\text{plane}} + q_{\text{line}} + q_{\text{vertex}} \tag{3.56}$$

and this yields the Gauss law directly, if the $(\nabla\cdot)$ operator is defined via finite differences $E^{(n+1)} - E^{(n)}$.

To minimize the field energy (3.52), we follow a successive over-relaxation, a standard approach in numerical mathematics [67]. This is a very ineffective way to calculate the electric field in a system. It not only has an asymptotic complexity of $\mathcal{O}(N^3)$ if $N$ is the number of lattice cells or the number of particles provided the system has roughly homogenous density. But because of the energy relaxation, it also contains a fairly large prefactor to this scaling and is not very efficiently parallelizable. To investigate the scaling behavior of the algorithm, we ran a simulation of several $10^9$ particles on just one compute core. While the dynamic MEMD algorithm was able to calculate each force update in less than two seconds, the initial solution took four orders of magnitude longer! This is, however, an extreme example and normal applications in biophysical simulations will be less off balance.

We chose this initial scheme nonetheless, since it has three key advantages: It easily allows for spatially varying dielectric properties, it is in compliance to the Gauss law down to machine precision, and we can manually control the total numerical error of the electrostatic potential by setting an exit condition for the energy minimization routine accordingly. We will expand on this for our error estimate in section 3.2.3.

It should be stated here that a very straight forward approach to minimize the electric field energy within this algorithm can be achieved by starting with any solution of the Gauss law and running the dynamic part of the algorithm with open boundary conditions, and with an integration scheme that is not time reversible and therefore not energy

conserving. In this case, the excess energy of the electromagnetic fields will be carried away by magnetic field waves and the system will equilibrate to a minimal field energy state. We have tried this method first, since the costly $\mathcal{O}(N^3)$ scaling has to be done only once, but it requires even more computational effort and was not beneficial, not even in the aforementioned extreme test case of $> 10^9$ particles.

### 3.2.2. Self energy correction

Even in the continuum, the solution of the Maxwell equations for point charges is singular at the position of the particle. The point charge carries along with it the electrostatic energy

$$\frac{1}{2} \int\limits_{|\mathbf{r}-\mathbf{r}_i(t)|\leq R} \frac{\mathbf{D}(\mathbf{r},t)^2}{\varepsilon} d^3\mathbf{r} \propto \int\limits_0^R r^2 (r^{-2})^2 dr = \int\limits_0^R r^{-2} dr$$

which is a diverging integral. This would mean that the particle has infinite mass and can not respond to forces. With a lattice spacing, a "cut-off" is introduced for this self-interaction, but still the particle is driven to the center of the cell by the field created from its own (interpolated) charge. It is, from an energy point of view, the most favorable for the particle to distribute its charge evenly on all surrounding lattice points, since it then produces the smallest possible curl in the cell and therefore no **B**-field. The electrostatic energy of a particle within a cell is shown in Figure 3.3.

This spurious self influence is in the original algorithm corrected by the use of Yukawa-potentials. However, if the permittivity of the system changes within the cell, every potential based correction scheme fails. With some trickery, however, this problem can be solved both with a lattice Greens function, or a direct subtraction scheme.

In our implementation, the permittivity, as it is a differential 1-form, is placed on the lattice links. Therefore it remains constant on the link, allowing us to set up a Green's function of the form

$$\Delta_{\mathbf{r}'} G(\mathbf{r} - \mathbf{r}') = -\frac{1}{a^2} \delta_{\mathbf{r},\mathbf{r}'} \tag{3.57}$$

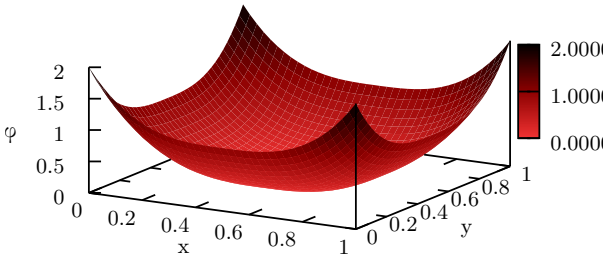



**Figure 3.3.:** The uncorrected electrostatic energy of a particle within a lattice cell. Units are in cell sizes and we show a two-dimensional map through $z = 0.5$ of a cell. The potential energy shows cubic behavior and a distinct minimum in the cell center to which the particle will be driven.

if $\mathbf{r}$ and $\mathbf{r}'$ are placed on adjacent lattice sites. For a point charge, the electrostatic potential can then be found using a convolution with the Green's function

$$\phi(\mathbf{r}) = \frac{q}{a\varepsilon(\mathbf{r} - \mathbf{r}')} G(\mathbf{r}) \tag{3.58}$$

This Green's function for a point charge on a cubic lattice can now be solved by a Fourier transform and limitation to the first Brioullin zone. For an infinitely large lattice, the back transform yields the integral

$$G(\mathbf{r}) = \left(\frac{a}{2\pi}\right)^3 \int\limits_{\mathbf{k}\in \mathrm{BZ}} \frac{e^{i\mathbf{k}\mathbf{r}}}{\varepsilon(\mathbf{k})} d^3k \tag{3.59}$$

If the Laplace operator on the left-hand side of equation (3.57) is used to construct a finite-differences operator and applied to each of the interpolated charges on the lattice, we end up with a solution of the self-energy influence that can be added up. The solution for the given integral only needs to be calculated once at the beginning of the simulation and it can be done analytically [113].

Another approach to allow for dielectric variations on very small scales (within one lattice site) is to use a direct subtraction scheme. With the assumption that the gradient of the permittivity, $\nabla\varepsilon(\mathbf{r})$, is constant on each lattice link, the influence of the interpolated charges can be directly calculated and subtracted. This also requires the charge interpolation scheme to be of linear order and, after some algebra, results in

$$\mathbf{E} = \sum_{d=x,y,z} \sum_{\substack{i,j=0 \\ i\neq j}}^{1} \varepsilon(r_{j,d} - r_{i,d}) \frac{q \cdot a \cdot (r_i r_{i,d} + r_j r_{j,d})}{r_i r_j r_{i,d} r_{j,d}} \tag{3.60}$$

where $q$ denotes the charge of the particle, $a$ the lattice spacing, $r_{i,d} = |\mathbf{r} - \mathbf{r}_{i,d}|$ the position of the charge relative to the vertex $(i,d)$, and $r_i = |\mathbf{r} \cdot (\mathbf{r}_{j,d} - \mathbf{r}_{i,d})|$ the position of the charge folded onto dimension $d$ relative to the vertex $(i,d)$, and $\varepsilon(r_{j,d} - r_{i,d})$ the permittivity on the lattice link between vertices $(i,d)$ and $(j,d)$.

Both correction schemes as well as the use of Yukawa potentials are included in our implementation within ESPResSo, and the direct scheme is used by default only for spatially varying dielectric systems.

### 3.2.3. Error estimate

It is apparent that MEMD consists of two different algorithms, the initial solver and the dynamic update scheme, both of which should be treated separately for an error estimate. We first investigate the numerical error of the initial solution. As stated in section 3.2.1, the Gauss law is obeyed up to machine precision. We implemented a function to check this and could confirm it. The energy minimization is done iteratively by successive over-relaxation, and we can implicitly set a maximum energy deviation as an exit condition for our implementation.

The initial solution can be easily checked by comparing it to an analytical solution of a known problem. More complicated setups including dielectric enclosures are difficult or impossible to solve, but a

static problem that is solvable and features smoothly varying dielectric permittivity is presented in Figure 3.4.

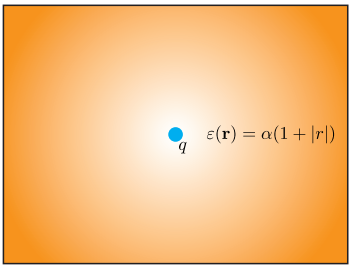



**Figure 3.4.:** Schematic picture of an analytically solvable setup including smoothly varying dielectric permittivity. One charge $q$ is placed in the center and the permittivity $\varepsilon$ linearly increases with the distance $r$. The resulting potential is proportional to $1/r^3$, since the common electrostatic potential $q/r^2$ is additionally dampened by a factor of $\alpha/(1 + r)$.

Here, we put a single charge $q$ in the center at $(0, 0, 0)$ and add a dielectric background which linearly increases from the center following

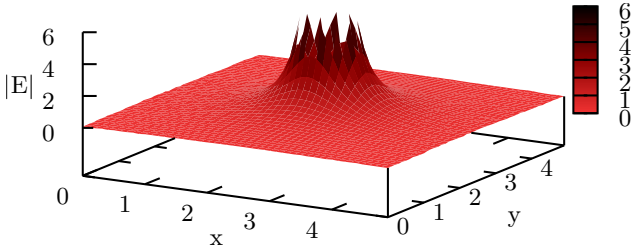
$$\varepsilon(\mathbf{r}) = \alpha(1 + |\mathbf{r}|), \tag{3.61}$$

where $\alpha$ is a prefactor that is set to 1 in our example. Because of the radial symmetry, the analytical solution for the electrostatic potential $\phi$ can simply be integrated out and reads

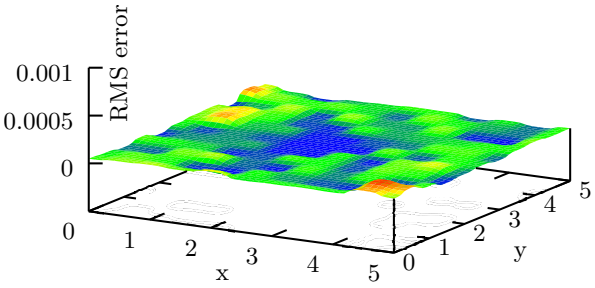
$$\phi = \frac{q\alpha}{r^2(1 + r)} \quad . \tag{3.62}$$

The numerical results and relative deviation from the analytical solution are presented in Figure 3.5.

We set the exit condition for the iterative energy minimization to $10^{-8}$, which is maintained throughout the system. Figure 3.5b shows that in the center, where the absolute potential values are higher, the relative errors are much smaller. This indicates that the error is independent of the field value, which is preferable to ensure that the given




**(a)** electric field values




**(b)** relative error in the **D**-field (root mean square)

**Figure 3.5.:** The numerical results of the initial solver for a charge in the center of a linearly increasing dielectric background permittivity. As expected, the result follows a $1/r^3$ characteristic. The relative numerical error is shown in Figure (b). These graphs have been published in [4].

absolute maximum error is not affected by the initial charge distribution. A more detailed discussion of the nature of these numerical errors and a way to reduce them will be presented in section 3.2.4.

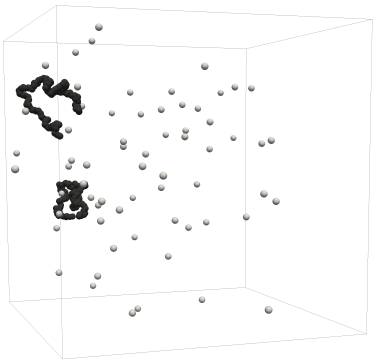
The second algorithmic part that requires an error analysis is the dynamic propagation of the system. We will present two different error sources and show how they are connected. To investigate the validity of these findings, we simulated three artificial test systems using MEMD as well as P3M tuned to a relative force precision of $10^{-9}$ and analyzed the resulting numerical errors by comparing the two simulations. The systems are depicted in Figure 3.6.

The three systems were chosen to represent three different setups that contain different types of error sources. The system of a polyelectrolyte in salt water solution in Figure 3.6a features a very dilute system with a high charge concentration around the polymer. Oppositely, the silica melt in Figure 3.6b is very dense and includes ions of different valency. The third example is an artificial setup of two infinite charged walls and a surrounding cloud of randomly placed charges. This system features a strong dipole moment and a significant long-range contribution throughout the simulation box.

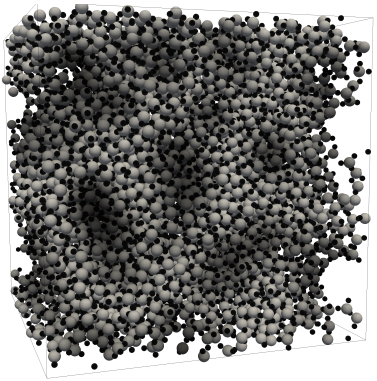
As stated before, other than the physical error from omitting the magnetic part of the Lorenz force, the MEMD algorithm carries a numerical error which consists of two parts.

The first contribution stems from the linear interpolation scheme of the electric current onto the lattice. It can be reduced by introducing a splitting of the far field and the near field part of the Coulombic interaction and calculating the near field (e.g. the 27 surrounding next neighbor lattice cubes) with a direct pairwise potential calculation. This splitting, however, comes with a considerable overload in computational effort, since the near field correction needs to be subtracted from the MEMD algorithm via a lattice Greens function or a similar construct. It is also not possible for spatially varying dielectric properties, since the Coulomb potential based near field approach breaks down for a non-constant dielectric background. This is why, for the extended algorithm, we stick with a linear interpolation scheme.

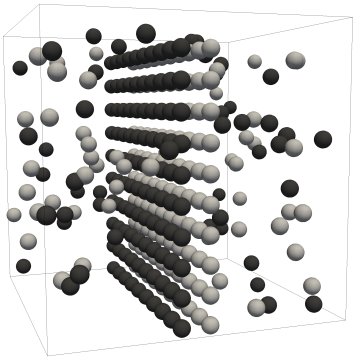
The second contribution is of algorithmic origin and relates to the artificially small speed of light. Since the propagation speed of the



**(a)** polyelectrolyte



**(b)** salt melt



**(c)** cloud-wall

**Figure 3.6.:** Three example systems to determine the numerical error of the algorithm. Two polyelectrolytes in aqueous solution (a), a melting silica crystal (b), and an artificial system (c) with two oppositely charged walls and a surrounding cloud of charges. These sketches have been published in [2].

magnetic fields is finite, the system does not feature true electrostatics but retarded solutions of the Maxwell equation. This error is also indirectly related to the lattice spacing, since a coarser lattice allows for the magnetic fields to be propagated over greater distances in one time step.

The two are connected via the stability criterion of the algorithm:

$$c \ll \frac{a}{dt} \tag{3.63}$$

With a lattice spacing $a$, the error introduced by a linear interpolation scheme scales accordingly with $1/a^3$, whereas the algorithmic error scales with $a^2$ (from equations (3.32) and (3.63)). The resulting overall numerical error is shown in Figure 3.7 for a random distribution of charges.

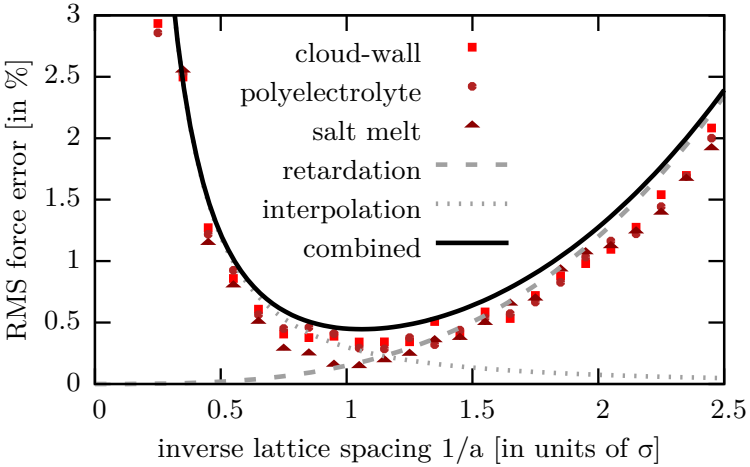



**Figure 3.7.:** Error estimate for the algorithm: The interpolation and the finite speed of light create numerical errors. Both depend on the lattice spacing. The graph has been published in [4].

From Figure 3.7, two things can be deduced: First, the numerical error has a predictable minimum, since the two error sources have a clear mathematical relation. Second, the relative RMS force error for a

typical system does not go below $10^{-3}$. This is sufficient for most MD simulations, but it should be considered before using this algorithm.

It should also be noted that the error increases at very small distances between two charges due to the linear interpolation. For inhomogenous systems with very dense areas not only the error increases but the algorithm slows down significantly, having to propagate all fields into empty regions.

### 3.2.4. Reducing the error

We have worked on tuning and reducing the numerical error for both parts of the algorithm, the initial solver as well as the dynamic propagation. For the initial solution, in addition to reducing the maximally accepted field deviation, we noticed a systematic direction of the deviation from the analytical solution. Hence, we investigated the system with a single charge in the center and a linearly increasing dielectric permittivity. The non-directional electric field energy error of the system, given as

$$\mathcal{H}_{\mathrm{EF}} = \frac{1}{2} \int \frac{\mathbf{D}^2}{\varepsilon}. \tag{3.64}$$

has been shown in Figure 3.5b. If we take a closer look at the electric field $\mathbf{D}$ including directional information and compare it to the analytical solution, we find the deviations presented in Figure 3.8.

It is very apparent that the vector values are not distributed randomly but show a very noticeable tendency towards the $x$-direction. The only situation where we have brought out one spatial direction in comparison to others is the order in which we apply our Gauss law scheme from Figure 3.2. And indeed, the numerical error in $z$-direction, which is summed up first, is negligible. To account for this behavior, we have implemented the initial solver in a way that will switch the order in which the field values are updated in a cyclical manner. This removes the systematic direction of the numerical error entirely and has therefore shown to be the sole source of the problem.

It has to be stated here that this does not "per se" improve the accuracy of the algorithm, since the successive over-relaxation only terminated on reaching the predefined accuracy. But it significantly

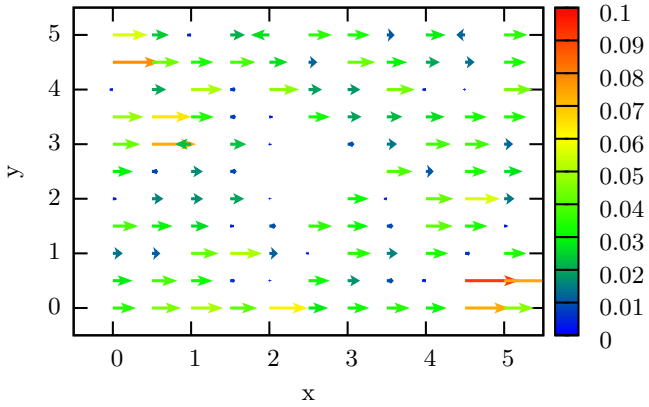
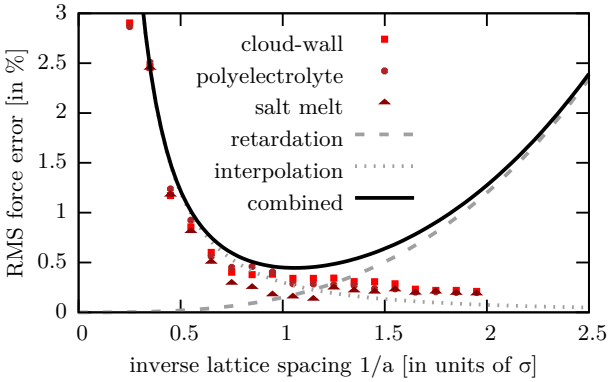



**Figure 3.8.:** Deviation in $x$- and $y$-direction of the field values from the analytical solution, looking at the plane $z = 2.5$. A distinct tendency in $x$-direction is observed. The vector values are increased to be more visible.

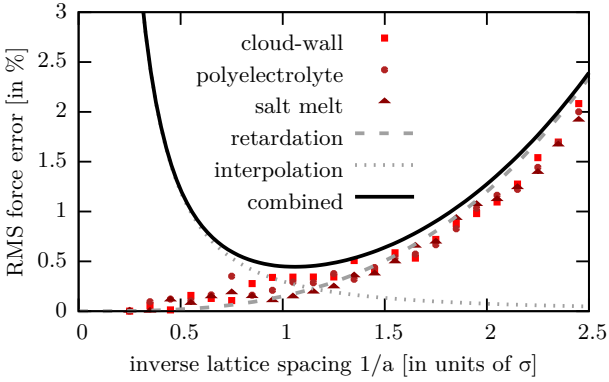
decreases the amount of iteration steps needed to achieve this goal and speeds up the initial solver. It also eliminates any problems that a small systematic drift might have on the simulation.

As depicted in Figure 3.7, the dynamic propagation error is a combination of the field retardation and the interpolation scheme. Both components can be reduced and lead to more accurate results, as can be seen in Figure 3.9.

**Retardation error** The error stemming from the retardation of the electrostatic interaction is directly linked to the parameter $c$ and therefore via equation (3.63) to the propagation speed of the electromagnetic waves. This speed can be artificially increased by keeping the lattice constant but performing multiple propagation moves per particle update step. For our example in Figure 3.9a, we propagated the field eight times per MD time step instead of the usual two times and found that the resulting numerical error was indeed decreased for smaller lattice spacings. However, the average reduction compared to the optimum value at $a = 1\sigma$ is less than a factor of two while the computational effort is increased by a factor of four. For even finer grids and even




**(a)** multiple **B**-field propagation




**(b)** interpolation over $5^3$ cells

**Figure 3.9.:** With additional computational effort, the numerical error can be reduced. The dashed and dotted gray lines show the theoretical error predictions for the retardation and interpolation, respectively, and the black line is the combination of both. The red symbols show the measured numerical errors of three different test cases. (a) If the fields are propagated more than once per time step, finer lattices result in a steadily decreasing numerical error. (b) If the charge interpolation is spread over several lattice sites, coarser lattices lead to smaller numerical errors.

more propagations per time step, the computing time grew exponentially while the relative RMS force error remained above $10^{-5}$. We therefore ruled this out as a feasible method to increase the accuracy of our MEMD implementation.

**Interpolation error** The second error stemming from the linear interpolation scheme can be reduced by interpolating a charge flux on more than the surrounding eight lattice vertices. We used a B-spline interpolation scheme onto $5^3$ surrounding cells. The computational effort was increased by an order of magnitude for comparable grid sizes, but we were able to reach a better accuracy even at four times the lattice constant, which results in a reduced computation time for the field propagation by a factor of $4^3 = 64$. With this interpolation, we were able to reach a relative RMS force error of $10^{-5}$, two orders of magnitude smaller, with an overall increase of a factor of 6 for the computational effort. With more optimized code, this might be an effective way to increase the speed or the accuracy of the MEMD algorithm. For our purposes with spatially varying dielectric properties however, this was merely a proof of concept, since interpolation across several lattice sites in not compatible with neither our initial solver nor our self-energy correction, and we were not able to find a computationally affordable scheme that will guarantee charge conservation when interpolating electric currents across permittivity variations.

Even without these two improvements however, we were able to achieve an RMS force error of $< 10^{-3}$, which is acceptable for biophysical simulations.

### 3.2.5. Periodicity

In MD simulations, the box geometry is often set to be periodic in all dimensions, to avoid boundary effects. This type of boundary condition is introduced very naturally in the MEMD algorithm. Because of its locality, the boundaries and according field propagations can be directly stitched together, creating infinitely many periodic replicas.

In fully periodic charged systems using Ewald-based algorithms, like the particle-particle particle-mesh method P3M, the boundary condi-

tions at infinity have a considerable effect on the solution via the dipole term [36]. For physical simulations, one normally assumes metallic boundaries at infinity to fix the electric field and potential to zero. This assumption cancels the system's overall charge and its dipole moment for force calculations, since the potential and the electric fields at the metallic boundary at infinity are forced to zero. Physically, this allows the algorithm to compensate for a non-neutrality of the system, and it allows the dipole moment of the system to perform an unrestricted random walk. This is intentional, because in realistic systems, the dipole moments of all periodic copies would not be exactly the same but perform individual random walks to cancel out on average after spatial integration [43].

For a standard Ewald method, the correction of the dipole term to the electric field at $\varepsilon(\infty) = 1$ has been calculated in [36] and [74] as

$$\mathcal{H}_\Delta = -\frac{\rho(\varepsilon' - 1)}{2N(2\varepsilon' + 1)} \sum_{1 \le i < j \le N} \mu(i) \cdot \mu(j), \tag{3.65}$$

where $\rho$ is the charge distribution, $N$ is the number of particles, $\mu$ are magnetic dipoles in the system, and $\varepsilon'$ is the permittivity at infinite distance. If this is transferred to electrostatic monopoles, it will create an energetic influence $\Phi_\Delta$ on each particle of

$$\Phi_\Delta = \frac{\mathbf{P}^2}{2(2\varepsilon_b + 1)\varepsilon_0 V}, \tag{3.66}$$

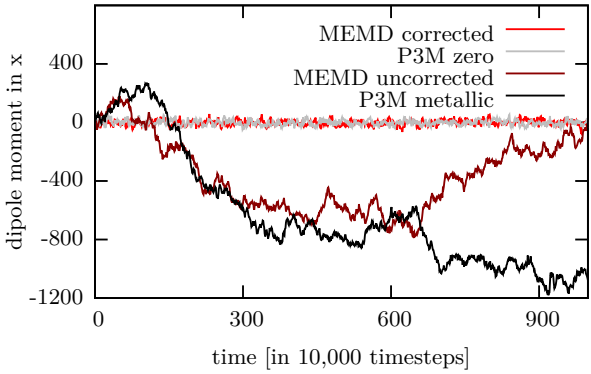
where $V$ is the volume of the simulation box, $\varepsilon_b$ is the boundary permittivity at infinite distance, and $\mathbf{P}$ is the total dipole moment of the unfolded coordinates of the charges. Similar corrections have been introduced in [156, 216] with additional moves to the original Monte Carlo algorithm, but were not transferred to the MD implementation.

In Ewald-based methods, the monopole and dipole terms can be calculated analytically and corrected directly in Fourier space, providing a simple and exact way to set metallic boundary conditions at infinity. This is not possible within the MEMD algorithm since it solves the Maxwell equations locally in real space. In addition, the contribution in MEMD will not only include the dipole moment contribution of the

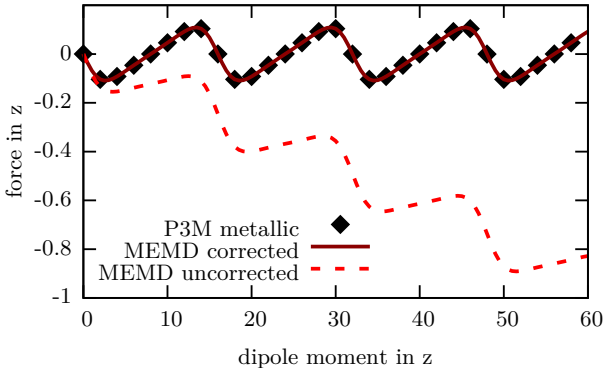
folded particle coordinates within the box geometry but also that of the unfolded coordinates, since the phase space history of the system is stored in the magnetic field. The correction is therefore done by calculating the dipole moment of all particles in unfolded coordinates and correcting the force on each particle by the resulting field.

For clarification, two simulations were set up, and the results can be seen in Figure 3.10. First, an electrolyte was simulated over a long time and the total dipole moment of the system was recorded for the original and the MEMD implementation with dipole correction, as well as P3M with metallic and $\varepsilon = 0$ boundaries (Figure 3.10a). For the corrected local algorithm and P3M with metallic boundaries, a random walk of the dipole moment can be observed, as expected for an unrestricted system. The uncorrected algorithm shows the same behavior as P3M with $\varepsilon = 0$ boundaries, forcing the dipole moment to remain around zero in a harmonic potential. Second, two charges of opposite sign are placed in a system and slowly pulled apart by an external field (Figure 3.10b). For the uncorrected algorithm, the oscillating force on one particle induced by the periodic images of the second particle is overshadowed by the influence of the resetting dipole force for the linearly increasing dipole moment of the system.

As can be seen in Figure 3.10a, this true 3D periodic behavior of MEMD relates to boundary conditions of $\varepsilon = 0$ at infinity in the P3M equivalent. This is an unwanted effect, and in a local real space method, there is no mathematical trick to apply metallic boundary conditions as in Ewald methods. The solution in our implementation is to calculate the system's dipole moment directly from the unfolded particle coordinates and subtract its influence from the force. This works reasonably well as long as there is no external driving force on the dipole moment. Systems with an additional external field or other means of introducing a net electric current will result in a drift in the dipole moment to a point where the dipole correction outweighs the actual force on the particles, and the algorithm breaks down. In these cases, the system has to be re-initialized quite often, and the bad scaling of the numerical relaxation leads to a slow performance. Therefore, systems with external driving forces on charged particles should be avoided for MEMD.




**(a)** Dipole moment fluctuations




**(b)** Force dependence on the dipole moment with and without dipole correction

**Figure 3.10.:** (a) The dipole moments of a simple electrolyte system are compared for the MEMD and a P3M algorithm. The original MEMD keeps the dipole moment at zero, which corresponds to P3M boundary conditions of $\varepsilon = 0$ at infinity. The corrected version shows the same behavior as P3M for metallic boundary conditions. (b) Two particles are dragged apart over several lengths of the simulation box via applying an external field. As the dipole moment of the system increases, these two boundary conditions also diverge quantitatively in the force calculation. The graphs have been published in [4].

### 3.2.6. Restrictions to the algorithm

After we have introduced and completed the MEMD algorithm in the last few sections, we want to mention some of the problems and restrictions we have encountered. The first and very obvious restriction is that because of a lack of error control, the algorithm can not be used for high precision electrostatic calculations. For biophysical simulations however, the resulting forces are acceptable and give good physical results, as can be seen in chapters 4, 5, and 6.

A second restriction is that simulations with a steadily increasing dipole moment, e. g. by application of an external electric field, will lead to problems with the dipole term correction as we have implemented it. It can and will be corrected by re-running the initial solver if the dipole moment of the system reaches a maximum value, but this will slow down the simulation significantly and should be kept in mind.

A third restriction that we will present in the following section 3.3 is that moving dielectric boundaries like walls can create numerical instabilities. If large magnetic field values or charges are "swallowed" by a sharp dielectric contrast, the simulation crashes or – even worse – will produce unphysical results. If the permittivity values in the system vary in time, the variations should be kept smooth and reasonable.

Not a real restriction, but more of a recommendation is that the MEMD algorithm performs very poorly in highly inhomogenous systems. The main computational effort is spent on the propagation of electromagnetic fields, while the optimal lattice size is mostly determined by the minimal distance between charges. This means that if there are cluttered charges in close vicinity to one another, the mesh has to be picked very fine, and the resulting fields have to be propagated using that same mesh through large areas of low charge density. Other approaches, like the P3M algorithm, can adapt to such problems more easily since their charge interpolation schemes are more complex and only have to be applied where charges are, and the mesh can also be chosen more coarsely for a wider charge interpolation.

## 3.3. Temporally varying dielectric properties

So far, we have not put any restrictions on the time propagation of the permittivity $\varepsilon(\mathbf{r}, t)$. In continuum theory, it can therefore be adjusted arbitrarily. However, this is not the case here, since we discretize the equation in space and time and introduce artificial jumps with this discretization. The same way charges are prohibited to move at arbitrary speeds in Molecular Dynamics simulation, some restrictions have to be enforced on the temporal adjustments of the dielectric permittivity.

Charges moving at high velocity in Molecular Dynamics simulations can cause problems, since they are able to overcome diverging potentials barriers within one time step, since the force acting on the particle stays finite. In the case of dielectric contrast in motion, the same effect can occur the other way around – a potential discontinuity is moving through a charge without exerting infinite force. In reality, a sharp dielectric boundary creates a force on any particle that can be calculated using an image charge convention: A particle with charge $q$ at distance $\mathbf{d}$ from the surface gives rise to a virtual image charge at distance $-\mathbf{d}$, therefore on the opposite side of the surface. This image carries the charge

$$q' = \frac{\varepsilon_1 - \varepsilon_2}{\varepsilon_1 + \varepsilon_2} q \quad , \tag{3.67}$$

where $\varepsilon_1$ is the permittivity in the domain containing the original particle, and $\varepsilon_2$ is the permittivity in the domain of the image charge.

Obviously, for non-zero permittivities, this charge will always have a finite value, and the distance between the real particle and the image charge is $2|\mathbf{d}|$. If the boundary moves in space towards the particle, this distance will approach zero and create a diverging force

$$\mathbf{F}_{\text{coul}} = q^2 \frac{(\varepsilon_1 - \varepsilon_2)}{(\varepsilon_1 + \varepsilon_2)} \frac{\hat{\mathbf{d}}}{(2\mathbf{d})^2} \xrightarrow{|\mathbf{d}| \to 0} \infty \tag{3.68}$$

even for weak dielectric discontinuities. Here, $\hat{\mathbf{d}}$ is the normal vector from the dielectric boundary to the particle.

Another problem arises if a dielectric boundary moves into an area where electromagnetic waves are present. The mathematical derivation is non-trivial, since our lattice provides a sort of "aether" for the

electromagnetic waves to move in at an absolute speed. This fixes the frame of reference and we can not simply apply a valid Lorentz transformation to a system where the boundary stands still. But a handwaving argument is that if the permittivity within a cell is increased on one lattice site, the electric field $\mathbf{E} = \mathbf{D}/\varepsilon$ will be dampened on this site and give rise to a rotation $\nabla \times \mathbf{E}$, which will in turn lead to an update of the magnetic field

$$\dot{\mathbf{B}} = -\nabla \times \mathbf{E}. \tag{3.69}$$

If these changes are too drastic and the temporal field update can not be carried away in the following propagation step, it will lead to instabilities in the algorithm.

It is therefore crucial that temporal changes in the permittivity distribution throughout the system are performed slowly and smoothly. Our implementation therefore contains a sanity check if a local permittivity is changed during the simulation that will not allow deviations by more than 10% from the current value to avoid unphysical behavior. This is directly linked to the according sanity check for particle velocities in ESPResSo, which checks that particles are not allowed to move more than 10% of a system cell, otherwise ESPResSo will stop the simulation with an error message.

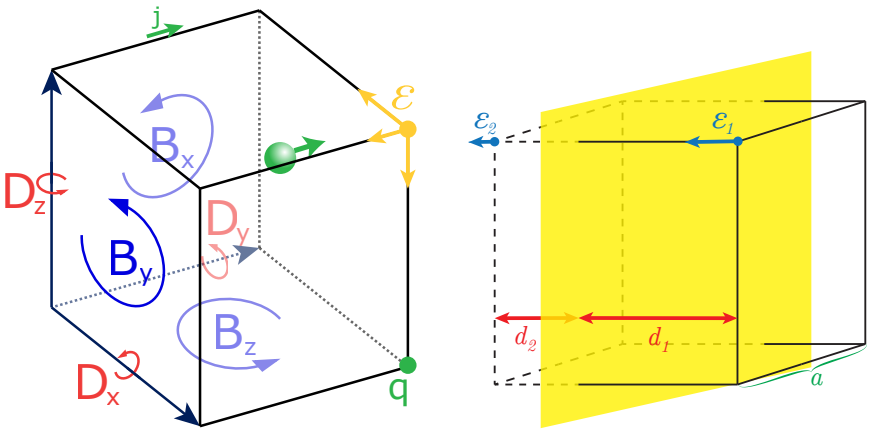
## 3.4. Implementation

The extended and completed MEMD algorithm is consistent and mathematically complete, as we have shown thus far in chapter 3. We have implemented it in the ESPResSo simulation package and the ScaFaCoS library, and we will present some assumptions and practices in this section. The goal is to help programmers who want to extend or reproduce the existing code base or implement the algorithm on their own.

### 3.4.1. Discretization

Obviously, we can not do simulations with fields in continuous spacetime. So a discretization in time and space is necessary. The first

of these does not pose any big problems and is done in a standard MD manner. For the latter space discretization, we have to construct a lattice in space that carries the fields. To update these fields, we also need to have electric currents on the lattice and therefore need to interpolate our charges on the grid. The representations of all variables, according to their differential forms, are shown in Figure 3.11a.




**(a)** Discretization **(b)** $\varepsilon$ interpolation

**Figure 3.11.:** (a) Discretization of the currents, fields, and permittivities onto a lattice cell. (b) Interpolation of dielectric permittivity values on the lattice. $\varepsilon(\mathbf{r})$ has a position and a direction (blue arrow). The values for $\varepsilon_1$ and $\varepsilon_2$ are determined and the value on the connecting link is set to the average value. If the gradient is too large, the value is determined by forming the harmonic average. The graphs have been published in [4].

The local permittivity values $\varepsilon$ can assume tensorial form, equivalent to a differential 2-form. In our implementation, we reduce the tensor to its diagonal entries (differential 1-form), which merely represents an optically isotropic medium. The local permittivity therefore has a value and a direction, and they are placed on the links of the interpolating grid. The electric displacement field values $\mathbf{D} = \varepsilon\mathbf{E}$ are still stored on the links, although they are represented graphically by a rotation around these links. This is due to their properties as differential forms, since $\mathbf{E}$ and $\varepsilon$ are both 1-forms and their product $\mathbf{D} = \varepsilon\mathbf{E}$

is hence a 2-form. As in the original algorithm, the magnetic fields **B** are placed on the lattice plaquettes, the electric currents **j** on the links, and the interpolated charges $q$ on the vertices. The representations are also apparent from equations (3.26) and (3.27), where **j**, **D** and **B** are connected via rotation. A short but slightly more in-depth description of differential forms in electromagnetism and the graphical representations in space can be found in appendix B.

To map given permittivity values, set by an interface or function, to the lattice, the finite difference between adjacent grid points is employed. If the difference is significantly bigger than the values, the link is marked as an *interface link*. The values for these interface links are then calculated by taking the harmonic average

$$\varepsilon_{\text{link}} = \varepsilon_1 \cdot \frac{d_2}{a} + \varepsilon_2 \cdot \frac{d_1}{a}, \tag{3.70}$$

where $\varepsilon_1$ and $\varepsilon_2$ are the permittivity values on the adjacent lattice sites on each side of the interface respectively, $d_1$ and $d_2$ are the distances of the according lattice site along the link to the interface, and $a$ is the lattice spacing, as depicted in Fig. 3.11b.

The definition of our gradient ($\nabla\cdot$) and curl ($\nabla\times$) operators is done canonically with finite differences (see Figure 3.12).

Following the depiction in Figure 3.12, the finite differences curl operators are defined as

$$\nabla \times \mathbf{D} = \mathbf{D}_1 + \mathbf{D}_2 + \mathbf{D}_3 + \mathbf{D}_4 \tag{3.71}$$

$$\nabla \times \mathbf{B} = \mathbf{B}_1 + \mathbf{B}_2 + \mathbf{B}_3 + \mathbf{B}_4 \quad , \tag{3.72}$$

with the order of rotation in mathematically negative sense, represented accordingly by negative values for vectors $\mathbf{D}_2$ and $\mathbf{D}_3$.

### 3.4.2. Parallelization

The ansatz for a parallelization of a local algorithm is very straight forward, it lends itself to the very common domain decomposition as described in section 2.6.2. Hereby, the system is divided into smaller spatial regions, each of which is subdivided into even smaller lattice

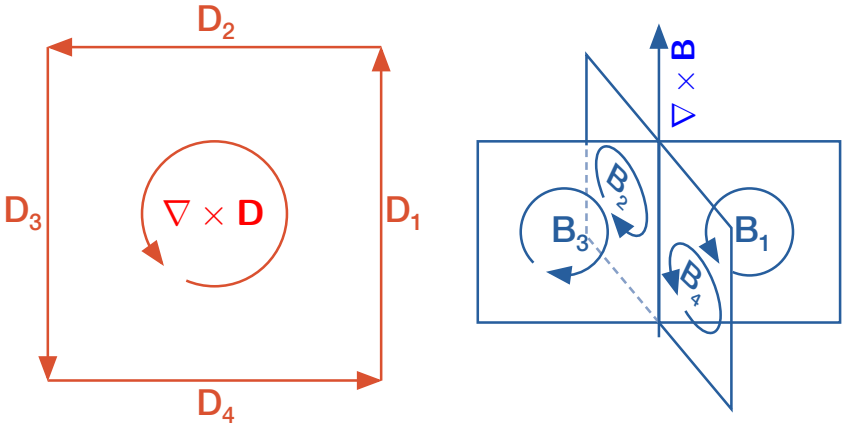



**Figure 3.12.:** Canonical definition of curl operators via finite differences. As dictated by the according differential forms, the **D**-fields are represented on lattice sites and the **B**-fields on lattice plaquettes.

cubes. For the MEMD, the domains boundaries all have to be multiples of the cell size, since the algorithm will only work on a cubic lattice. While an alternate non-cubic space discretization is possible for other algorithms, e. g. P3M, the curl operators ($\nabla\times$) can not be defined for such a setup.

The data exchange is implemented using the "Message Passing Interface" (MPI) standard and can be run on compute clusters running any implementation of this standard. Like it is commonly done in domain decomposition, each domain includes a layer of "ghost cells" which contain information on the first cell layer in the adjacent neighbor domain.

Communication between domains only has to be done for next neighbor regions, since no global information needs to be exchanged for a local algorithm. The domains will exchange a total number of 13 `double` variables per boundary cell, as depicted in Figure 3.13, in addition to the particle positions. The communication is implemented asynchronously, meaning that a processor will start updating particle positions and field values for all inner cells while communicating the outer cells and ghost cells.

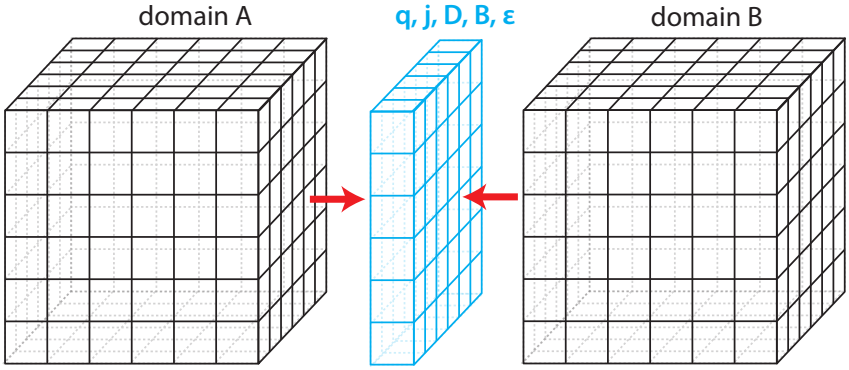



**Figure 3.13.:** Two adjacent domains have to exchange information on charges $q$, currents $\mathbf{j}$, fields $\mathbf{D}$ and $\mathbf{B}$, and permittivity values $\varepsilon$ at each time step.

On paper, this reads like a very effective parallelization, and it does scale fairly well as shown in section 5.3. However, two things need to be considered: First, the amount of boundary cells in usual simulations is quite significant. A typical system in Molecular Dynamics simulations will contain around 1000 lattice cells, for cubic domain decomposition this is a $10 \times 10 \times 10$ distribution. Of those 1000 cells, 48.8% would be boundary cells, so almost half the memory used by MEMD has to be communicated. Second, the amount of data exchanged between domains is large. It not only contains particle positions like P3M in comparison, but all currents, fields, and permittivities. During our work, we have updated the MPI communication functions from the deprecated `MPI_send` and `MPI_recv` to the bidirectional `MPI_sendrecv`, which reduced the MPI lag time by almost an order of magnitude. For the P3M algorithm, this made a big difference in communication time and scaling, whereas for MEMD, we merely saw a 5% speedup. While this is not an insignificant speedup, it clearly shows that only around 5% of the communication delay is due to lag, and 95% is actually transferring the data.

One more thing to consider for the implementation is the inclusion of boundary conditions. Most commonly in Molecular Dynamics simulations, the simulation cell is replicated periodically in all directions to

avoid boundary effects and allow particles to simply pass through the boundary. This poses a problem for long-range interactions, since it requires an infinite sum of periodically repeated particles. In the case of a local algorithm like MEMD, however, periodic boundary conditions are easily applied by simply connecting the edge of a simulation box to the oppositely positioned edge. This allows particles as well as fields to simply move through the boundary and "reappear" on the other. There is, however, a difference between this periodic replication and a real three-dimensional torus. These differences are handled in other algorithms like P3M by means of boundary conditions for the electrostatic potential at infinite distance, and have to be corrected in any other algorithm as well. We focus more on this in section 3.2.5.

### 3.4.3. Integrators, ScaFaCoS and ESPResSo

We have implemented and verified our extended MEMD algorithm in two different software platforms: ESPResSo is a Molecular Dynamics software package that features a wide range of short-range and long-range interactions and provides integrators as well as analysis functions, all via an easy to use TCL scripting interface. Besides our MEMD implementation, it contains a well established P3M algorithm with the ICC$\star$ (Induced Charge Calculation) extension. This extension allows to put surface charges on dielectric boundaries that will iteratively adapt to any electric field present [250]. This way, we can include dielectric enclosures in P3M simulations and easily compare them to MEMD results using the exact same system setup and analysis functions.

The implementation of the second software, ScaFaCoS, a "Scalable Fast Coulomb Solver", was the result of a BMBF (Bundesministerium für Bildung und Forschung) project. Physicists, mathematicians, and computer scientists from different German universities worked together to create a library for `C`, `C++`, and `FORTRAN` that focuses solely on calculating long-range interactions in particle simulations. This library can be included in simulation codes that are not yet able to calculate long-range interactions, and it was of great service to us, since it provided a good way to compare not only the results of MEMD but its serial and parallel performance against other, more established numerical solvers.

For an example on how to use the algorithm within the ScaFaCoS library, we refer to the user manual [291] and the example code included in the library download. A short conceptual script to use with ESPResSo can be found in appendix A.

The software package ESPResSo, like all Molecular Dynamics codes, features a time-reversible integrator, Velocity-Verlet is hereby most commonly used. It is important that the MEMD algorithm is embedded into this integration scheme in a way that maintains the time-reversibility, and therefore the energy conservation of the numeric integration. The straight forward way to first update the fields and then calculate the forces breaks this symmetry and can therefore not be used. We need to split up the update process of the fields to propagate by half a time step before the force calculation and half a time step afterwards. An analog to the Verlet integrator would be

1. Update the particle momenta by half a time step.
2. Update the A field by half a time step.
3. Update the particle positions by half a time step.
4. Update the electric field by a full time step.
5. Update the particle positions by half a time step.
6. Update the A field by half a time step.
7. Update the particle momenta by half a time step.

This integration scheme is time-reversible. However, the updates of $\mathbf{A}$ and the particle momenta are interchangeable, since we do not include the magnetic part of the Lorentz force in our implementation. We can thus create an equivalent integration scheme that only requires one field propagation per time step

1. Update the particle momenta by half a time step.
2. Update the A field by a full time step, from $(t-\delta t/2)$ to $(t+\delta t/2)$.

3. Update the particle positions by half a time step.

4. Update the electric field by a *full* time step.

5. Update the particle positions by half a time step.

6. Update the particle momenta by half a time step.

This scheme is still energy conserving, but has a conceptual problem. The electric field update (step 4) is calculated for a particle configuration which has only progressed by half a time step. This would mean that the constraint, the Gauss law (3.14), is only satisfied within the accuracy of the time discretization. It is therefore necessary to apply a third and further simplified integration scheme.

1. Update the particle momenta by half a time step.

2. Update the A field by a full time step, from $(t-\delta t/2)$ to $(t+\delta t/2)$.

3. Update the particle positions by a *full* time step.

4. Update the electric field by a full time step.

5. Update the particle momenta by half a time step.

This scheme is still time-reversible and has all desired features that the first two schemes lacked. This is how we have implemented the algorithm in ESPResSo. For the ScaFaCoS library, we were not able to control the integration scheme in the same way. Here, we implemented the following force calculation scheme.

1. Update the **A**-fields by half a time step.

2. Update the electric field by a full time step.

3. Update the **A**-fields by half a time step.

This guarantees that externally, if a time-reversible integration scheme is used, the energy is conserved. It does, however, require two field propagation steps, which is computationally more costly.

# 4. Verification

In chapter 3, we presented how we have extended and reimplemented an algorithm for electrostatic interactions in particle simulations. It is imperative that a novel implementation of an algorithm that has never been used this way before needs to be thoroughly tested and verified to produce correct results. We have done this with a countless number of analytically solvable or well known physical systems as well as in comparison to more established algorithmic approaches. Some of the early tests led to corrections we presented in section 3.2, but the majority confirmed that the algorithm is stable and provides correct results within the given error boundaries.

In this chapter, we will present a small sample of said test cases that we feel represents a valid and concise verification of our algorithmic approach. While some other comparative calculations have been created within the ScaFaCoS library, the simulations shown here were carried out using the ESPResSo software package. This is mostly because our key addition to the algorithm, the inclusion of spatially varying dielectric properties, is not present in any other algorithm within the ScaFaCoS library, and its interface does not easily allow addressing it at all. But all test cases we simulated in ScaFaCoS with a constant dielectric background gave the same resulting forces as all other methods, including our proof of concept implementations of a simple Ewald summation and a direct summation solver.

## 4.1. Distinction between initial and dynamic algorithm

The MEMD algorithm essentially consists of two different parts: The initial solution presented in section 3.2.1, and the dynamic propagation

of the system. We have made changes to both those parts and additionally implemented the option for varying permittivity values, so we need to test both algorithmic parts exhaustively. In order to keep this section brief, we excluded the tests for constant dielectric background, since the varying dielectric background is the novel part and is what we will use.

This leaves us with four systems to verify: The initial and the dynamic solver for a system including sharp dielectric contrasts, as well as for a smoothly varying dielectric permittivity accessible to moving charges. The former situation will be compared against analytical solutions as well as results from the ICC$\star$ and ICMMM2D algorithms that can simulate sharp dielectric interfaces. The latter situation with smoothly varying permittivity values is a novel possibility for which we first had no other algorithm available for comparison. The initial solver for this case can, however, be compared to an analytical solution, and for the dynamic part, Zhenli Xu, in collaboration with us, developed the HIM method that will be explained in section 6.3.3 and is described in our joint publication [5]. The HIM algorithm is implemented in a Monte Carlo code and can simulate a radially symmetric distribution of smoothly varying dielectric permittivity, as long as the distribution follows a piece-wise harmonic function. This allowed us to confirm the dynamic propagation of a system with smoothly varying dielectric properties against another algorithm.

## 4.2. Verification of the initial solver

The scheme used for all field calculations per definition follows the Gauss law exactly (see figure 3.2), and the algorithm iterates to a minimum electric field energy. This means that the maximum error in the electrostatic potential is used as exit condition and will therefore be satisfied per definition. Nonetheless it is important to test its validity, since there can still be unexpected drifts or numerical errors, as the directional correction introduced in section 3.2.4 demonstrates.

### 4.2.1. Analytical solutions

A first and very simple setup would be that of a single particle with charge $q$ in front of an infinite dielectric wall. This case can be solved analytically with one virtual mirror charge. The mirror charge $q'$ will be scaled by a prefactor

$$q' = \frac{\varepsilon_1 - \varepsilon_2}{\varepsilon_1 + \varepsilon_2} q \quad , \tag{4.1}$$

where $\varepsilon_1$ is the permittivity in the domain containing the original particle, and $\varepsilon_2$ is the permittivity in the domain of the image charge.

But since the MEMD algorithm is only implemented for periodic boundary conditions in all three dimensions, the next best symmetrical setup is a particle between two infinite dielectric walls. The slit between walls will be repeated infinitely many times, but this infinite sum can still be solved in Fourier space. It is, however, easy to show that for $\varepsilon_2 \gg 1$, the contributions of box copies decays faster than $1/(\varepsilon_2 L)$, where $L$ is the box length, and the difference between the correct solution and a solution with only two image charges for our setup is less than 0.01% and not noticeable in a graph. The results of the theoretical prediction and the MEMD initial solver, as well as a sketch of the setup, are presented in figure 4.1.

The results agree to an accuracy of $10^{-7}$ in the resulting force in simulation units. This is very impressive, since this places the maximum electric field energy error at around $10^{-14}$, the maximum number of available digits in `double` floating point variables, and for these simulations we set the exit condition for the successive over relaxation to a field error of $\delta\mathbf{D} < 10^{-13}$, an order of magnitude higher.

Another benefit of choosing this system for the verification is that it is non-neutral and we can control that the monopole moment, i. e. the total charge, of the system correctly. While non-neutral systems in periodic boundary conditions are unphysical, they can occur in simulations, for example when particles are added or removed, and the algorithm should correct this by shifting the potential to $\Phi \equiv 0$ at infinity. In Ewald methods like P3M, the neutrality condition is enforced by adding a neutrality background term.

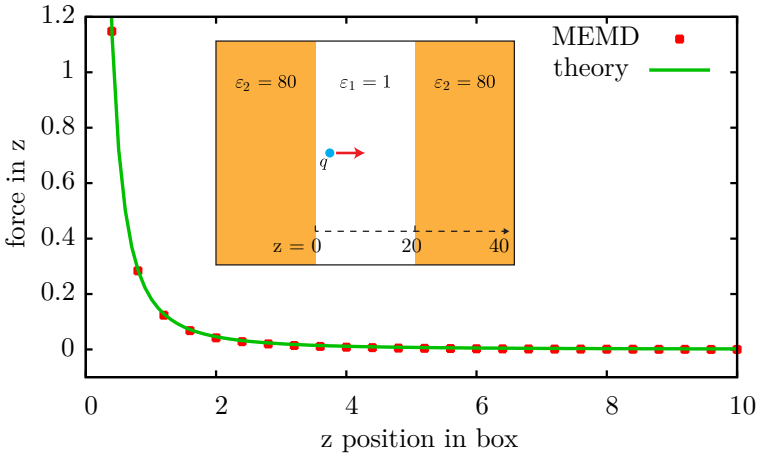



**Figure 4.1.:** A single particle is placed between two thick dielectric walls in $z$-direction. We call the initial solver for different positions in $z$ and compared to the analytical solution for an infinitely thick wall.

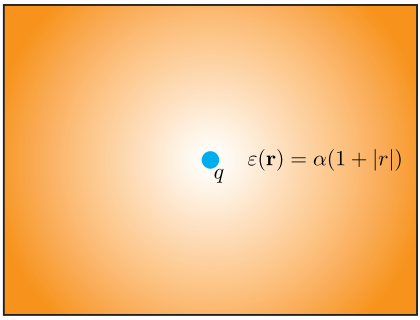



**Figure 4.2.:** A single charge is placed inside a box with a linearly increasing dielectric permittivity $\varepsilon(\mathbf{r}) = \alpha(1 + |\mathbf{r}|)$. The analytical solution predicts a potential scaling with $1/r^3$, since in addition to the regular $1/r^2$, the electric $\mathbf{E} = \mathbf{D}/\varepsilon$ field is dampened by the dielectric permittivity.

To validate the initial solution for spatially varying dielectric media, the field is compared to the analytical solution in a system that consists of a single charge in the box center and a dielectric background that is $\varepsilon = 1$ at the charge position and linearly rises with distance ($\varepsilon(r) = |r|$). A sketch of this setup is shown in figure 4.2 and it can be solved analytically using direct spherical integration, yielding

$$\phi_{\text{analytic}} = \frac{1}{1+\alpha} \cdot q \frac{1}{4\pi\epsilon_0 r^3} + C, \tag{4.2}$$

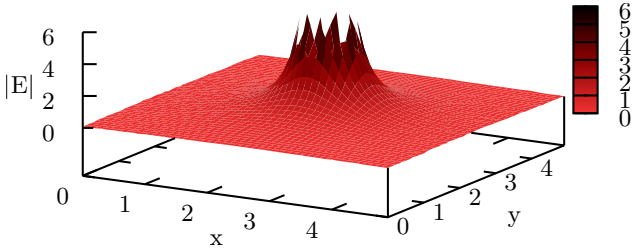
where $C$ is an integration constant that can be set so the potential vanishes at infinite distance.

The result can be seen in figure 4.3a and matches the analytic prediction. The relative RMS error of the absolute field value stays well below the algorithmic precision limit of $10^{-3}$, which is not surprising since the error contribution of the retarded solutions due to the dynamic algorithm is not present yet. While the absolute error shows no spatial preference, the relative error shown in figure 4.3b is very small in the cell center and increases near the boundaries. This behavior shows that the absolute error is independent of the local field strength and only influenced by the lattice spacing.

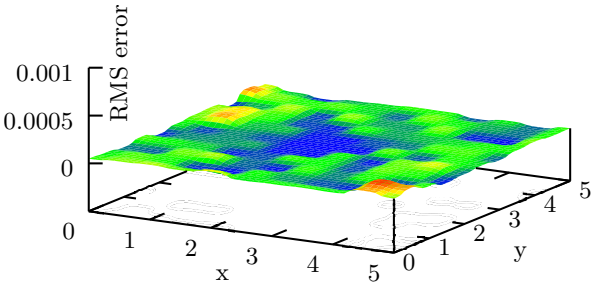
### 4.2.2. Verification against other available algorithms

Comparing the initial solver to other available methods is fairly simple, since for the first time step, the particle positions will be exactly the same and the forces can be directly compared. Something similar is much more difficult for the dynamic propagation, since following the Lyapunov exponent, even very small deviations will result in entirely different paths of motion.

A system that can be calculated with MEMD, ICC$\star$ and MMM2D alike is depicted in figure 4.4. As shown in section 2.5.2, the dielectric walls will significantly influence the forces on the particles. For such a system, we reliably get an RMS force error of less than $10^{-3}$ for MEMD when compared to high precision simulations with the other two algorithms.




**(a)** resulting $1/r^3$ potential behavior




**(b)** root mean square field error

**Figure 4.3.:** The numerical results of the initial solver for a charge in the center of a linearly increasing dielectric background permittivity. As expected, the result follows a $1/r^3$ characteristic. The relative numerical error is shown in Figure (b). The graphs have been published in [4].

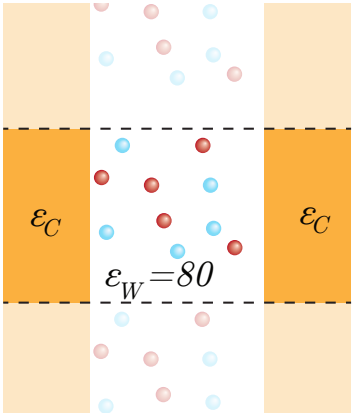



**Figure 4.4.:** An electrolyte is placed between two dielectric walls. This system can be simulated with MEMD, ICC⋆ and ICMMM2D to compare the force results. The sketch has been published in [3].

## 4.3. Verification of the dynamic solver

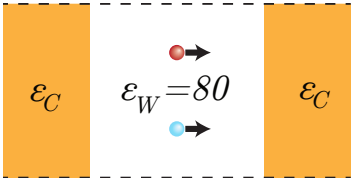
In contrast to the initial solver, one can not simply compare the forces on particles to check the correct dynamic propagation of a system. Even slight deviations will result in very different paths for each particle and two almost identical simulations are not comparable anymore. Thus we need to look at static observables of a dynamic system in equilibrium.

### 4.3.1. Analytical solutions

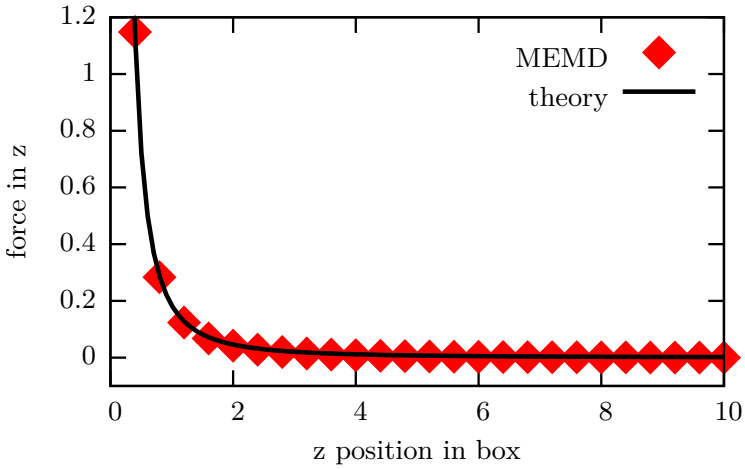
Since analytical solutions for observables in systems with many particles and dielectric enclosures are typically based on mean field theories like Poisson-Boltzmann, they can not provide a theoretical prediction with which we can confidently compare our dynamic solver.

In order to validate the dynamic behavior of the extended algorithm, we used a trick that a script-based simulation software like ESPResSo allows: We placed two particles with charges $q_1 = 1$ and $q_2 = -1$ in water with $\varepsilon_W = 80$ between two dielectric walls with $\varepsilon_C = 1$ in $z$-direction and manually coupled their motion so they moved in a per-

fectly synchronous way. We restricted the particle coordinates to the are between the two walls and started the particles at half the box length $L/2$ distance in $x$-direction, but motion in $x$, $y$, and $z$ was performed randomly. In this artificial case, the force on the particles can be theoretically predicted in every position and compared to the simulation. The results can be seen in figure 4.5 and match the analytical prediction up to algorithmic precision.




**(a)** schematic




**(b)** resulting force

**Figure 4.5.:** A particle is pulled through the simulation box with an infinite dielectric wall. The results are compared to the analytical result.

An additional benefit of this setup is that we can ensure that the force in $x$-direction remains zero as enforced by the system's symmetry. This means that the non-zero dipole moment of the system is corrected and

does not contribute to the forces, in agreement with the assumption of metallic boundary conditions at infinity.

### 4.3.2. Other available methods

#### Induced Charge Computation (ICC⋆)

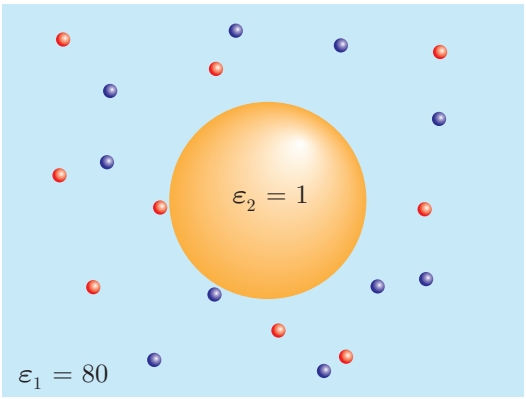
Within the ESPResSo software package, only one other algorithm, ICC⋆, allows for non-planar dielectric boundaries in the system. Therefore, to ensure that this type of dielectric permittivity distribution is represented correctly in the MEMD algorithm, we chose to present such a setup for a good completion of our verification.

It is a less conceptional and more physical simulation to verify the equilibrium observables with a fixed colloid (charge $Q_c = 60$, radius $R_c = 10$ in MD units, hard sphere boundary) in a solution of counterions and salt (concentration $c = 50$mMol), as depicted in figure 4.6a. The radial distribution of the ions around the colloid is measured and compared against the results of a Monte Carlo simulation from [134] and an MD simulation using the ICC⋆ algorithm for sharp dielectric boundaries [250, 255]. The data is shown in figure 4.6b and matches very well, but shows slight deviation at short distances from the boundary. This is due to the linear lattice interpolation of charges which leads to inaccuracy where charges are very close to each other, at the order of one lattice spacing.

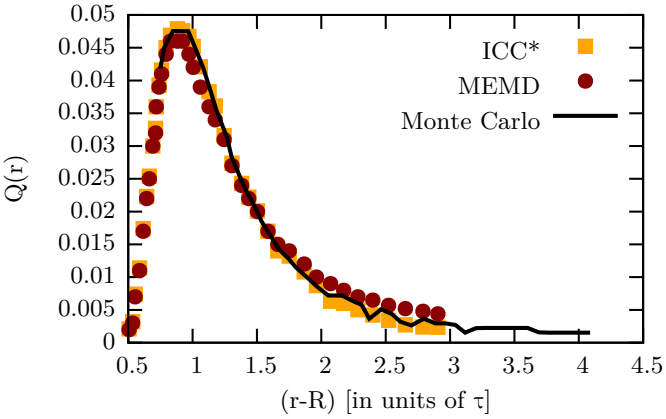
#### Harmonic Interpolation Method (HIM)

The only other method we know of that can calculate electrostatic interactions with a smoothly varying dielectric background into which charges can move freely, is the Harmonic Interpolation Method that was developed by Zhenli Xu during his visit at our institute for the purpose of collaborating with us. It is restricted to piece-wise harmonic functions and radially symmetric geometry, which led us to our next model for the MEMD verification.

An example where such smoothly varying changes in $\varepsilon$ can play a substantial role is the simulation of a charged colloid in salt water suspension (figure 4.7a). A first approach at dielectric coarse-graining is




**(a)** schematic

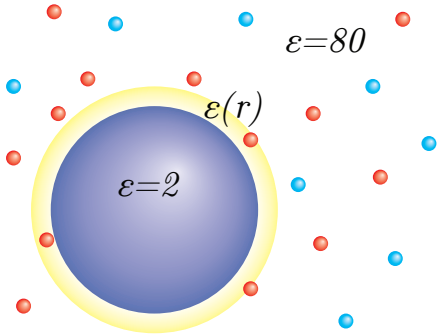



**(b)** resulting force

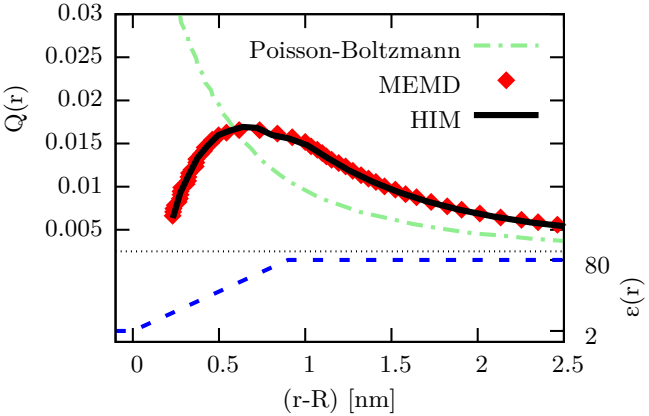
**Figure 4.6.:** A dielectric sphere is placed in the center of the box and the ions assemble around it. The figure shows the ion density as a function of the radial distance.

e. g. the application of the ICC$\star$ method. With means of this algorithm, a system can be simulated with dielectric permittivity $\varepsilon = 2$ within the colloid and $\varepsilon = 80$ for the surrounding implicit water. However, realistically the polarizability and therefore bulk permittivity of water close to charged surfaces and around high ion concentrations is generally reduced and varies greatly [261]. Many workarounds were proposed to address this behavior, including the introduction of an artificial Stern layer [35] to reproduce the desired Gouy-Chapman predictions. A more direct and more physical approach is to interpolate the bulk permittivity from the colloid to free water in first order (see bottom of figure 4.7b). This matches the results of atomistic simulations [277] sufficiently well.

The bulk permittivity within the colloid is $\varepsilon_C = 2$ and in free water $\varepsilon_W = 80$, as depicted in figure 4.7a. Between the two regions, it is interpolated linearly $\varepsilon(|r|) = \varepsilon_C + (\varepsilon_W - \varepsilon_C)/(2\tau) \cdot (r - R)$ over the distance of 2 ion diameters $d = 2\sigma$. This system is simulated with the MEMD algorithm in ESPResSo and HIM in a seperate Monte Carlo simulation code. Figure 4.7b shows the radial charge density of the counterions around a colloid of charge $Z = 60$ with radius $R_c = 30\sigma$ in a monovalent salt solvent of $c = 50$mMol concentration. This yields a very good confirmation for our algorithm, since the simulations were carried out not only by two different algorithms, but by two entirely different codes and simulational approaches, Molecular Dynamics and Monte Carlo. The fact that the results match so well gives us confidence about the validity and applicability of the extended MEMD algorithm.




**(a)** Schematic of the colloidal setup




**(b)** Radial charge density profile

**Figure 4.7.:** A charged colloid (charge $Z = 60$, radius $R = 30$) is suspended in a salt solution (concentration $c = 50$mMol). With the MEMD and the HIM algorithm, the dielectric constant is modeled to radially increase linearly within 2 ion diameters between the two regimes.
The resulting radial charge density profiles are shown on the right, and the two entirely different algorithms produce exactly matching radial density profiles. Sketch (a) has been published in [5].

## Summary

We have tested the extended MEMD algorithm in various ways, and just with the few artificial and physical systems we have presented here, we verified that:

1. MEMD correctly handles monopole and dipole moments in the simulation box.
2. The initial solver reproduces analytical solutions for sharp dielectric contrasts as well as smoothly varying dielectric backgrounds.
3. In direct comparison to the ICC$\star$ and ICMMM2D algorithms, MEMD calculates the exact same static forces using the initial solver.
4. The dynamic propagation calculates forces correctly for particles being artificially pulled through the simulation box.
5. For static observables at non-planar dielectric interfaces, MEMD gives the same results as the more established ICC$\star$ algorithm and a Monte Carlo simulation.
6. For smoothly varying and freely accessible dielectric backgrounds, MEMD agrees to an astounding degree with the HIM method and simulations of the same system carried out in a separate Monte Carlo code.

Both adapted algorithms (initial and dynamics) match the established methods and theoretical predictions to the expected precision. As long as the systematic limitations are kept in mind, the algorithm produces valid results.

# 5. Comparison with other methods

The MEMD algorithm has been introduced, extended and verified. Now, it needs to be compared against other available electrostatics methods. This includes its theoretically possible and practically implemented features as well as the computational performance.

## 5.1. Capability of the algorithm

The key values an electrostatics algorithm can contribute to a Molecular Dynamics simulation are the resulting force on each particle, and the electrostatic pressure tensor of the system. Additionally, the system can feature a periodicity in 0 (non-periodic), 1 or 2 (partially periodic), or all 3 dimensions (3D-periodic). For the performance, the abilities to tune the accuracy to a given value and to delegate the calculation of the near-field contribution to the standard routine for short-range interaction in the simulation software are of special interest.

In table 5.1, we present a short comparison of the most commonly available electrostatics methods. The features of each method are shown as implemented and usable in either ESPResSo or ScaFaCoS (+), theoretically possible but not implemented in these solvers (?), and not available because of general restrictions of the algorithm (–). Here, the multigrid solvers VMG and PP3MG, as well as the algorithms based on Fast Fourier Transform (FFT), P3M and $P^2$NFFT, were combined into one method. We included the reference implementations of a direct sum with nearest neighbor cutoff and $\mathcal{O}(N^2)$ scaling, and a standard Ewald summation. The other available methods are the Fast Multipole Method (FMM), the MMM*D family of algorithms, and MEMD. A short description of all these methods will be presented in section 5.3.1.

| | FMM | MEMD | MMM*D | FFT-based | multigrid | direct | Ewald |
|---|---|---|---|---|---|---|---|
| **3D-periodic** | + | + | – | + | + | + | + |
| **non-cubic, cuboid** | – | + | | + | + | + | + |
| **non-cubic, non-cuboid** | ? | – | | + | ? | + | ? |
| **Virials** | ? | +[a] | | + | +[a] | + | – |
| **Nonperiodic** | + | ? | – | + | ? | + | – |
| **non-cubic** | – | ? | | + | ? | + | |
| **Virials** | + | ? | | + | ? | + | |
| **Partially periodic** | + | ? | + | + | ? | + | – |
| **Tunable accuracy** | + | +[b] | + | + | ? | – | + |
| **Delegate near-field** | – | – | – | + | ? | – | – |

[a] scalar
[b] up to $10^{-4}$

**Table 5.1.:** Overview of the features of different electrostatics solvers. "+" means that the feature is available and implemented either in ESPResSo or ScaFaCoS, "?" means that the feature is conceptually possible but not implemented in our codes, and "–" means that the feature is not available due to algorithmic restrictions. Hereby, *Virials* refers to the mechanical stress on an atomic scale and can have tensorial form.

Taking a closer look at the feature matrix in table 5.1, we see that MEMD is theoretically capable of mixed and open boundary conditions. However, for the implementation we began, this would be very difficult to include. One can of course simply not transfer propagating fields periodically in a torus geometry, but the energy and momentum carried away by those fields will diffuse from the system, and with any open boundary condition, energy and momentum conservation will not hold anymore. It also was not of high interest for our research purposes, therefore we did not implement this possibility. The geometry of the system also follows some restrictions. Since the internal lattice sites have to be cubic for our finite-differences representations of the curl-operator ($\nabla\times$), MEMD can not include triclinic systems, and the cuboid size is restricted so both sides can be multiples of the same lattice constant. The FFT-based methods are the only ones that can delegate the calculation of the near-field interaction, since they have a well-defined cut-off radius in real space, which is a feature that other methods lack.

### 5.1.1. Precision

One of the biggest issues of the MEMD algorithm is its restriction in the accuracy of the calculation. While for many systems an RMS force error of $10^{-3}$ is sufficient, there are certainly simulations that require a more precise electrostatics solver. More importantly, a major part of this numerical error occurs at very short distances between particles, due to the linear interpolation scheme. The forces at such short distances are not just very high but can have a significant influence on the dynamics of a system.

For typical Molecular Dynamics simulations where most particles have charges of 1 or 2 in simulation units and will be restricted in the minimum distance by a Lennard-Jones or WCA potential, our verifications and tests have shown no systematic deviations. However, for systems with highly charged particles – e. g. multivalent ions – or with very small particles distances – e. g. a Coulomb explosion – one should carefully study if such a problem arises.

To increase the precision while keeping an acceptable computational

effort, a good adaptation is the charge interpolation on several lattice sites. This will increase the accuracy for short-range interactions and will even allow for coarser meshes. Since this directly influences the main computational effort of the algorithm, it will certainly speed up the simulation for a given accuracy. However, for our purposes of locally varying dielectric permittivity, a higher order charge interpolation is not possible.

### 5.1.2. Varying dielectric properties

Of the methods included in the feature matrix in table 5.1, the MMM*D family of algorithms can simulate planar dielectric interfaces in the non-periodic dimensions. This is done with an infinite but convergent sum over scaled mirror charges. The method names are then extended by a preceding IC (Image Charge) acronym.

Arbitrarily shaped sharp dielectric interfaces can be simulated with the ICC$\star$ algorithm. Conceptually, ICC$\star$ can be attached to any of the above electrostatics solvers, since it merely adds self-consistent surface charges on the dielectric interface to the system and does not change the algorithm structurally. However, for algorithm like FMM or other tree-based methods, not having a domain decomposition but instead a Morton $Z$-order curve will significantly increase the communication overhead for the iteration steps. In ESPResSo, the ICC$\star$ algorithm is used with P3M and is therefore named ICCP3M.

Computationally, the additional effort to include dielectric interfaces in an MMM*D calculation is not significant, since the same code is used for corrections to the 3D-periodic solution. This method is, however, restricted to planar dielectric surfaces in the non-periodic dimensions and can not be used for any other dielectric inclusions. The addition of the ICC$\star$ algorithm to an electrostatics method increases the computing time by a prefactor. The number of charges $N$ in the system is increased by the amount of surface charges $M$, which in case of P3M

with a scaling of $\mathcal{O}(N \log(N))$ creates a prefactor of

$$\frac{(M+1)}{N}\frac{\log(M+N)}{\log(N)} = \left(\frac{M}{N} + \frac{1}{N}\right)\log(M) \approx \frac{M}{N}\log(M) \tag{5.1}$$

This means that for systems with long dielectric interfaces and/or few moving charged particles, the computational cost can be significant. In addition to this prefactor, ICC$\star$ will iterate the correct surface charges, and requires the full electrostatic solution of the system for each iteration step. Usually, the algorithm takes between one and three iterations to converge, which means that in addition to $M/N$, the computational cost in increased by a factor of 1 to 3 times $M/N$.

In theory, multigrid methods like VMG or PP3MG are able to solve arbitrary differential equations, not just the Poisson equation, and can be modified to include locally varying dielectric permittivity as well. Since this has not been implemented in ScaFaCoS or ESPResSo, it is not easy to quantify the amount of additional computational effort for changing the underlying partial differential equation.

The inclusion of varying dielectric permittivity is the big advantage MEMD has in comparison to the other presented methods. Not only is it possible to include arbitrarily shaped dielectric interfaces and even smooth changes, but there is no additional computational cost to the algorithm whatsoever. That is why in this thesis, we mostly focus on the algorithmic extension for dielectric permittivity and not on improvements to the accuracy and speed by means of nonlinear interpolation or more effective corrections for the self energy interaction.

### 5.1.3. Potential improvements

As mentioned above, the accuracy and speed of MEMD can theoretically be significantly increased by using a higher order interpolation scheme. But this sacrifices the capability for locally varying dielectric properties. And even with the proof of concept implementation we did for a spline interpolation scheme, MEMD will not reach the accuracy

and speed of competing algorithms like P3M, and in our opinion, this would not be advisable.

The ease of usability for the algorithm remains something to be desired. In its current state, the algorithm can become unstable or unreliable if unphysical parameters are used. Our implementation performs a set of sanity checks and will warn the user if it foresees problems. But since the propagation of the system is not known at this point, there is no guarantee.

A very welcome improvement of the algorithm would most likely be a transfer of the code base to a GPU architecture. Structurally, the MEMD is very similar to a D3 Lattice-Boltzmann implementation, and this has been shown to gain a factor of 40 or more when transferred to GPU architectures [286]. In the current state of graphics chips, the system size would be restricted by the maximum amount of available memory, but this maximum is steadily increased and would already be enough for most simulations performed for this thesis, currently $\approx 256^3$ lattice points.

When it comes to performance, the initial solver does not compete against more advanced algorithms, and requires a large amount of computational power for this very first solution. Certainly, especially for larger systems since the algorithm also does not scale very well, the initial solution can be improved in many ways. If the system does not contain dielectric variations, our implementation can call a high precision result from the P3M solver to start the simulation. For varying permittivity, one would have to connect the algorithm to another method that has this capability.

## 5.2. Serial performance

Even today in Molecular Dynamics simulations, the calculation of long-range interactions like electrostatics and hydrodynamics uses up to 80% of the computing power each simulation step. At the same time, advancements in computer technology and software allow for larger and larger systems to be simulated. Therefore, the scaling and performance of an electrostatic algorithm is crucial for its applicability and success.

The performance of the MEMD algorithm again consists of two parts: The convergence of the initial solver and the computational cost of the dynamic propagation each following time step. The initial solver scales almost cubicly $\mathcal{O}(N^3)$ with the number of charges, but in most systems it only has to be called once. So for a simulation that does not run too short, the dynamic algorithm is the more relevant part to judge the performance of the algorithm.

### 5.2.1. Tuning the implementation

In theory, a purely local algorithm such as MEMD should require a fixed amount of floating point operations for each lattice cell. In a system with approximately homogenous density, this leads to a linear scaling $\mathcal{O}(N)$ of the computational effort with the number of charges $N$. This can certainly never be achieved perfectly in an implementation, and we want to check how close we can get.

A big part of the performance in MEMD is the choice of an appropriate parameter set. For our timings, we always tuned the simulation to find the error minimum und run simulations with those parameters. The use of our very basic tuning algorithm provided very stable and consistent results, and the scaling up to 2 000 000 particles is shown in figure 5.1.

For comparison, we added a line with a slope of 1, which represents perfectly linear scaling. While we can see slight deviations, the algorithm holds up pretty well. We chose a water/salt solution of 500 mM concentration for our benchmarks. The simulations were run with ESPResSo on a Core2Duo CPU with 2.1 GHz clock speed, and 8 GB of DDR3 memory. While the dynamic solver for even the largest simulations with several $10^6$ charges took a maximum of less than 4 seconds per MD time step, the scaling of the initial solver is much worse and the initial solution for 2 000 000 particles took more than 24 hours CPU time.

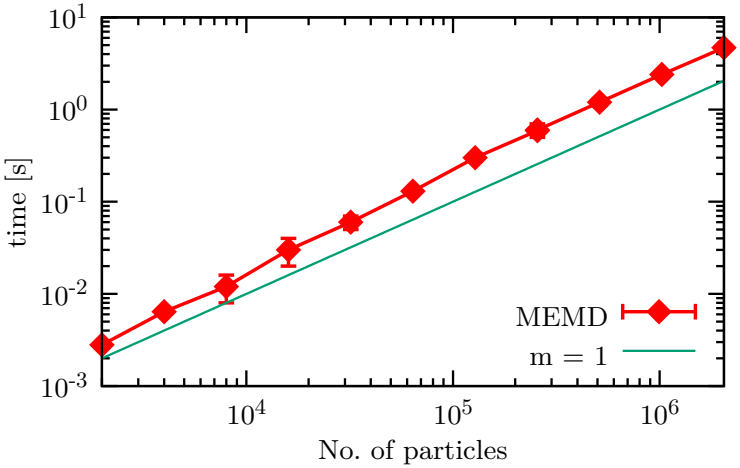



**Figure 5.1.:** Using the automatic tuning routine, we ran simulations for a maximum RMS force error of $10^{-3}$ on a single core. We measured the CPU time per MD time step, and it seems to follow a linear scaling law. Note that while the absolute time even for several million particles on one processor core never exceeded a few seconds, it took a whole day to obtain the initial solution for this system.

### 5.2.2. Comparison to other methods

While we will compare the parallel performance – and with it of course the performance on just one node – in section 5.3, we want to compare the single CPU scaling of MEMD and P3M separately. The benchmarks were carried out using the same particle distribution and the same CPU as before, and the results of our comparison can be seen in figure 5.2. The accuracy was fixed at $10^{-3}$, which means that with our tuning, the grid size of MEMD scaled as $N^{1/3}$ with the particle number.

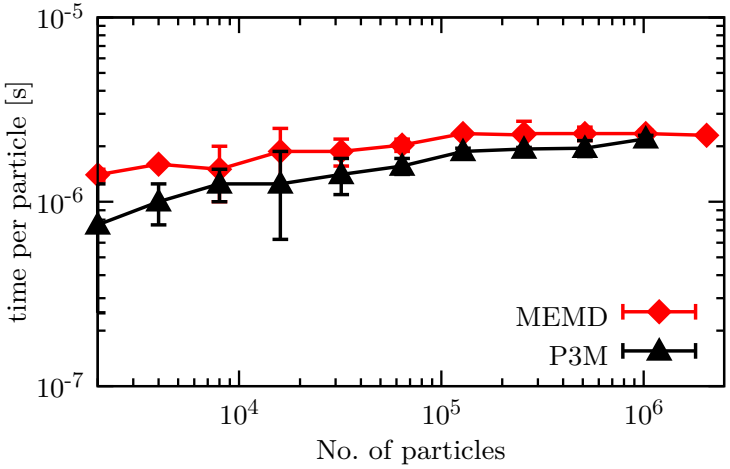



**Figure 5.2.:** Scaling comparison to P3M. Throughout the test, MEMD is slower than P3M, although never by more than a factor of two. For perfectly linear scaling, the CPU time per particle would remain constant. Both algorithms scale very well, and it is even visible that for very large systems, the $\mathcal{O}(N \log N)$ scaling of P3M decreases the timing difference between the two algorithms.

We can see a few things from this graph. First and most obviously, P3M always outperforms MEMD in this setup. Second, the advantage is never greater than a factor of 2, which is very respectable. And third, for very large particle systems, MEMD noticeably catches up to P3M, since the linear scaling will perform slightly better than the

$\mathcal{O}(N \log N)$ scaling from the Fast Fourier Transform.

## 5.3. Parallel performance

A strong focus of the implementation within the ScaFaCoS library was the parallelization of each algorithm, since the library was planned out to be highly scalable for particle numbers and cluster sizes. Thus we were able to compare MEMD to other, more established algorithms using the library, up to a very large number of compute cores.

The findings presented in this section were published in collaboration with the other participants of the ScaFaCoS BMBF-project [2], and some formulations are taken directly from said collaborative manuscript.

### 5.3.1. The ScaFaCoS library

Particle simulation methods, like Molecular Dynamics or Monte Carlo sampling, are a well-established numerical tool to understand the dynamics and structure of many-particle systems. Long-range interactions such as electrostatic or gravitational interactions pose a particular challenge to such simulations, since their computation is very time consuming. Simple truncation schemes for electrostatic interactions have been shown to produce artifacts [109]. Therefore one has to take into account all pair interactions, leading to an unfavorable complexity of $\mathcal{O}(N^2)$ (where N is the number of particles). A number of efficient algorithms, in particular for electrostatic interactions, have been devised to reduce this computational effort. Although these algorithms compute the same quantities, namely electrostatic forces and energies, they differ largely in their properties.

The ScaFaCoS library has been designed with the idea to make the complex task of calculating long-range interactions readily available for simulation code that does not yet include them. It contains several popular methods and a common interface in multiple programming languages to easily use and test each of the methods with scientific simulation packages. The library has been developed by a group of scientists throughout Germany and is available as open source.

A big advantage of this design is that all methods can be tested and compared using a common simulation code and interface. This was impossible before, since a simulation package typically only implements one or two different algorithms for electrostatic interaction. The results of these comparisons are published in [2] and will partly be presented in this section.

### Available methods

ScaFaCoS mainly features six different algorithms for long-range interactions. There are two multigrid-solvers (MG), two Particle-Mesh Ewald (PME) methods, a Fast Multipole Method (FMM) implementation, and our MEMD code. In addition, the results can be compared against an Ewald sum implementation and a direct solver. For a more detailed description of all methods and references in literature, we refer to our publication [2] to keep this section brief.

### 5.3.2. Benchmark setup

#### Systems

In order to calculate the electric field $\mathbf{E} = -\nabla\Phi$ for a $1/r$-potential, all methods presented here split the long-range part of the potential from the divergence at $r \to 0$. Since the performance relevant part of the calculations is the treatment of the long-range components, all benchmarks presented here were carried out for two systems, consisting of different charge distributions, both of which feature a significant long-range contribution which challenges the achievable accuracy of the methods.

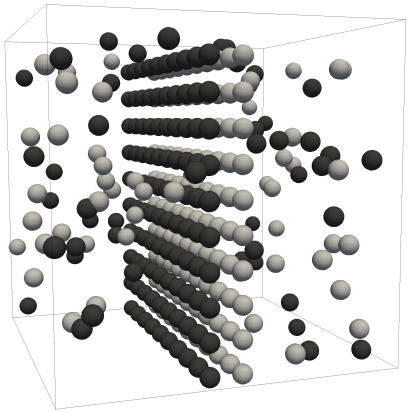


**Figure 5.3.:** The cloud-wall system (300 charges): two oppositely charged walls in the center of the box and a surrounding diffuse cloud. The system was artificially created to contain a strong long-range field component. The sketch has been published in [2].

The cloud-wall model system, shown in Fig. 5.3, consists of 300 particles, which represent two oppositely charged walls centered in a cubic box together with a diffuse cloud of charges. This ensures a strong long-range contribution in the potential. The periodic box was replicated 3, 7, 15, 32, and 70 times in every direction to yield cubic boxes filled with 8100, 102 900, 1 012 500, 9 830 400, and 102 900 000 particles, respectively.

The cloud-wall systems were used for both the performance measurements as function of accuracy in Sect. 5.3.4 as well as for the scalability benchmarks in Sect. 5.3.4. Since these test cases represent periodically replicated systems, the reference values for potentials and forces can be obtained even for very large numbers of particles.

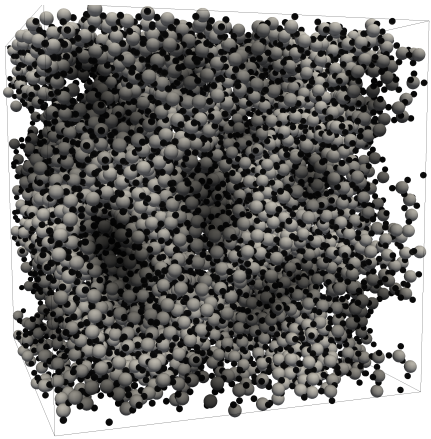


**Figure 5.4.:** Silica melt (12 960 charges): A system that is sufficiently homogeneous while retaining a significant long-range contribution. The sketch has been published in [2].

The second test system consists of a cubic box filled with 12 960 particles of a silica melt shown in Fig. 5.4. It was taken from an MD simulation of a melting silica crystal using the BKS force field [62]. The overall charge neutral system consists of positively and negatively charged ions which are sufficiently homogeneously distributed, while the electrostatic potential still has a significant long-range contribution. For the scaling and benchmark runs the original silica melt system was replicated 2, 4, 8, 16, and 32 times in every direction to yield cubic boxes filled with 103 680, 829 440, 6 635 520, 53 084 160, and 424 673 280 particles, respectively.

The silica melt test systems were used for both the stability benchmarks in Sect. 5.3.3 and the complexity benchmarks in Sect. 5.3.4.

### Error Measure

In order to compare the accuracy of the different methods, the following error measure is defined. Let $\Phi_{\mathrm{FCS}}(\mathbf{r}_j)$ denote the potential which is calculated by one of the presented fast Coulomb solvers and $\Phi_{\mathrm{REF}}(\mathbf{r}_j)$ the highly accurate reference potential computed by the Ewald summation method for which the parameters were chosen to yield an accuracy close to machine precision. In the following, we compare the different solvers with respect to the relative RMS potential error given by

$$\varepsilon_{\mathrm{pot}} := \left( \frac{\sum_{j=1}^{N} |\Phi_{\mathrm{REF}}(\mathbf{r}_j) - \Phi_{\mathrm{FCS}}(\mathbf{r}_j)|^2}{\sum_{j=1}^{N} |\Phi_{\mathrm{REF}}(\mathbf{r}_j)|^2} \right)^{1/2} .$$

Since all methods differ in their definition of relative short- and long-range contribution, only the total potential can be used as common reference point.

### Architectures

The benchmark tests were performed on two different hardware architectures at Jülich Supercomputing Centre. In the meantime, the JUGENE architecture has been shut down and replaced by the JUQUEEN system [318].

1. *Blue Gene/P (JUGENE) [317]*: One node of a Blue Gene/P consists of 4 IBM PowerPC 450 cores that run at 850 MHz. These 4 cores share 2 GB of main memory. Therefore, we have 0.5 GB RAM per core, whenever all the cores per node are used. The nodes are connected by a 3d-torus network with 425 MB/s bandwidth per link. In total JUGENE consists of 73 728 nodes, i. e. , 294 912 cores. The software has been built with the IBM XL compilers (Advanced Edition for Blue Gene/P, V9.0).

   In this work, we consider this architecture as prototypical for a well interconnected HPC machine.

2. *Jülich Research on Petaflop Architectures (JUROPA) [319]*: One node of JUROPA consists of 2 Intel Xeon X5570 (Nehalem-EP)

quad-core processors that run at 2.93 GHz. These 8 cores share 24 GB DDR3 main memory. Therefore, we have 3 GB RAM per core, whenever all the cores per node are used. The nodes are connected by a QDR InfiniBand network with non-blocking fat tree topology. In total JUROPA consists of 2208 nodes, i. e. , 17 664 cores. The software has been built with the Intel Compilers (version 11.1).

In this work, we consider this architecture as prototypical for a convenience cluster.

### 5.3.3. Stability

The long-term stability of an MD simulation is a good test for the accuracy of the computed forces. If a symplectic integrator is used, the time discretization error does not lead to a long-term energy drift, so that any remaining drift must be due to systematic errors in the forces. A symplectic integrator has the property that the discretized solution for the original Hamiltonian $H$ is equal to the *exact* solution for a near-by Hamiltonian $H'$ [129]. Since the energy for $H'$ is exactly conserved, its deviation from the energy for $H$ remains bounded. Likewise, a time-reversal symmetry of the integrator guarantees the conservation of total momentum. If total momentum is not conserved, there must be systematic errors in the forces.

To evaluate the long-term stability, we have run MD simulations of the silica melt system described in Sect. 5.3.2 using the BKS force field [62], the Coulomb component of which is computed with the different methods from the ScaFaCoS library. The simulations were performed with the MD code IMD [97], using a symplectic NVE leap-frog integrator. The system consisted of 12 960 atoms and was first equilibrated at 2950° C. The simulations were run over 100 000 time steps of 0.2 fs each. This time step is a rather small, conservative choice, so that remaining time discretization errors should be negligible.

### Conservation of Momentum

The results for the conservation of momentum are shown in Fig. 5.5, for two different required accuracies. At the beginning of the simulation, the total momentum was set to zero. The development of the specific kinetic energy of the center of mass motion, calculated in energy per particle, is a good measure to monitor the momentum conservation in the system.

As can be seen from Fig. 5.5, for the methods FMM, P3M, and VMG the center of mass momentum remains zero for all practical purposes, whereas with the methods P$^2$NFFT, PP3MG, and MEMD, a small but noticeable increase of the center of mass momentum is visible. The reason for the difference between P3M and P$^2$NFFT is the use of different differentiation schemes. The $i\mathbf{k}$-differentiation scheme used in P3M conserves the momentum exactly, but is somewhat slower, whereas the analytical differentiation scheme used in P$^2$NFFT conserves the momentum only approximately but is faster. Both methods can use either of these differentiation schemes. There is a trade-off between higher accuracy and higher performance involved when one selects the differentiation scheme. Note that VMG artificially enforces conservation of momentum while PP3MG does not.

### Conservation of Energy

The conservation of total energy is shown in Fig. 5.6, for the same required accuracies as for the momentum. For the higher accuracy, the drift of the energy is negligible for all practical purposes, even though the multigrid methods seem to have slightly higher energy fluctuations. The situation is different for the lower required accuracy, where the multigrid methods VMG and PP3MG show a very clear drift. This means that the forces contain a systematic error, which adds up during the simulation. The other methods, especially the Fourier-based ones (P3M and P$^2$NFFT) behave much better in this respect. They show no energy drift (or a much smaller one) even at low accuracy. This must be taken into account in the method selection. If a drift must be avoided, the multigrid methods have to be run at higher accuracy,

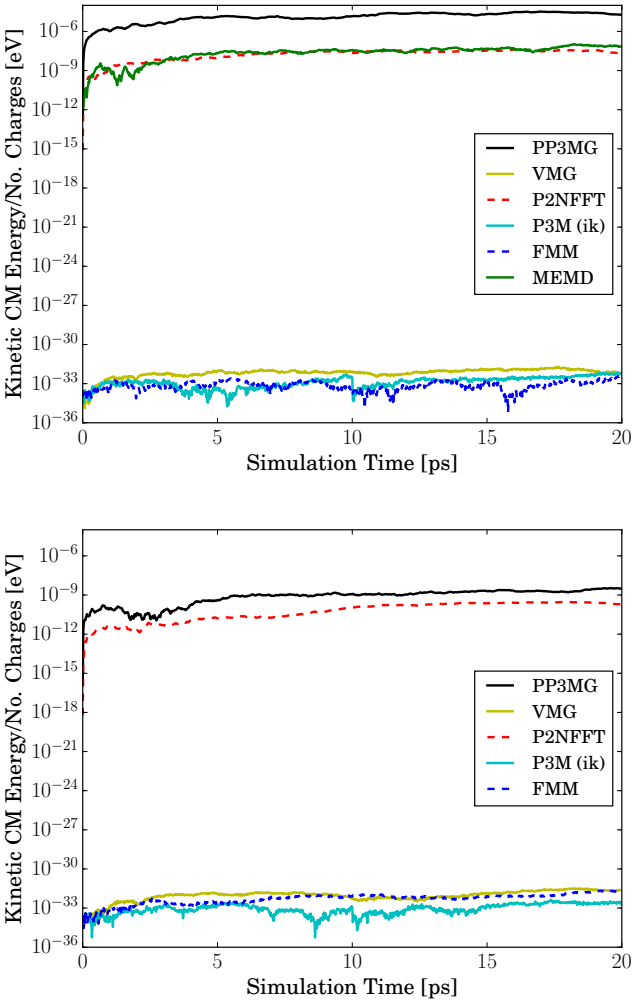



**Figure 5.5.:** Kinetic energy per particle of the center-of-mass motion of the system according to different methods and accuracies: $\varepsilon_{\text{pot}} = 10^{-3}$ (top), $\varepsilon_{\text{pot}} = 10^{-5}$ (bottom). Note that the P3M method used here uses an *ik*-differentiation for the forces instead of analytical differentiation and does not not apply interlacing. With MEMD, only an accuracy of $\varepsilon_{\text{pot}} = 10^{-3}$ could be reached. The graph has been published in [2].

which costs performance, whereas the other methods can be used at lower accuracy without having to deal with a drift.

It is apparent that MEMD is not capable to achieve an accuracy of $\epsilon < 10^{-5}$, which was found to provide long-term stability for the other methods. The lack of accuracy of MEMD is due to a low near-field resolution in directly adjacent cells, where no explicit particle interactions are considered. The application of a thermostat could possibly circumvent the numerical drift at lower accuracy but cannot guarantee the correct description of the dynamics of the system. However, if the main focus is on the configuration of a system and not on dynamics, e. g. , in Monte Carlo simulations, MEMD might be considered as a fast method.

### 5.3.4. Performance of the methods

In the following we assess the performance of each method in terms of accuracy, complexity, and parallel scalability. For the accuracy measurement, we inspect how much longer a method runs for a specific increase in desired accuracy of the relative potential error. The complexity tests check on the theoretically expected complexity with respect to the experimentally measured dependency of runtime on the number of particles. Finally, parallel scalability extends the benchmarks to very large systems of charges and examines parallel efficiency of the methods up to very large numbers of cores.

#### Accuracy

For the following benchmarks, the parameters of the fast Coulomb solvers were tuned in order to achieve different potential errors $\varepsilon_{\text{pot}}$. Fig. 5.7 shows the runtime per particle for the cloud-wall system with 102 900 particles on JUROPA. We discuss each method's source of error individually to explain the details of Figure 5.7.

The error introduced by the multigrid-based methods (VMG and PP3MG) is composed of a discretization error, an algebraic error, and an interpolation error. The first is due to the discretization of the Poisson equation (3.13) and depends on the chosen discretization scheme.

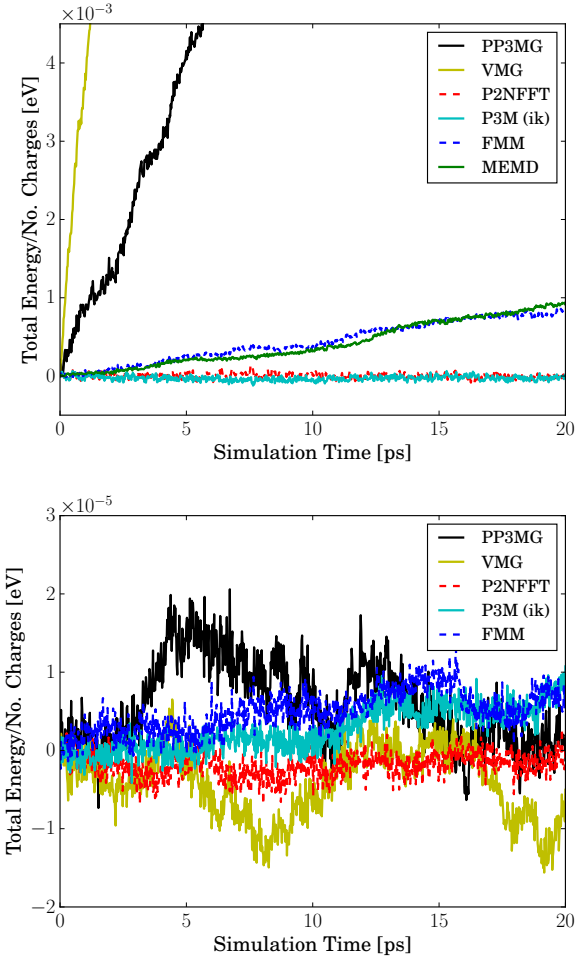



**Figure 5.6.:** Total energy per particle according to different methods at accuracies $\varepsilon_{\text{pot}} = 10^{-3}$ (top) and $\varepsilon_{\text{pot}} = 10^{-5}$ (bottom). The initial energy has been subtracted in order to make the fluctuations and the drift better visible. Note the different scales of the two figures (units of $10^{-3}$eV and $10^{-5}$eV). Note that the P3M method used here uses an $ik$-differentiation for the forces instead of analytical differentiation and does not not apply interlacing. With MEMD, only an accuracy of $\varepsilon_{\text{pot}} = 10^{-3}$ could be reached. The graph has been published in [2].

It scales typically like $\mathcal{O}(h^2)$, $\mathcal{O}(h^4)$, or $\mathcal{O}(h^6)$. Further, an appropriate splitting function has to be chosen that, in order to be represented adequately on the grid, has to be smooth enough and must have a large enough support. As a consequence the choice of a higher order scheme leads to a larger number of grid points a charge is sampled onto. The order of the interpolation scheme is also chosen according to the order of the discretization scheme, this results in a runtime behavior similar to the sampling part. The algebraic error due to the iterative solution of the linear system is controlled to be of the same order.

Putting this information together explains Fig. 5.7: The grid sizes, inverse proportional to the discretization parameter $h$, can only be increased by a factor of $2^3$ or powers thereof and the resulting error decreases then like the order of the discretization scheme, e. g. , $\mathcal{O}(h^4)$. However, as the accuracy was measured in powers of 10, not only the grid sizes are changed and the support of the splitting function are adjusted accordingly but also the optimal discretization scheme, interpolation degree, and type of splitting function are chosen individually.

Regarding the Fourier based methods, the error of the truncated Ewald sum splits into the errors caused by the truncation of the near field sum at a given near field cutoff range $r_{\text{cut}}$ and the error caused by the truncation of the Fourier series after $M$ mesh points in every direction of space. Note that both errors depend on the Ewald splitting parameter $\alpha$ in opposite direction, e. g. , for every given $r_{\text{cut}}$ and $M$ there exists an optimal choice of $\alpha$ for which both errors are balanced. In addition, P3M and P$^2$NFFT spread the charges $q_j$ with a B-Spline function on the grid in order to make the problem suitable for FFTs. This introduces another approximation error that decreases exponentially for increasing order $P$ of the B-spline. The P3M method seems to be significantly worse than P$^2$NFFT, however it was noticed that this seems to be caused mostly by problems in the automatic parameter tuning procedure. As the P3M method is equivalent to the P$^2$NFFT method, it can be expected that the method can be improved, by manually tuning the parameters, to perform on the same level as P$^2$NFFT.

The FMM has two distinct error sources contributing to the overall error. Both error sources emerge in the far field only. The FMM near field is free of errors except for numerical rounding-off visible in

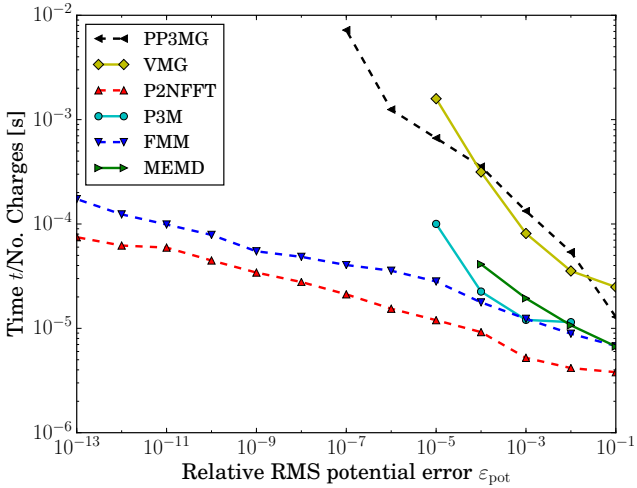



**Figure 5.7.:** Required wall clock time per particle for 102 900 charges versus the relative RMS potential error $\varepsilon_{\text{pot}}$ on one core of JUROPA. The graph has been published in [2].

all methods. The first algorithmic error source, the limited number of poles, introduces a truncation error. The second error source occurs whenever the translation operator M2L is applied. Both errors can be controlled in the current implementation such that the results do not show any errors compared to the direct summation or even exceed the precision due to fewer terms in the summation. One cannot state any accuracy scaling properties for systems with arbitrary particle distribution. However, for well behaved, almost homogeneously distributed systems like the ones used in this comparison, the FMM shows the following scaling behavior. First, up until $\Delta E_{\text{rel}} = 10^{-4}$ the runtime of the FMM is constant. Second, the runtime increases linearly until $\Delta E_{\text{rel}} = 10^{-8}$. Finally, the FMM shows a quadratic scaling behavior until machine precision. The theoretical $\mathcal{O}(p^3)$ scaling, with $p$ being the number of poles in the expansion, is not visible when increasing the requested accuracy up to machine precision, since the FMM does tune the near and far field contributions automatically which improves

the accuracy scaling.

More detailed information on the numerical errors of the MEMD algorithm were presented in section 3.2.3 within this thesis. In comparison in Fig. 5.7, MEMD performs acceptably but can only cope to a maximum precision of $10^{-4}$ in the relative potential error.

### Complexity

In this section we compare the theoretical complexity of each method with the measured runtime complexity. Fig. 5.8 shows the run time per particle as a function of the number of particles for the various methods on a single core of the JUROPA system with the silica melt test system duplicated as described in Sect. 5.3.2. All the methods are required to maintain the relative RMS potential error of $\varepsilon_{\text{pot}} < 10^{-3}$. Note that not all the implementations are intended to run such large problem sizes on a single core. Therefore, a lack of memory or other implementation depended limitations lead to some missing data points in Fig. 5.8.

We observe an almost linear increase of runtime with increasing number of particles for all compared methods, which agrees perfectly with the theoretically $\mathcal{O}(N)$ complexities of the MEMD, PP3MG, VMG, and FMM methods. The time per particle varies between $5.2\times10^{-6}$ and $1.4\times10^{-4}$. Although P3M and P$^2$NFFT yield a theoretical $\mathcal{O}(N \log N)$ complexity, this is not visible in the compared range of particle numbers. This does not emerge as the runtime share of the FFT, which is responsible for the asymptotic $\mathcal{O}(N \log N)$ scaling, ranges only between 3% and 10% of the total runtime. Further note that there is no crossover point of the FMM and P3M run time up to $5 \times 10^7$ particles. However, the run time difference between FMM, P3M, and P$^2$NFFT are rather marginal for all compared system sizes.

### Scalability

Our last benchmark is focused on parallel scalability. Although each method presented here shows linear scaling behavior for a small number of cores, at some point implementation dependent restrictions will

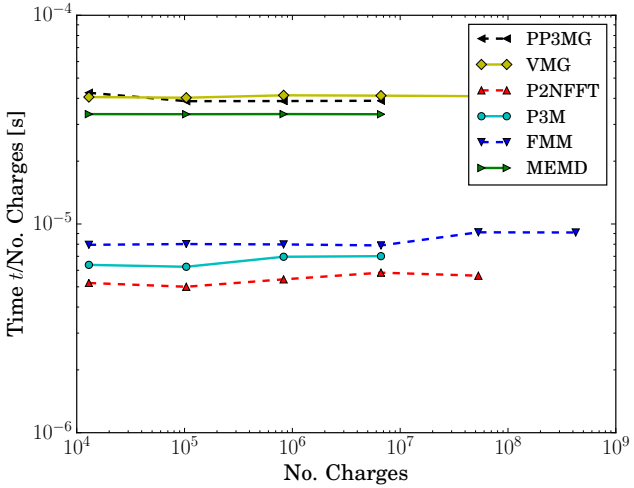



**Figure 5.8.:** Required wall clock time per particle at relative RMS potential error $\varepsilon_{\text{pot}} < 10^{-3}$ on one core of JUROPA. The graph has been published in [2].

cause a deviation from the $\mathcal{O}(N/P)$ behavior at large problem sizes $N$ and large numbers of cores $P$ and thus a decrease in efficiency for highly parallel execution, as expected for strong scaling. The timings presented in this section were performed on the two different architectures described in Sect. 5.3.2. All algorithms are tested for parallel scalability using the cloud-wall test case from Sect. 5.3.2 on the JUGENE system, which has a large number of slowly clocked cores, and on the JUROPA system, an architecture more similar to common compute clusters. Results of the wall clock measurements for 1 012 500 charges on the JUROPA system are presented in Fig. 5.9.

The gray dotted lines drawn in Fig. 5.9 each denote a factor of 10 in the run times with respect to the fastest method on one core. A graph starting out on the lower line therefore is reduced to a relative efficiency of 0.01 when crossing the upper line, providing the best visible resolution in a region of very low efficiency. All methods in Fig. 5.9 show very similar efficiency and their scaling behavior at low core numbers

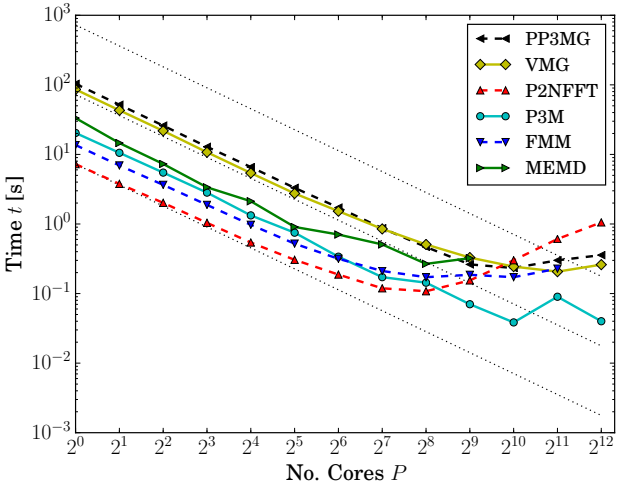



**Figure 5.9.:** Required wall clock time of the cloud-wall test case with 1 012 500 charges versus the number of cores on JUROPA at relative RMS potential error $\varepsilon_{\text{pot}} < 10^{-3}$. The graph has been published in [2].

appears linear.

To provide a better resolved and easier to read comparison, the scaling plots presented from here on will feature the *relative parallel efficiency* in dependence on the number of cores. Let $P_{\text{min}}$ denote the minimal number of cores that was included in the measurements and $t_{\text{best}}$ the run time of the fastest method at this level of parallelism. We then plot the *relative parallel efficiency* $e(P)$, which we define by

$$e(P) = \frac{t_{\text{best}}}{t(P)} \frac{P_{\text{min}}}{P}. \tag{5.2}$$

The same scaling measurements as in Fig. 5.9 are shown in Fig. 5.9c using equation (5.2). In direct comparison, the non-linearity even at small numbers of cores can be seen, and the scaling trend of each algorithm in the most interesting scope above 10% efficiency of the fastest method is more apparent on the non-logarithmic scale. The only disadvantage of plotting the parallel scaling like this is that it is

not possible to read off the actual timing of an algorithm from these plots. To allow for this, we give the best timing $t_{\text{best}}$ at the minimal number of cores $P_{\text{min}}$ in the caption of the scaling graphs. The actual timing can then be calculated as $t(P) = \frac{t_{\text{best}}}{e(P)} \frac{P_{\text{min}}}{P}$. Where $P_{\text{min}} \neq 1$, we give it explicitly.

Note that the P3M method used here uses an *ik*-differentiation for the forces instead of analytical differentiation and does not not apply interlacing. We now discuss the scalability of each method individually.

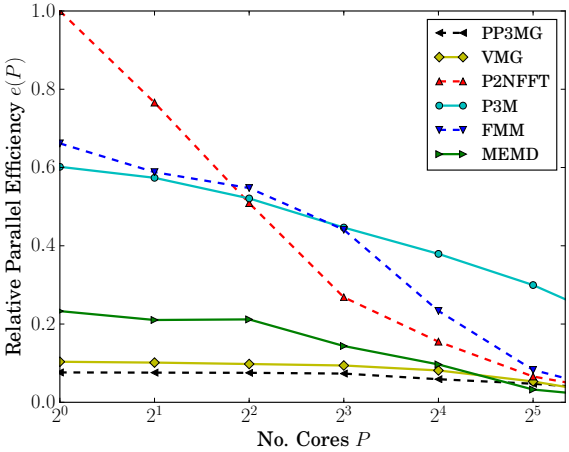
### MEMD performance

The MEMD algorithm performs acceptably on the JUROPA system. The absolute timings do not reach those of the fastest methods but are situated in the mid range compared to all alternatives. The parallel scaling is not as strong as expected from a purely local method, which is due to the large amount of data sent between cells at the node boundaries. In addition to interpolated charges as in other methods, currents and electric and magnetic fields have to be exchanged, which leads to a communication overhead on clusters with good single CPU performance.

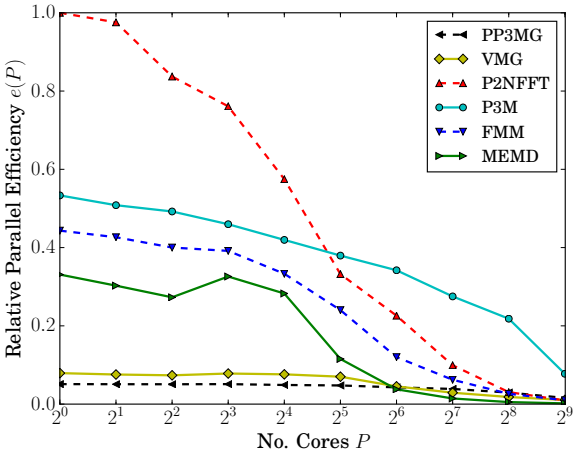
On a system with slower clocked cores like JUGENE, MEMD performs quite competitively, as can be seen in Fig. 5.11. A comparison between the two architectures reveals room for improvement within the communication structures.

The timings shown here are all measurements of the dynamic solution of the algorithm and do not include the first time step, which is calculated via the initial numerical relaxation scheme. Since MEMD does not provide a tuning method for a given error estimate, manually tuned parameters were used for all simulations to obtain the required accuracy at the coarsest possible mesh. On the JUGENE architecture, due to memory consumption, the simulations had to be restricted to higher numbers of cores.

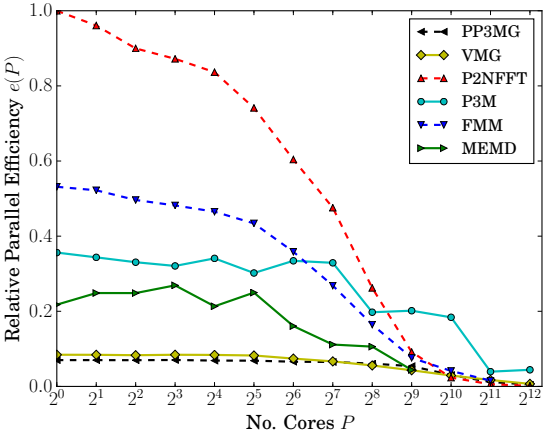
For a detailed analysis of the performance of all other methods, we refer to the publication [2].




**(a)** Cloud wall test case with 8100 charges. $t_{\text{best}} = 6.35 \times 10^{-2}$s.

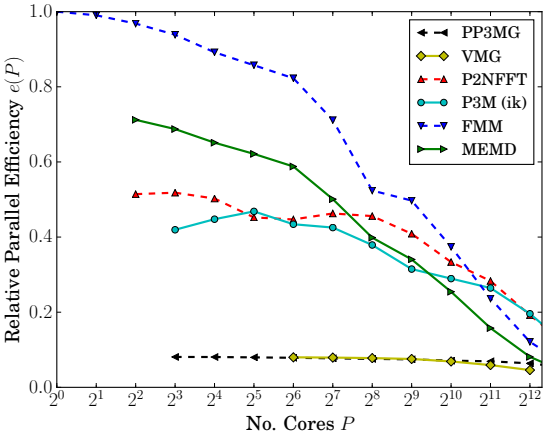



**(b)** Cloud wall test case with 102 900 charges. $t_{\text{best}} = 6.60 \times 10^{-1}$s.

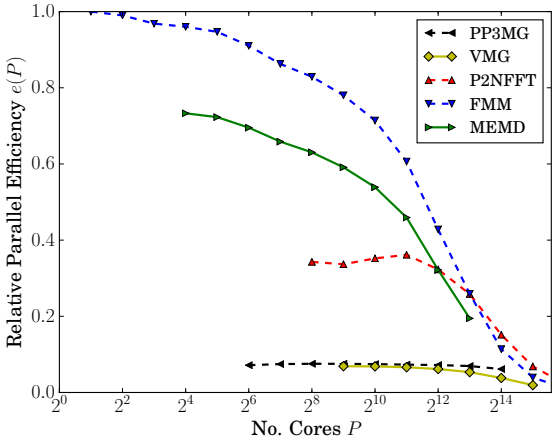



**(c)** Cloud wall test case with 1 012 500 charges. $t_{\text{best}} =$ 7.25s.

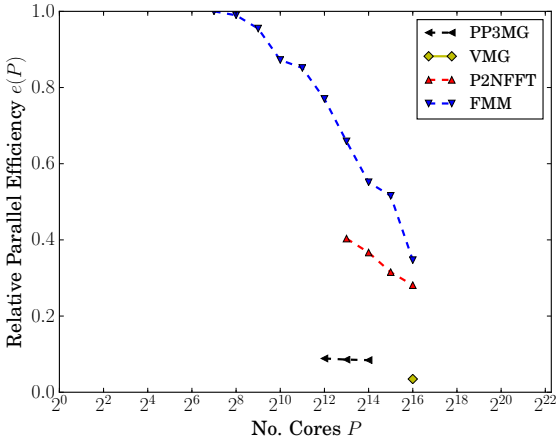
**Figure 5.10.:** Relative efficiencies $e(P)$ (see (5.2)) of the cloud-wall test case at different sizes versus the number of cores on JUROPA at relative RMS potential error $\varepsilon_{\text{pot}} < 10^{-3}$. The graphs have been published in [2].




**(a)** Cloud wall test case with 1 012 500 charges. $t_{\text{best}} =$ 106.7s.




**(b)** Cloud wall test case with 9 830 400 charges. $t_{\text{best}} = 454.8\text{s}; P_{\text{min}} = 2$.




**(c)** Cloud wall test case with 102 900 000 charges. $t_{\text{best}} = 753.1\text{s}; P_{\text{min}} = 16$.

**Figure 5.11.:** Relative efficiency (see (5.2)) of the cloud wall test case with relative RMS potential error $\varepsilon_{\text{pot}} < 10^{-3}$ on the JUGENE architecture. The graphs have been published in [2].

### 5.3.5. Performance comparison

As a general remark we emphasize that the comparison was made between methods included in the parallel library ScaFaCoS. Due to the common interfaces, positions and charges are copied into the library and re-sorted internally. Although there are various kinds of optimizations already done, a performance gain could be obtained by transferring sorted arrays of charges and positions of particles in, e. g. , block structures or along space filling curves to the library for an improved cache usage. Furthermore, the evaluation of the short range part of interactions within the force-loop of the MD code has the potential to increase performance due to the combined calculation with empirical potentials such as Lennard-Jones, which avoids a double calculation of mutual distances between near-field particles in the MD code and in the library. For the present comparison the implementation of the near-field contribution was calculated consistently with the methods offered by ScaFaCoS in order to put all methods on a common ground. Finally, changes to particle position in an MD simulation are usually small such that especially the PDE-based methods may benefit from starting at solutions gained from the previous time step. However, such optimizations have not been explored for this comparison for sake of a common ground.

The systems chosen for the benchmarks fulfill the requirements to exhibit a sufficient contribution from long range contributions to the electrostatic potential, whereas the distribution of particles is relatively homogeneous. It has to be pointed out that for systems which show strong inhomogeneous particle distributions, mesh-free methods like FMM will most likely gain in relative performance with respect to mesh based methods, like $P^2$NFFT and P3M, since a sufficient resolution of the particle distribution will call for large meshes.

It is apparent that for architectures consisting of powerful single cores, the FFT based methods $P^2$NFFT and P3M show the best performance of all methods included in the ScaFaCoS-library. It can be seen that the chosen parameters are crucial for accuracy and speed. For example, the real-space cutoff radius might be set small when a small number of processors is used, since the far-field part can then be

very efficiently calculated by a well scaling FFT. However, when the number of cores is increased and the efficiency of the FFT is degraded due to communication, more work is transferred to the short range part by finding a balance between far-field and near-field calculations by a larger near-field cutoff radius.

For architectures with a smaller ratio of communication to computation time, i. e. , a fast network structure together with slow single cores, the fastest algorithm is the FMM. It also features the smallest memory requirements, allowing for very large systems even on small numbers of cores. The implementation has been optimized for a small memory footprint, and it is most likely that the timings can even be further improved by use of a less memory efficient sorting and communication schemes. For a small error threshold ($\varepsilon < 10^{-5}$), controlling the approximation, the given implementation of the FMM is sufficiently energy and momentum conserving for long trajectory calculations. A further advantage of the FMM, that was not explored in the present article, is its ability to deal with partially periodic systems.

The multigrid methods perform acceptably and show a very good scaling behavior. The only global communication which is required is the reduction of the defect over all cores which is necessary for controlling the overall convergence of the method. Further optimization could reduce the workload of local computations, i. e. , charge assignment and near field correction. Although the method shows a moderate performance compared with other methods at the same accuracy, the underlying multigrid solvers are flexible enough to allow for many variations, e. g. , spatially varying dielectric properties of the Poisson equation, which makes them attractive for a broader class of applications.

MEMD can certainly compete with the more established algorithms presented in this section, especially on architectures with a favorable ratio of single core performance and communication performance. The performance at higher accuracies remains an open question but is, in principle, achievable. Methodically, MEMD is highly scalable and of the implementations presented here, it is the only one capable of dealing with dielectric variations within the system. It is very encouraging to see that in addition to this fact, it can keep up with other available options when it comes to accuracy, stability and effectivity. If we factor

in the additional amount of computational work needed to include dielectric enclosures into the other methods, for example via ICC$\star$ iterations, it might even outperform most of them.

# 6. Applications

A crucial part of any new algorithm is how and where it can be applied in simulations of real world problems. As we have shown in Chapter 5, MEMD is an algorithm for general electrostatic problems. But in terms of maximum precision and restrictions, as well as overall computational performance, it can not outperform FFT-based methods like P3M. Its true strength lies in the inclusion of varying dielectric properties, for which it can not only outperform other methods but simulate conditions that were not accessible before to MD simulations. Naturally, the inclusion of dielectric interfaces and gradients is what we focus our applications on.

In this chapter, we present a selection of simulations of physical systems to which we have applied our newly developed algorithm. This will not only give us confidence in its usability and correctness, but will also shed light on the impact that spatial changes in the dielectric permittivity have in real world applications.

To present everything we have looked into would easily get out of hand, so we focus on three pivotal examples: First, a system that can be simulated with several different algorithms for electrostatics including dielectric contrasts within ESPResSo and other software. Second, a coarse-grained system that can to our knowledge only be simulated with one other simulation software, the HIM algorithm developed at the ICP in collaboration with Zhenli Xu. With this, we enter new territory but can still compare our results to another algorithm. And third, an entirely dynamic and flexible system that can to our knowledge not be simulated by any other coarse-grained Molecular Dynamics code and has certainly not yet been studied.

## 6.1. Soft matter systems

The simulational area of soft charged matter is very suitable for our purposes. First, our simulation software ESPResSo is optimized for and targeted to such systems. Second, biophysical systems often contain objects in aqueous solution. Besides water providing a significant bulk permittivity of roughly $\varepsilon_W = 78$, such systems contain charged solvated objects as well as counterions and often additional salt ions, and electrostatic interactions play an important role. And third, the solvated objects generally have a very different polarizability than water, and highly charged objects as well as salt concentration can even locally lower the polarizability of water. Thus, we will find realistic simulation setups that will certainly contain sharp dielectric interfaces as well as areas of smoothly varying dielectric permittivity.

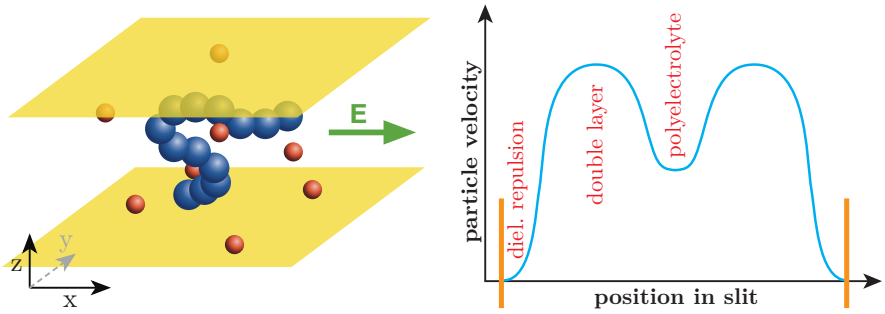
Last but not least, all systems presented here have been studied by ourselves and other groups using alternative algorithms without spatial changes in the dielectric constant or with sharp interfaces. We can therefore compare our results to established simulations and see if similar setups give the same results, and what the influence of varying dielectric properties is.

## 6.2. Slit pores

### 6.2.1. Polyelectrolyte in an infinite slit pore

A system that has been studied extensively in theory [171, 188, 202, 226, 249] and experiment [136, 141, 157], is the motion of a polyelectrolyte in a confined nanochannel. As depicted in figure 6.1a, a charged polymer is put between two constraining charged walls and its motion, sometimes with an externally applied electric field, is recorded.

The polyelectrolyte mobility even in an unconstrained system, and therefore its speed in the direction of the electric field, depends on the external field, and the electrostatic and hydrodynamic drag. The latter two contain information about electrostatic and hydrodynamic screening and friction. If the system is placed inside a nanoslit constraint, the walls will exert an additional friction to the polymer. Therefore,




**(a)** Sketch of the system setup **(b)** Effects on the flow profile

**Figure 6.1.:** A polyelectrolyte with salt ions is placed between two charged plates. If we apply an external field in $x$-direction, the speed at which the polyelectrolyte moves in field direction will, amongst other things, be depending on the friction between the polymer and the wall boundary.

the statistical distribution of its position in $z$-direction, perpendicular to the plates, is of key importance to the process.

For the $z$-position of the polyelectrolyte, we will see a combination of several effects. The negatively charged polyelectrolyte will be electrostatically repelled by the negatively charged walls. The positively charged counterions will be attracted to the wall. The polyelectrolyte will move against the direction of the electric field, whereas the counterions will move along with it. The polyelectrolyte as well as the ions will interact with the wall boundary and with the flow field via hydrodynamics, a sketch of which is provided in figure 6.1b.

The flow field consists of two major regions: First, a double layer in the vicinity of the boundary, where positive ions will assemble close to the negatively charged wall and move in direction of the electric field. And second, a polyelectrolyte in the center that will move the opposite direction. The polyelectrolyte is electrostatically repelled from the walls, but these interactions are mostly screened by counterions and distance from the wall. But since the double layer flow field close to the wall will have a noticeable momentum in the direction of the electric field, it is energetically more feasible for the polyelectrolyte to

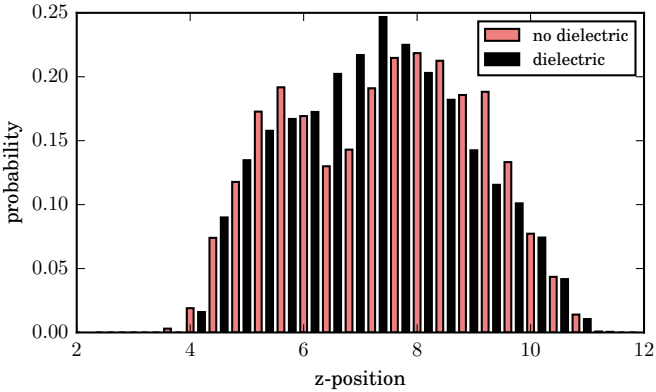
stay restricted to the center of the nanoslit. As sketched in figure 6.1b, a dielectric repulsion of all charges from the wall will lead to a shift of the double layer towards the center of the slit. This directly leads to an increased friction between the electrophoretic double layer and the polyelectrolyte, which will stick to the center area even more. A resulting histogram of the center of mass position is shown in figure 6.2

We simulated polyelectrolytes with 4, 8, 16, 24, and 32 charged monomers in a 50 mM salt solution between plates of 20 nm distance. Simulations were executed for a system with constant background permittivity of $\varepsilon_W = 80$, and with dielectric boundaries of $\varepsilon_C = 1$ at the wall constraints. We used Dissipative Particle Dynamics (DPD) with tunable slip boundaries to simulate the hydrodynamic interactions, since it is momentum conserving. We hereby used the no-slip condition for our setup. While for bigger electric field values in figure 6.2a, there is hardly any change, for the smaller external electric field in figure 6.2b, there is a clear difference between the simulations with and without dielectric. As explained, the double layer close to a dielectric wall is widened due to image charge repulsion. The effective friction between the polyelectrolyte and the wall via the double layer is increased and statistically, it will keep closer to the center of the slit.

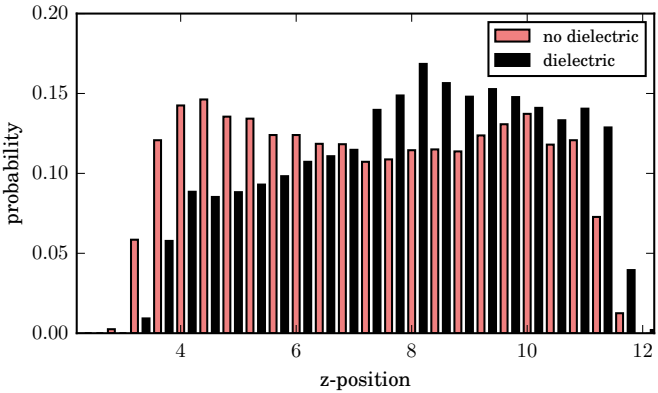
If we apply an external electric field, we can directly measure the polymer velocity and therefore its mobility $\mu E$. The results are shown in figure 6.3.

It is obvious that the two curves differ qualitatively. While the mobility for simulations without dielectric interfaces exhibits a drastic drop in mobility between $N = 16$ and $N = 24$, such a strong decrease only occurs between $N = 8$ and $N = 12$ if dielectric discontinuities are included.

Such systems can experimentally be used to determine the length of a polymer, since the wall friction and therefore the mobility of the polymer strongly depends on its length $N$ [222, 230] However, our quantitative predictions of coarse-grained Molecular Dynamics simulations thus far did not match the experimental evidence. The inclusion of dielectric interfaces for the nanoslit is certainly a step in the right direction, since the double layer within the nanoslit is significantly widened, and the polymer experiences friction from the wall bound-




**(a)** particle position for $\mu\mathbf{E} = -0.6$ (with $\mathbf{E} = 0.1$)




**(b)** particle position for $\mu\mathbf{E} = -0.3$ (with $\mathbf{E} = 0.1$)

**Figure 6.2.:** Position in $z$-direction of the center of mass of the polymer. Figure (a) shows the distribution for a higher external electric field, figure (b) for a lower. The red bars depict the simulations without dielectric boundaries, the black bars the ones with dielectric interfaces. It is apparent that the polyelectrolyte is more drawn to the center if dielectrics are included. This effect is much more pronounced for lower electric field values.

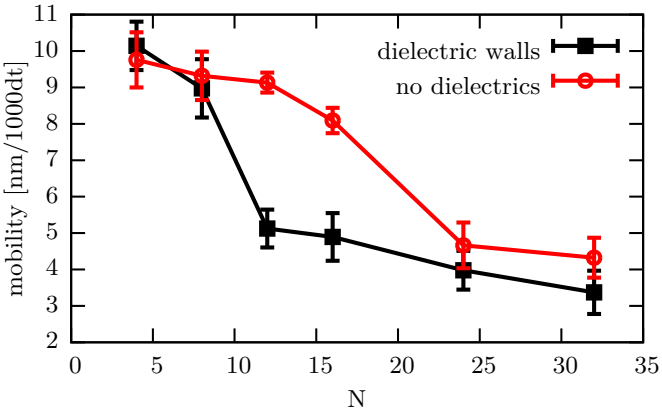



**Figure 6.3.:** Mobility of the polyelectrolyte, dependent on the number of monomers $N$. With the inclusion of dielectric interfaces at the wall boundaries, a qualitative difference can be observed. Without dielectrics, the mobility sharply drops between polymer lengths of 16 and 24, whereas with dielectric boundaries, the sharp drop is between 8 and 12.

aries already at shorter chain lengths, matching the experimental data more closely [189].

While this is certainly an interesting and active field, we did not publish our findings or pursued them further. Partially, this is because the results are not groundbreaking or of high impact to the research community. But mostly, such systems can be examined with much less computational effort using the established MMM2D algorithm. Since the setup is supposed to be periodic only in two dimensions, we had to include a vast empty area between slits to ensure correct behavior, and the computational effort needed was quite significant for such a simple system. This is also the reason why we had to work with large statistical errors. We were looking for a setup that would be better suited for MEMD, maybe not even accessible for other methods.

### 6.2.2. Dielectric corners

We have included the above section on polyelectrolytes in slit pores in this thesis, since it is a convenient start into the world of nanochannels. Such a strict confinement of polyelectrolytes can lead to unexpected and counterintuitive behavior. A prime example of those qualities was observed experimentally by M. Krishnan in 2007 and led to a row of publications on the subject [194, 215, 251, 294, 295].

The group placed flourecently labeled polyelectrolytes in a nanochannel confined in two dimensions, displaying a cross section very similar to the one depicted in figure 6.8. To their surprise, they saw that the negatively charged polyelectrolytes were pulled over time into the corners of the negatively charged pore and stuck to it. The process can be seen in figure 6.4.

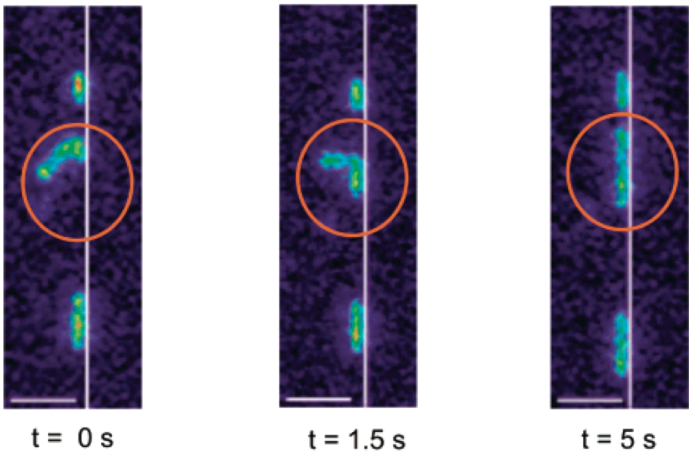



**Figure 6.4.:** Experiments done by M. Krishnan in 2007. A polyelectrolyte touches the similarly charged corner of a slit pore, stretches out, and sticks to the wall. Picture taken from [194].

Their experimental setup included negatively charged $\lambda$-DNA strands in a negatively charged glass channel. Interestingly, the behavior shown in figure 6.4 was only observed for sufficiently narrow channels. If the channel has a height of more than 100 nm, the effect did not occur. The group additionally studied the system using static mean field simula-

tions with the COMSOL software package, but the results they found were inconclusive and could not sufficiently explain the effect. We essentially replicated their results in COMSOL and the graph is shown in figure 6.5.

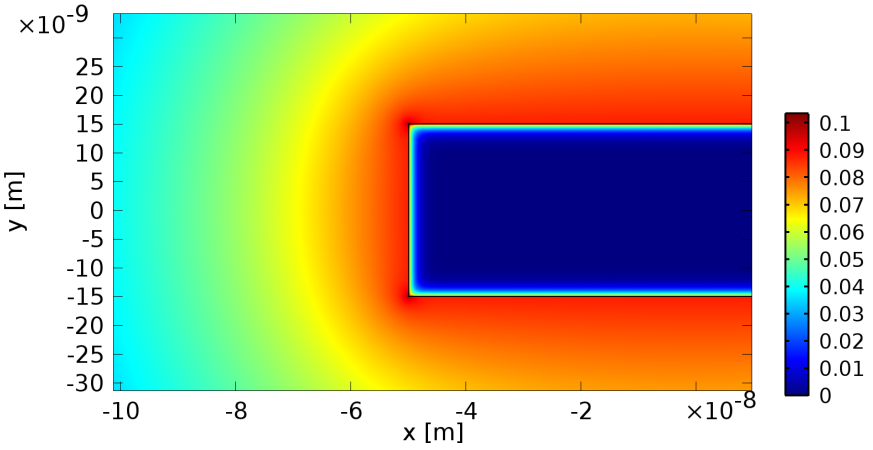



**Figure 6.5.:** The electrostatic potential of a nanopore with highly charged walls and sharp dielectric interfaces. The potential does not display a noticeable drop in the corner which would suggest electrostatic attraction of any sort.

Looking at the potential in figure 6.5, we can already see that the electrostatic potential does not display any noticeable irregularities in the slit corners. Additionally, it is very apparent that for the given salt concentration of the experiment, electrostatic interactions are completely screened over the slit height of 20 nm and the potential is identical zero in the slit center. In figure 6.6, we take a closer look at the potential as a function of distance from the corner, when moving on a straight line towards the slit center.

The potential increases monotonically in close proximity of the corner, which does not allow any attractive force

$$\mathbf{F}_{\text{Lorentz}} = q\mathbf{E} = -q\nabla\Phi < 0 \quad . \tag{6.1}$$

What these simulations do not take into account but is mentioned by

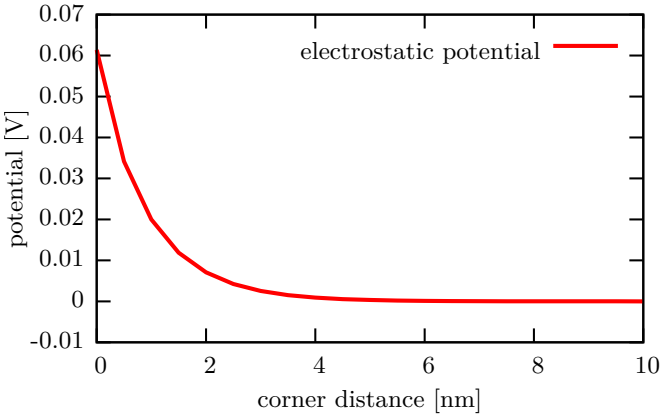



**Figure 6.6.:** Moving in a straight line from the corner to the slit center, this is the resulting electrostatic potential from our COMSOL simulations. It shows monotonic increase towards the corner.

M. Krishnan in her articles is the fact that if the amount of polyelectrolytes and additional salt is kept constant, but the channel volume decreases, this directly influences the concentration of monomers and counterions in the system. Making a narrower channel with an otherwise identical geometry will also decrease the volume linearly while keeping the glass surface almost constant. This will add even more solvated counterions to the system, which results in a significantly increased salt concentration. This leads to a shorter Debye length and therefore shorter range for electrostatic interactions.

We can use the implementation of the iterative Poisson-Boltzmann solver (iPBS) [270, 316] provided by A. Schlaich to simulate the same setup but with a correct value for the salt concentration. The simulation provides again a static solution of the two-dimensional problem, where we placed an infinitely long polyelectrolyte along the nonexistent dimension, parallel to the slit. With a surface charge density of $\sigma = 1\,\mathrm{e/nm^2}$ and a dielectric permittivity of $\varepsilon_C = 2$ for the walls and the polyelectrolyte alike, and a surrounding bjerrum length calculated from the correct salt concentration in the system, we simulated a cuboid box for different ratios $h/w = [0.25 \ldots 1.0]$ of the box height $h$

to the box width $w$. The resulting force on the polyelectrolyte is shown in figure 6.7.

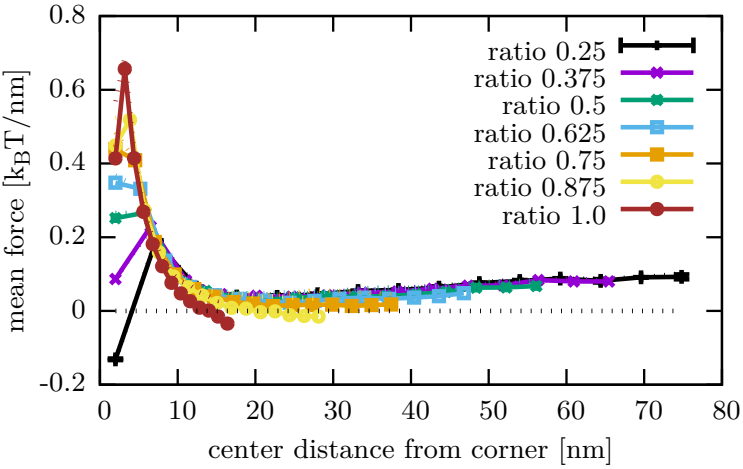



**Figure 6.7.:** For different ratios $h/w$ of the channel height $h$ and width $w$, we calculate the force on a polyelectrolyte moving towards the corner. The results are the solution of the two dimensional static iPBS method. We can see that for very narrow channels, we can actually see a force reversal.

We can see that even for wider channels (with a high ratio $h/w$), the force suddenly decreases at very short distances from the corner. This attractive force could be an indication for the clinging of the polyelectrolyte to the corner, but at a maximum of $|\mathbf{F}| = -0.15\,k_BT/nm$ at a position of 2.5 nm away from the crossover it is not strong enough to restrict a monomer with an average kinetic energy of $1\,k_BT$, much less for a wider channel.

Since we are researching sharp interfaces and sharp corners at high surface charge densities, the mean field solvers we have used thus far might not be a sufficient representation of the system. Additionally, we can adapt the geometry of the channel to be closer to the experimentally used ones. For our new simulations, we used ESPResSo and MEMD with the setup as depicted in figure 6.8.

For a channel width of $w = 200\,nm$, we varied the channel height

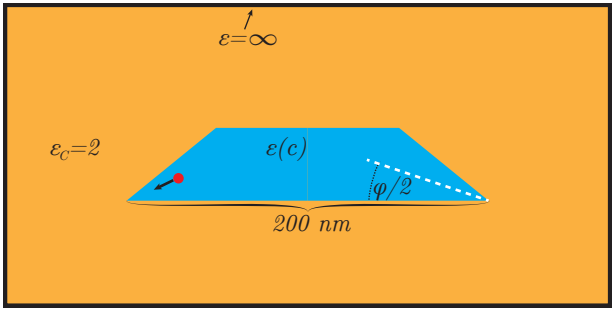



**Figure 6.8.:** Sketch our setup for MD simulations. The box is surrounded by a layer of infinite permittivity to ensure zero potential at the boundaries. The channel itself has sharper corners and a salt-dependent permittivity $\varepsilon(c)$, while the surrounding material is fixed at $\varepsilon = 2$. We can vary the channel height and measure the electrostatic energy of a monomer moving into the corner along the white dashed line at an angle of $\varphi/2$.

$h = [5 \dots 40]\,\mathrm{nm}$ and calculated the resulting salt concentration and Debye length. For the simulations presented here, we kept the corner angle $\varphi = 45°$ constant, since this provided a very straight forward way to interpolate the dielectric interface for the MEMD algorithm, where lattice links are either entirely outside or inside the channel and do not ever cross the interface. We varied the surface charge density of the channel, although the data shown here always contains a surface charge of $\sigma = 1\,\mathrm{e/nm^2}$.

We measure the electrostatic potential of a single charge being pulled into the corner along the bisecting line of the angle $\varphi$, depicted by the dashed white line in figure 6.8. Plotted against the distance from the corner and for several different channel heights, the results are shown in figure 6.9.

In the data set shown in figure 6.9, it is very visible that for very narrow channels ($h < 11\,\mathrm{nm}$), we can observe a clear minimum in the electrostatic energy that even exceeds $-1.5\,\mathrm{k_BT}$ in comparison to the adjacent maximum. The potential rise is very steep, and the end of a polyelectrolyte could certainly be trapped in this minimum, since it is confined in a polymer chain and can not easily muster a kinetic energy of more than $2k_BT/N$, where $N$ is the polymer length.

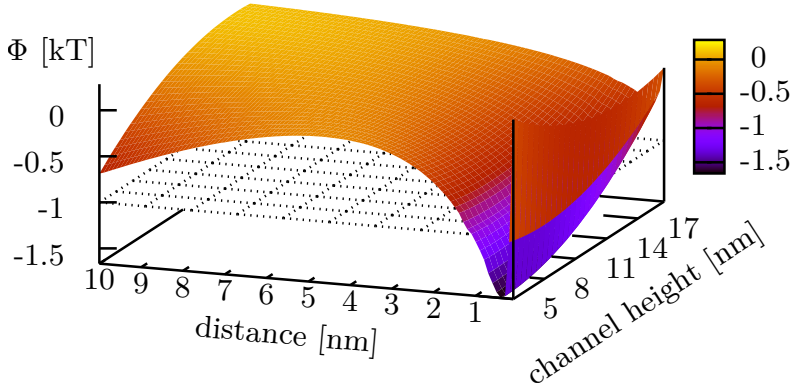



**Figure 6.9.:** Electrostatic potential as a function of the distance from the corner, for several channel heights between 5 and 20 nanometers. The data is shifted so the potential for the narrowest channel ($h = 5\,\mathrm{nm}$) reaches a maximum of $0\,k_BT$. For very narrow channels, it can clearly be seen that a minimum in the electrostatic energy occurs at very short distances from the corner. The depth of this minimum exceeds $1.5\,k_BT$ and is enough to trap a monomer in the corner.

We expect that for long time scales, even with the entropic and electrostatic disadvantage, a monomer can slip into this minimum and get trapped. It will then, just as can be seen in experiment in figure 6.4, slowly pull in the rest of the polymer chain, leaving it stuck to the corner.

We did not publish these finding, since the setup presented here, with a very sharp corner, a very narrow channel, and a very high surface charge density, is the only data set we found that exhibits this peculiar behavior. The used parameters do not represent the experimental system of M. Krishnan and are unrealistic. We have some amount of confidence that the data is reasonable, since the iPBS code gave similar results for extremely narrow channels. But the fact that even for channels with $h = 20\,\mathrm{nm}$, or for surface charge densities of $\sigma = 0.8\,\mathrm{e/nm^2}$, or for a rectangular corner, the minimum does not show up anymore or is not pronounced enough to trap a monomer, suggests that more research is needed to reach a scientifically sound conclusion.

It should also be stated that the simulations leading to these results took a very long time. The dielectric permittivity resolution, and therefore the MEMD lattice, needs to be fine enough to represent the system, the channel needs to be wide enough to screen any influence of the opposing corner, and the simulation cell needs to be big enough to cancel any influence from periodic images. This very large MEMD lattice, coupled with the fact that for these simulations we had to rely on the initial solution for every new setup and then only ran a few thousand dynamic propagation steps to pull the charge into the corner, resulted in months-long simulation times for the parameter sweeps.

Since this particular system can easily be simulated with algorithms that include sharp dielectric interfaces, particularly ICC$\star$, it would be beneficial to extend the parameter sweeps and maybe find some more conclusive or systematic results.

## 6.3. Colloidal systems

We researched a colloidal system in collaboration with Zhenli Xu during his stay at the Institute for Computational Physics. The results were published in [5], and some of the graphs and explanations in this section reappear in our publication.

### 6.3.1. Introduction

When a charged nanoparticle (colloid or biopolymer) is immersed in an aqueous solution, counterions accumulate near the interface between the macroion and the solvent, forming an ionic cloud which neutralizes the bare charge. The structure of this ionic cloud is called the electric double layer (EDL) and plays an important role in the physical and chemical properties of many systems at small scales. [128, 131, 154, 197, 211, 245, 273, 302] The ion distributions in EDLs are the result of the balance of electrostatic interactions and entropic repulsion, and are affected by various factors such as surface charge density and discreteness, ionic size and valency, and the system temperature.[245, 273]

In response to the already present, widely spread, experimental data, researchers are increasingly examining the properties of colloids from a simulational approach.[100, 107, 179] The use of coarse-grained models is crucial for dealing with systems on such a big scale. In addition to reducing the colloid to one large sphere or sometimes treating it as an infinite plane, [300] one of the most common coarse-graining approaches is to treat the solvent particles in the colloidal suspension on a continuum level. This includes the use of implicit fluid solvers, e.g., the Lattice-Boltzmann method, and in addition the introduction of a bulk dielectric permittivity to account for the polarizability of the water molecules.

With the introduction of techniques such as the induced charge computation method, [142, 255, 265] extended Poisson-Boltzmann solvers, [316] and other functional approaches, [239, 281, 293] it has become possible to model a dielectric jump at the surface of the colloid, setting the dielectric permittivity of the colloid to a more realistic number

$\varepsilon_C = 2$, and that of the surrounding water solvent to $\varepsilon_W = 80$. It has been shown that this dielectric jump even has a minor influence on the far field electrostatic potential of the colloid, and therefore should not be neglected in simulations.[122, 275, 281, 293]

While this sharp dielectric contrast is already closer to the physical behavior of the system, it is still far from an accurate description. It has been shown that, in close proximity to a highly charged surface, solvent molecules align and form structures, reducing the strength of dielectric displacement in this region. This effect is enhanced by the presence more salt ions in the EDL, further decreasing the flexibility of the solvent dipoles and therefore the polarizability.[261, 277] While the influence of the dielectric mismatch between solute and solvent has been studied, [41, 45, 46, 237, 275, 278] research on dielectric changes within the EDL itself is still relatively new. In general, Poisson's equation combined with a Boltzmann distribution (Poisson-Boltzmann) with a space-dependent coefficient can only be solved by grid-based finite difference or finite element methods. [104, 218] These methods are computationally expensive if used in particle-based computer simulations, since the equation has to be solved billions of times per simulation.

In this section, we present a second methods capable of dealing with spatially varying dielectric permittivities that has already been mentioned for the verification in chapter 4: the harmonic interpolation method (HIM) which extends the three-layer model of biomolecular solvation, [233, 301] which uses an analytical-based Poisson solver to include an environment of spherical-symmetric dielectric functions within the EDL. Using both HIM and MEMD, we study the influence of different dielectric functions (schematically illustrated in Fig. 6.10b) on the EDL structure of colloidal systems in solution. We compare the results from the two entirely different algorithms to ensure the correctness of our findings. The results are then analyzed via the radial distribution function (RDF) and charge renormalization [44] which is useful for providing effective charges in the DLVO theory, [15, 18] and it is shown that the EDL structure and the effective charge of a colloid is modified significantly by the dielectric environment near the interface. These simulation results also demonstrate the usefulness of the developed algorithms.

In what follows, the theory of the EDL model will be first explained, and the new electrostatics algorithm, HIM, will be introduced. Next, the simulation setup will be introduced, and finally, the results of our simulations using both approaches will be presented, compared, and discussed.

### 6.3.2. Electric double layer model

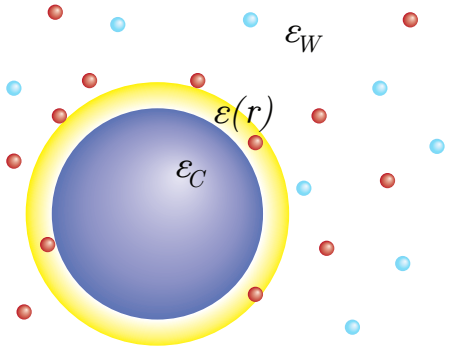
Theoretical descriptions of EDLs of charged colloids in electrolytes are often based on the primitive model. [16, 167] In this model, ions of different species are represented by charged hard spheres, differing in size and valency (Fig. 6.10a). The ions interact with each other by a combination of a short-range hard-sphere potential and a long-range electrostatic potential in a solvent which is described by its macroscopic dielectric permittivity. The equilibrium-state properties of the EDLs can be calculated by integral equation theories or Molecular Dynamics/Monte Carlo (MD/MC) computer simulations with the Hamiltonian composed of ion-ion and ion-interface interactions, and appropriate boundary conditions.

We consider a colloidal sphere of radius $R$ with surrounding electrolyte, shown in Fig. 6.10a. The dielectric permittivity is a function of radial distance, $\varepsilon(r)$, which takes a constant $\varepsilon_C$ within the sphere $r \leq R$, another constant $\varepsilon_W$ in the bulk solvent, and depends on $r$ in the intermediate region. We study the profiles shown in Fig. 6.10b. For a source point charge $q_s = 1$ at $\mathbf{r}_s$, the potential due to this charge, called the Green's function $G(\mathbf{r}, \mathbf{r}_s)$, is the solution of the following variable coefficient Poisson's equation,

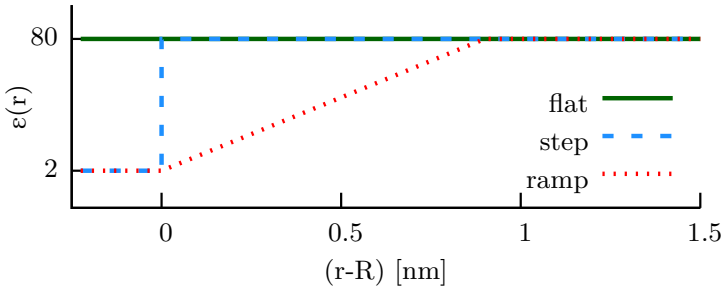
$$-\nabla \cdot \varepsilon(r) \nabla G(\mathbf{r}, \mathbf{r}_s) = 4\pi\delta(\mathbf{r} - \mathbf{r}_s), \tag{6.2}$$

where $\delta$ is the Dirac delta, and the divergence $\nabla\cdot$ and the gradient $\nabla$ are both with respect to coordinates $\mathbf{r}$. The Green's function is a function of the source point $\mathbf{r}_s$ and the field point $\mathbf{r}$, thus living on a six dimensional domain, and can therefore not be efficiently solved by numerical methods.

As will be discussed in the next section, the solution of the Green's function in a sphere-symmetric dielectric medium can be written into




**(a)** Schematic of the system




**(b)** Permittivity functions

**Figure 6.10.:** (a) The simulation setup: A charged colloid is suspended in a solvent with surrounding counterions and salt ions. The system's dielectric permittivity is divided into three regions: $\varepsilon_C$ within the colloid, $\varepsilon_W$ in bulk water, and $\varepsilon(r)$ in the intermediate region. (b) Three different models are investigated: The first model **flat** is the most simplistic, and often used, approach with a constant background permittivity $\varepsilon_C = \varepsilon_W = \varepsilon(r) = 80$ throughout the system. The second model **step** introduces a sharp contrast in form of a step function with $\varepsilon_C = 2$ within the colloid, and $\varepsilon(r) = \varepsilon_W = 80$ outside. The third model **ramp** linearly interpolates between the two regimes $\varepsilon_C = 2$ in the colloid and $\varepsilon_W = 80$ in bulk water. The interpolation occurs over 2 ion diameters, which means that the ions can enter the region of lowered dielectric permittivity. The sketches have been published in [5].

a sum of a multipole series and a direct Coulomb interaction,

$$G(\mathbf{r}_i, \mathbf{r}_j) = G_{\text{pol}}(\mathbf{r}_i, \mathbf{r}_j) + G_{\text{coul}}(\mathbf{r}_i, \mathbf{r}_j), \tag{6.3}$$

where $G_{\text{pol}}$ and $G_{\text{coul}}$ correspond to the series term and the Coulomb term shown in Eq. (6.11) below, respectively. When the Green's function is given, the electrostatic interaction between any two charges $q_i = Z_i e$ and $q_j = Z_j e$, with valencies $Z_i$ and $Z_j$, can be expressed as

$$U_{ij} = \frac{e^2}{4\pi\varepsilon_0} Z_i Z_j G(\mathbf{r}_i, \mathbf{r}_j), \tag{6.4}$$

where $e$ is the unit charge of an electron and $\varepsilon_0$ is the vacuum dielectric permittivity. Besides pairwise interactions, mobile ions have self energy fluctuations in an inhomogeneous dielectric medium. The second term $G_{\text{coul}}$ in Eq. (6.3) is divergent for the self Green's function with overlapping source and field points, which is invariant and can be absorbed into the chemical potential for uniform media. This term cannot be discarded when the medium has a varying dielectric permittivity as the solvation energy changes with it. The direct Coulomb term could be substituted by a modification of the Born energy [11] to introduce the local self energy contribution, and the self energy of ion $i$ is given by

$$U_i^{\text{self}} = \frac{Z_i^2 e^2}{4\pi\varepsilon_0} \left[ \frac{G_{\text{pol}}(\mathbf{r}_i, \mathbf{r}_i)}{2} + \frac{1}{2\varepsilon(r_i) a_i} \right], \tag{6.5}$$

where $a_i$ is the Born radius of the ion, which assumes the value of the ion radius in our simulations. The modified Born energy is a local contribution due to the finite size of the ion, and the $G_{\text{pol}}$ term is a global contribution due to the dielectric variation.

We want to highlight the significance of the second term in equation 6.5. The influence of a smoothly varying dielectric background can be split into three different contributions: Its influence on the pair interaction between point charges, as well as the first term of the self energy in equation (6.5) can be described by virtual charges or functional approaches. But the change in solvation energy is to our knowledge neglected in all other simulations, particularly steady-state

Poisson-Boltzmann solutions via finite-elements or finite-volume calculations. However, for a noticeable permittivity gradient, this term can even dominate over the other two, making it an important addition that should not be discarded.

The splitting of the self energy can be derived by a perturbation expansion, since that the Born radius is treated as a small parameter [257]. The Hamiltonian of the system is then given by, $U = U_{\text{hs}} + U_{\text{elec}}$, where

$$U_{\text{elec}} = \sum_{i<j} U_{ij} + \sum_{i=1}^{N} U_i^{\text{self}}. \tag{6.6}$$

The hard-sphere potential $U_{\text{hs}}$ is the infinite repulsion when any two spherical particles have an overlap, and otherwise does not contribute to the Hamiltonian.

One characteristic length for electrolytes is the Bjerrum length $\ell_B = \beta e^2/(4\pi\varepsilon_0\varepsilon_W)$, where $\varepsilon_0$ is the vacuum permittivity, $\varepsilon_W$ is the relative permittivity of the bulk solvent, and $\beta = 1/k_BT$ is the inverse thermal energy. The Bjerrum length is the distance between two unit charges at which they interact with the thermal energy $k_BT$. At room temperature, the Bjerrum length of bulk water solvent is $\ell_B = 0.714\,\text{nm}$.

### 6.3.3. Harmonic interpolation method for Green's function

The Green's function (6.3) can be used to investigate the physics of charged systems through particle-based computer simulations (MC or MD), which are often limited due to the lack of an efficient Poisson solver. Only limited analytical Green's function solutions exist such as for a planar or spherical interface separating two media of constant permittivity. [110] When $\varepsilon(r)$ is constant, it is known that the Green's function is the reciprocal distance divided by the dielectric, $1/(\varepsilon|\mathbf{r} - \mathbf{r}_s|)$. If the solute and solvent medium have a sharp dielectric mismatch, characterized by $\varepsilon_C$ and $\varepsilon_W$, respectively, the solution is obtained by spherical harmonics [110] or image charges. [8, 71] For a general $\varepsilon(r)$ the Green's function is not exactly solvable. However, in the following, we will develop the harmonic interpolation method

to obtain accurate approximate solutions using an almost-analytical expression, by extending the algorithm for the three-layer model of biomolecular solvation. [233]

To solve Eq. (6.3), we divide the radial distance into $L$ intervals with $L+1$ points $r_0, r_1, \cdots, r_L$ and denote the $l$th interval $(r_{l-1}, r_l)$ by $I_l$. This division could be general, but in our problem we choose the first interval to cover the region in the sphere, i.e., $r_0 = 0$ and $r_1 = R$. We approximate the dielectric function by piecewise functions, $\varepsilon(r) \approx \varepsilon_l(r)$, for $r \in I_l$. In physically realistic systems, the transition layer from $\varepsilon_C$ to $\varepsilon_W$ is less than 1 nanometer in width. It therefore usually suffices to work with a small value of $L$. The permittivity in the $l$th interval is approximated by,

$$\varepsilon_l(r) = \left(a_l + \frac{b_l}{r}\right)^2, \tag{6.7}$$

where the coefficients $a_l$ and $b_l$ are interpolated from the values $\varepsilon(r_{l-1})$ and $\varepsilon(r_l)$ at two interval ends. Clearly we have $\nabla^2\sqrt{\varepsilon_l(r)} = 0$, because the square root of the permittivity in each interval is harmonic.

Suppose the source charge resides in the $k$th interval, $r_s \in I_k$. The solution in each layer is rewritten as,

$$G(\mathbf{r}, \mathbf{r}_s) = \Phi_l(\mathbf{r}), \quad \text{for} \quad r \in I_l. \tag{6.8}$$

Here the electric potential at $l$th layer, $\Phi_l(\mathbf{r})$, should be also the function of the source position, but we dropped the dependence for notational simplicity. Since $\sqrt{\varepsilon_l(r)}$ is harmonic, by a simple derivation, the potential satisfies, [233]

$$-\sqrt{\varepsilon_l(r)}\nabla^2\left[\sqrt{\varepsilon_l(r)}\Phi_l(\mathbf{r})\right] = 4\pi\delta(\mathbf{r}-\mathbf{r}_s). \tag{6.9}$$

Since $r_s \in I_k$, we can write $\delta(\mathbf{r}-\mathbf{r}_s)/\sqrt{\varepsilon_l(r)} = \delta(\mathbf{r}-\mathbf{r}_s)/\sqrt{\varepsilon_k(r_s)}$. The equation becomes a constant coefficient Poisson's equation for $\sqrt{\varepsilon_l(r)}\Phi_l(\mathbf{r})$,

$$-\nabla^2\left[\sqrt{\varepsilon_l(r)}\Phi_l(\mathbf{r})\right] = \frac{4\pi\delta(\mathbf{r}-\mathbf{r}_s)}{\sqrt{\varepsilon_k(r_s)}}. \tag{6.10}$$

The potential function can then be expanded in terms of spherical harmonics series,

$$\begin{aligned}\Phi_l(\mathbf{r}) &= \frac{1}{\sqrt{\varepsilon_l(r)}}\sum_{n=0}^{\infty}\left[A_l(n)r^n + B_l(n)r^{-n-1}\right]P_n(\cos\theta) \\ &+\frac{\delta_{lk}}{\sqrt{\varepsilon_l(r)\varepsilon_k(r_s)}|\mathbf{r}-\mathbf{r}_s|},\end{aligned} \tag{6.11}$$

where $\delta_{lk}$ is the Kronecker delta, $\theta$ is the angle between $\mathbf{r}$ and $\mathbf{r}_s$, and $P_n(\cdot)$ is the Legendre polynomial of order $n$. We have $B_1(n) = 0$ and $A_L(n) = 0$ since the potential is finite in the innermost and the outermost layers, and other coefficients $A_l(n)$ and $B_l(n)$ are determined by the continuity conditions on each layer boundary,

$$\begin{cases}\Phi_l(\mathbf{r}_l) = \Phi_{l+1}(\mathbf{r}_l), & \text{and} \\ \varepsilon_l(r_l)\partial_r\Phi_l(\mathbf{r}_l) = \varepsilon_{l+1}(r_l)\partial_r\Phi_{l+1}(\mathbf{r}_l),\end{cases} \tag{6.12}$$

for $l = 1, \cdots, L-1$. In the case of the dielectric function being continuous, we have $\varepsilon_l(r_l) = \varepsilon_{l+1}(r_l)$.

We need to introduce the spherical harmonics expansion of the reciprocal distance,

$$\frac{1}{|\mathbf{r}-\mathbf{r}_s|} = \sum_{n=0}^{\infty}\frac{r_<^n}{r_>^{n+1}}P_n(\cos\theta), \tag{6.13}$$

where $r_<$ and $r_>$ are ordered to obey $(r_<) < (r_>)$ for $r$ and $r_s$, respectively. Then the potential can be written in a compact form,

$$\Phi_l(\mathbf{r}) = \sum_{n=0}^{\infty}M_{l,n}(r)P_n(\cos\theta), \tag{6.14}$$

with

$$M_{l,n}(r) = \frac{A_l(n)r^{2n+1} + B_l(n)}{\sqrt{\varepsilon_l(r)}r^{n+1}} + \frac{\delta_{lk}}{\sqrt{\varepsilon_l(r)\varepsilon_k(r_s)}}\cdot\frac{r_<^n}{r_>^{n+1}}. \tag{6.15}$$

The spherical harmonics are orthogonal, which means for each $n$ the boundary conditions lead to two equations for $A_l(n)$ and $B_l(n)$ at each

interface between two layers. This gives,

$$\begin{cases} M_{l,n}(r_l) = M_{l+1,n}(r_l), \\ \varepsilon_l(r_l)\partial_r M_{l,n}(r_l) = \varepsilon_{l+1}(r_l)\partial_r M_{l+1,n}(r_l), \end{cases} \tag{6.16}$$

for $l = 1, \cdots, L-1$. There are $2(L-1)$ equations for $2(L-1)$ unknowns. We thus obtain the following system of linear equations, $\mathcal{M}_n \mathbf{b}_n = \mathbf{f}_n$, where $\mathcal{M}_n$ is the coefficient matrix for the $n$th multipole term, $\mathbf{b}_n = (A_1(n), A_2(n), B_2(n), \cdots, A_{L-1}(n), B_{L-1}(n), B_L(n))^T$ with $B_1(n) = A_L(n) = 0$, and $\mathbf{f}_n$ are from the source contribution, the second term of Eq. (6.15).

Note that the matrix $\mathcal{M}_n$ is independent of the source point $\mathbf{r}_s$, and its inverse can be calculated before the running of simulations, i.e., in each step, the solution is $\mathbf{b}_n = \mathcal{M}_n^{-1}\mathbf{f}_n$ where $\mathcal{M}_n^{-1}$ is given explicitly by the Gaussian elimination and stored in memory throughout all calculations. Therefore, only operations of matrix-vector multiplication are required when the method is used in Monte Carlo simulations, and the number of operations is the order of $(L-1)^2$ for the computation of the coefficients for each $n$.

### 6.3.4. Permittivity interpolation

The MEMD algorithm and implementation is well known at this point. But we want to remind the reader how the off-lattice permittivity values are interpolated onto the mesh.

The values for $\varepsilon(\mathbf{r})$ on the links are determined as depicted in Fig. 6.11. The values for two adjacent lattice sites are averaged if their difference is less than 10 percent of their value, i.e., $2|\varepsilon_1 - \varepsilon_2|/(\varepsilon_1 + \varepsilon_2) \leq 0.1$. If the difference is larger, the link is marked as an *interface link*, meaning that is passes through a sharp interface. In a second pass, the exact intersection of these sharp interfaces with the marked links is determined, and the value is interpolated linearly

$$\varepsilon_{\text{link}} = \varepsilon_1 \cdot \frac{d_2}{a} + \varepsilon_2 \cdot \frac{d_1}{a}, \tag{6.17}$$

where $\varepsilon_1$ and $\varepsilon_2$ are the permittivity values on the adjacent lattice sites on each side of the interface respectively, $d_1$ and $d_2$ are the distances

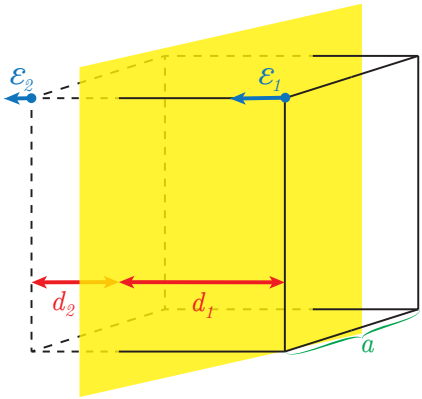



**Figure 6.11.:** Interpolation of dielectric permittivity values on the lattice. $\varepsilon(\mathbf{r})$ has a position and a direction (blue arrow). The values for $\varepsilon_1$ and $\varepsilon_2$ are determined and the value on the connecting link is set to the average value. If the gradient is too large, the link is marked as *interface link*. The values for interface links are determined by linearly interpolating the sharp dielectric interface according to its position. The sketch has been published in [5].

of the according lattice site along the link to the interface, and $a$ is the lattice spacing, as depicted in Fig. 6.11.

In this algorithm, the charges can move freely through a smoothly varying dielectric medium. The numerical error depends on the finite propagation speed of the magnetic fields and the coarseness of the linear interpolation scheme. A relative force error of $10^{-3}$, however, is achievable in sufficiently homogenous systems.

### 6.3.5. Simulation setup

We have employed canonical ensemble (NVT) Monte Carlo simulations with the HIM for electrostatics and the Metropolis algorithm, [19, 127] and equilibrium Molecular Dynamics simulations for charged systems in different media. A spherical colloid of radius $R$ is placed at the center of a simulation volume which is filled with an electrolyte solution described by the primitive model of constant dielectric permittivity (see Fig. 6.10a). The bare charge of the colloid $Q_C = Z_C e$ is assumed to be

uniformly distributed on the surface, and is equivalently placed at the center by Gauss' law. Initially, small ions are randomly distributed in the solvent. Each ion brings a charge $q_i = Z_i e$ where $Z_i = \pm Z$ and is modeled by a hard sphere of diameter $D_{\text{ion}}$. Overall the system is electrically neutral, $(N_+ - N_-)Z + Z_C = 0$. In all calculations, we take $R = 4\,\text{nm}$, $D_{\text{ion}} = 0.45\,\text{nm}$ and a large cell radius $R_{\text{cell}} = 10\,\text{nm}$ to reduce boundary effects. The simulations are performed at room temperature.

The cell in the MC simulations is modeled by a Wigner-Seitz (WS) spherical cell of radius $R_{\text{cell}} = 10\,\text{nm}$, with hard wall boundary conditions. The MD simulation cell is cubic with box length 20 nm and has periodic boundaries. We chose the cell size intentionally big compared to the Debye length of the system (between 1 and 2 nm for our salt concentrations) to avoid any influence of the difference in periodicity between HIM and MEMD.

The following three models of different dielectric environments are adopted for comparison, corresponding to Fig. 6.10b. The function is separated into three spherical regions, $\varepsilon_C$ within the colloid, $\varepsilon_W$ in the bulk, and $\varepsilon(r)$ within the solvent close to the surface.

Model 1 (**flat**) uses a homogeneous medium and the dielectric constant of the whole system is the permittivity of water, $\varepsilon_C = \varepsilon(r) = \varepsilon_W = 80$; model 2 (**step**) is the piecewise-constant dielectric model with $\varepsilon_C = 2$ within the colloid and $\varepsilon(r) = \varepsilon_W = 80$ outside; and model 3 (**ramp**) has an intermediate layer between the colloid and the bulk solvent, where the layer has thickness $2D_{\text{ion}} = 0.9\,\text{nm}$ and a linear transition dielectric permittivity, $\varepsilon(r) = (\varepsilon_W - \varepsilon_C)(r - R)/2D_{\text{ion}} + \varepsilon_C$ for $r \in [R, R + 2D_{\text{ion}}]$. For model 3 (**ramp**) in the HIM algorithm, the linear function $\varepsilon(r)$ in the transition layer is uniformly divided into 8 intervals ($L = 10$) so that the approximate Green's function solution can be obtained. The results are verified to be accurate by using division refinement. For the MEMD algorithm, the linear function $\varepsilon(r)$ in the **ramp** model is interpolated onto a rectangular lattice of mesh size $a = 0.104\,\text{nm}$.

The simulation results are measured by the macroion-microion radial distribution function (RDF) and the integrated charge distribution

function (ICDF). The RDF of each ion species is defined by,

$$g_{\pm}(r) = \frac{\langle N_{\pm}(r, r+\Delta r)\rangle}{\frac{4}{3}\pi[(r+\Delta r)^3 - r^3]}, \tag{6.18}$$

which are normalized by $\int 4\pi r^2[g_+(r)+g_-(r)]dr$, where $\langle N_{\pm}(r, r+\Delta r)\rangle$ is the mean particle number of the spherical shell between $r$ and $r+\Delta r$. The ICDF is the total charge within the sphere of radius less than $r$,

$$Q(r) = Q_M + Z_+ e\langle N_+(a, r)\rangle + Z_- e\langle N_-(a, r)\rangle. \tag{6.19}$$

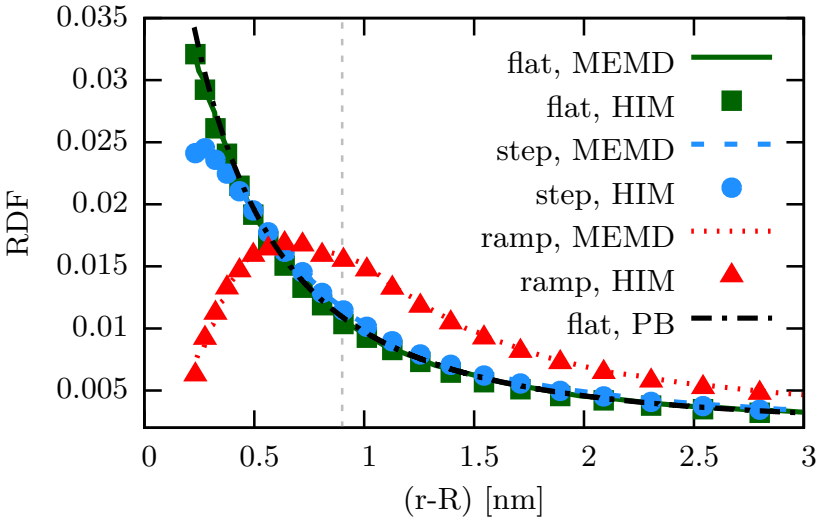
The ICDF is useful for the computation of renormalized charges.

## 6.3.6. Results and discussion

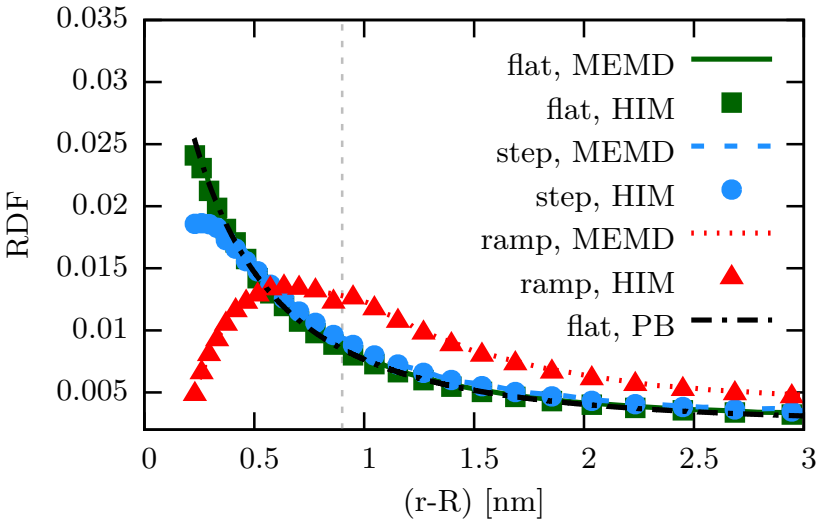
### Structure of the EDL

In the first group of simulated simulations, we set the bare colloidal charge to be $Z_C = 60$, which corresponds to a surface charge density of $0.30\,\mathrm{e/nm^2}$. We simulate a group of three systems. A number of monovalent salt ions, according to the desired salt concentration, are placed in these systems. Extra counterions are added to neutralize the macroion, such that the salt concentrations are 20, 40 and 60 mM, respectively. To ensure that the salt concentration remains the same after ion condensation on the colloid, we measure the bulk salt concentration close to the cell boundary. We arrive at 19.8, 39.6, and 59.7 mM for the **ramp** model, and comparable results for the other two models. Additionally, this confirms that the size of the simulation cell is large enough to ensure a bulk concentration at the boundary. Therefore cell boundary effects or possible influences from the two different periodicities should not play a role in our results.

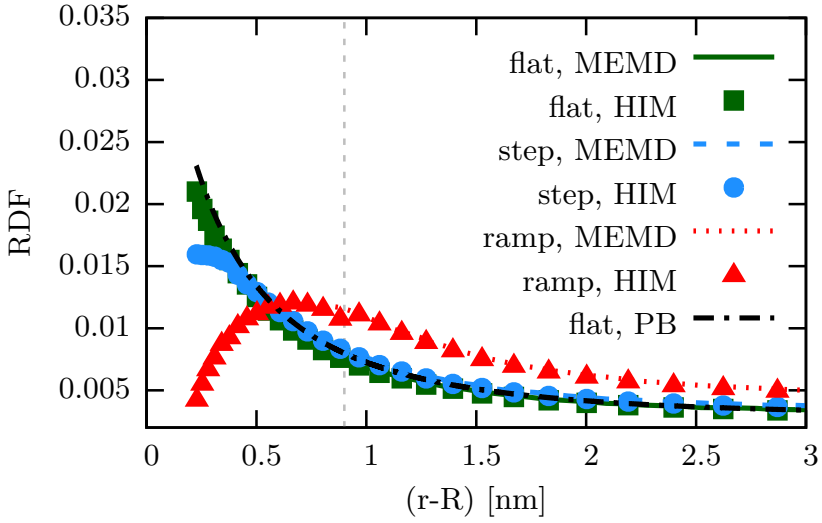
From the spherical concentration of ions, the radial distribution functions (RDFs) are calculated. The point of closest approach to the colloid is $r = 4.225\,\mathrm{nm}$, the sum of colloid and ion radius. The counterion RDF curves are illustrated in Fig. 6.10. A difference between the two different algorithms is barely noticeable. This shows that our results can be trusted, in particular for the **ramp** model, for which the results




**(a)** $c = 20\,\mathrm{mM}$




**(b)** $c = 40\,\mathrm{mM}$




**(c)** $c = 60\,\mathrm{mM}$

**Figure 6.10.:** Counterion RDF curves of 1:1 salts calculated with three different dielectric models. The salt concentrations are 20 mM, 40 mM, and 60 mM, in Figure 6.12a, 6.12b, and 6.11c, respectively. The bare surface charge is $Z_C = 60$. For comparison, we have included the solution of a numerical Poisson-Boltzmann solver for the **flat**/**step** models, and the results overlap with our simulation. In the **step** model with a sharp dielectric contrast, the counterions are repelled from the colloid surface, something that is not included in the Poisson-Boltzmann solution, widening the double layer slightly. In the **ramp** model, the difference is a lot more pronounced, since ions can enter the region of lower permittivity, increasing the solvation energy and the electrostatic ion-ion repulsion. The crossover between the linear increase and the bulk is marked with a vertical line. This data has been published in [5].

to the best of our knowledge can not easily be confirmed by any other technique.

We observe, in agreement with Messina *et al.*, [122] that the **flat** and **step** models show noticeable differences at close distances as well as in the far field, but both still result in monotonic counterion radial distributions. The **flat** model matches the theoretical prediction from a Poisson-Boltzmann solver very well. The **step** model slightly modifies the ion densities of the **flat** model near the colloid's surface, since the dielectric jump of the colloid-solvent interface creates an image charge repulsion on each mobile ion, reducing the counterion density near the surface.

The **ramp** model introduces a region of reduced dielectric permittivity around the colloid, into which the ions can enter. This significantly changes the structure of the EDL well beyond the region of varying permittivity between 0 and 0.9 nm from the surface. The RDFs show a steep increase close to the surface and reach a maximum around 0.6 nm. This can be explained by a solvation energy effect. The counterions are repelled from regions of low polarizability, which introduces a preferred motion towards the direction of $\nabla\varepsilon(r)$, as can be seen indirectly in equation (6.2). Energetically, this can be linked to an increase in solvation energy when the ion enters a region of low dielectric permittivity, since the last term in equation (6.5) dominates for small $\varepsilon$.

The RDFs of our MC simulations include a noticeable kink at the interface of linear interpolation $\varepsilon(r)$ and bulk, $\varepsilon_W$, meet at 0.9 nm (see more closely in Fig. 6.11). This is expected, since the gradient of the dielectric function $\varepsilon(r)$ has a sharp jump at this point. Ions in the region of increasing dielectric permittivity ($\nabla\varepsilon(r) > 0$) left of the jump will therefore be repelled from this jump $\nabla\varepsilon(0.9\,\text{nm}) = -\infty$. Since the dielectric function is also discretized into 8 intervals, the repulsion is not infinite but only occurs in the last interval from 0.78 to 0.9 nm. While this further reinforces the idea that sharp dielectric jumps in systems of freely moving charges are unphysical and will produce numerical artifacts, the influence in our case is very minor, as can be seen in Fig. 6.11 by comparing to the MEMD simulation results.

In comparison, the EDL structures in Figs. 6.12a, 6.12b, and 6.11c are not sensitive to the change of salt concentration. A threefold in-

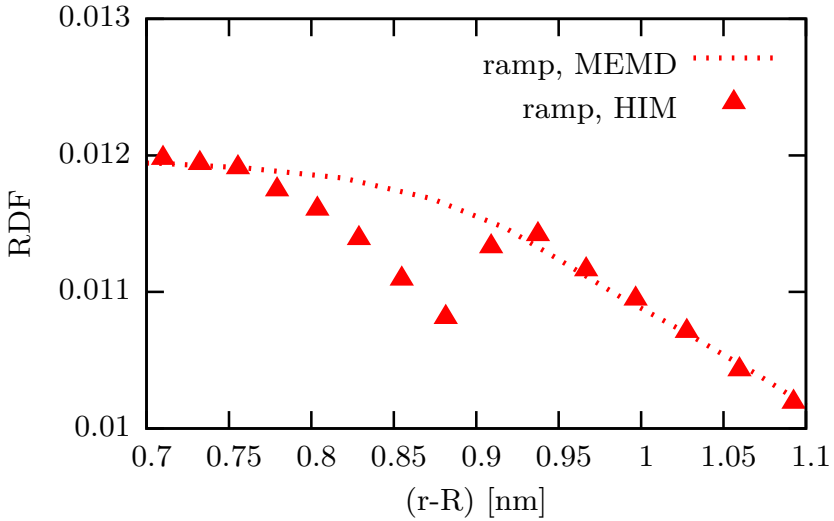



**Figure 6.11.:** Closeup of the transition region between $\varepsilon(r)$ and $\varepsilon_W$ at 60 mM salt concentration. There is a defined kink with the HIM algorithm at the boundary due to the sharp jump of the permittivity gradient $\nabla\varepsilon$ at this point. This small effect is not present in the MEMD algorithm, since the interpolation of $\varepsilon$ onto a lattice corresponds to a smoothening of the dielectric function. This data has been published in [5].

crease in the concentration (from 20 to 60 mM) only shows a small alteration of the local ionic distributions.

### Renormalized charge

In the second group of simulated systems, we investigate the effect of inhomogeneous dielectric permittivity on the colloid's effective charge by varying the bare charge. We simulated the system described above at a constant salt concentration of 20 mM, with increasing bare surface charge. The radial distribution functions for counterions and coions can be seen in Figs. 6.12 and 6.13.

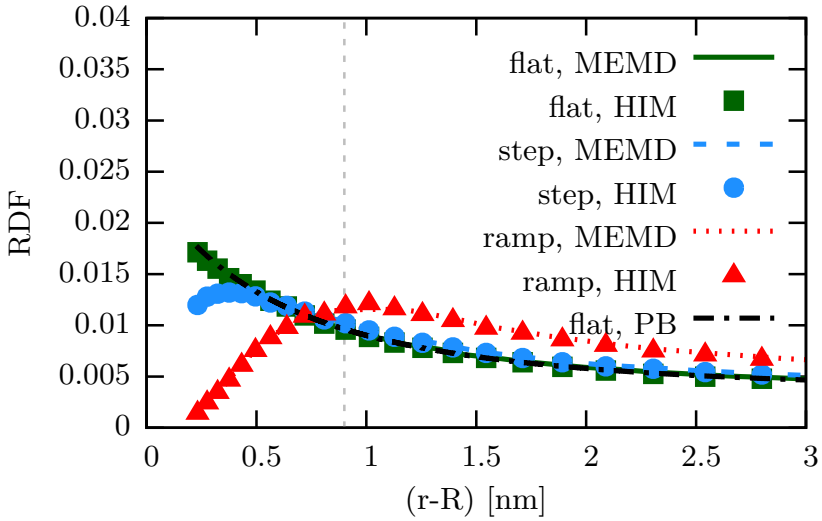
It is apparent from Fig. 6.12a that the EDL is significantly thicker at very low surface charges, which leads to a steeper increase of the electrostatic potential in the far field. The structure of the EDL itself is also modified significantly, with counter- and coions being pushed away from the colloid surface very strongly, resulting in an ion density of almost zero for lower colloid charges. To further investigate this behavior, the so-called renormalized charge was calculated by fitting an extended Debye-Hückel formula (6.20) below to the far field of the simulated ion distribution.

For charge renormalization, we use the concept from Alexander *et al.* [44] with an additional prefactor to include the colloid radius more prominently. Originally, the charge renormalization is applied to solve the nonlinear PB equation, which states that the electrostatic potential far from the surface can be described by the solution of the linearized PB equation, but with an effective renormalized charge. This renormalized charge replaces the bare charge of the colloid, which is reduced by screening effects due to the counterions. Then, the effective interaction between colloids can be calculated with the Debye-Hückel theory. [13]

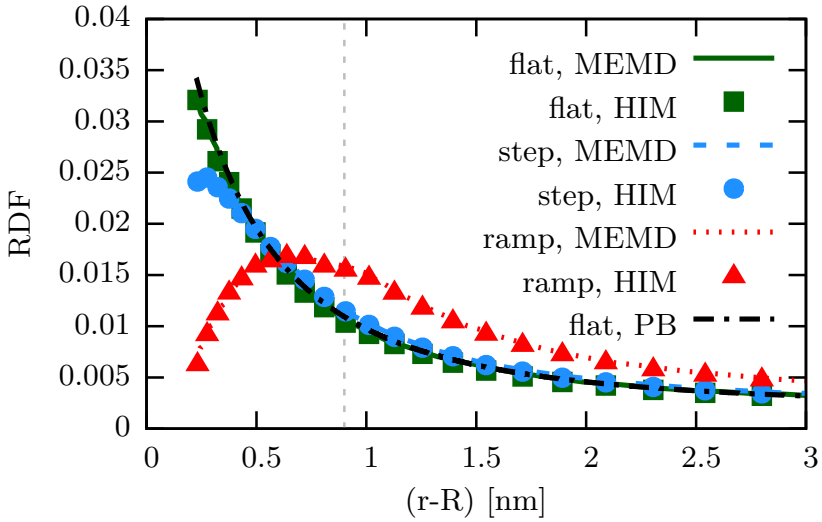
We calculate the renormalized charge through the following formula,

$$Z(r) = Z_{\text{eff}} \cdot \frac{1 + \kappa r}{1 + \kappa R} e^{-\kappa(r-R)}, \tag{6.20}$$

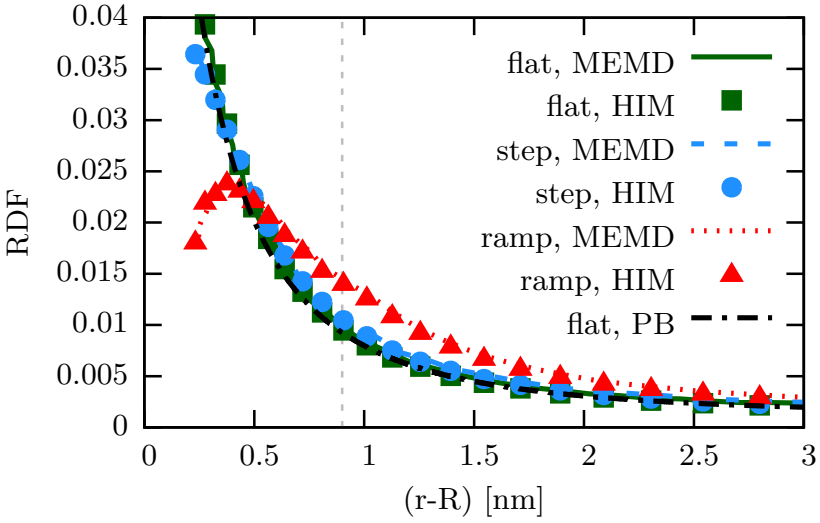
which is adapted from the linearized PB equation as presented in the original publication.[44]


RDF
(r-R) [nm]
flat, MEMD
flat, HIM
step, MEMD
step, HIM
ramp, MEMD
ramp, HIM
flat, PB
0.04
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0
0
0.5
1
1.5
2
2.5
3

**(a)** $Z_C = 30$


RDF
(r-R) [nm]
flat, MEMD
flat, HIM
step, MEMD
step, HIM
ramp, MEMD
ramp, HIM
flat, PB
0.04
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0
0
0.5
1
1.5
2
2.5
3

**(b)** $Z_C = 60$




**(c)** $Z_C = 90$

**Figure 6.12.:** Counterion RDF curves of 1:1 salts calculated with three different dielectric models. The salt concentration is 20 mM, and the bare surface charges are 30, 60, and 90, in Fig. 6.12a, 6.12b, and 6.11c, respectively. The crossover for the **ramp** model between the linear increase and the bulk is marked with a vertical line. For very low surface charges, the counterion distribution and therefore the Debye Layer are quite thick, reaching the counterion maximum only at 1 nm from the surface at $Z_C = 30$. This leads to a comparably steep increase in the electrostatic potential in the far field. This data has been published in [5].

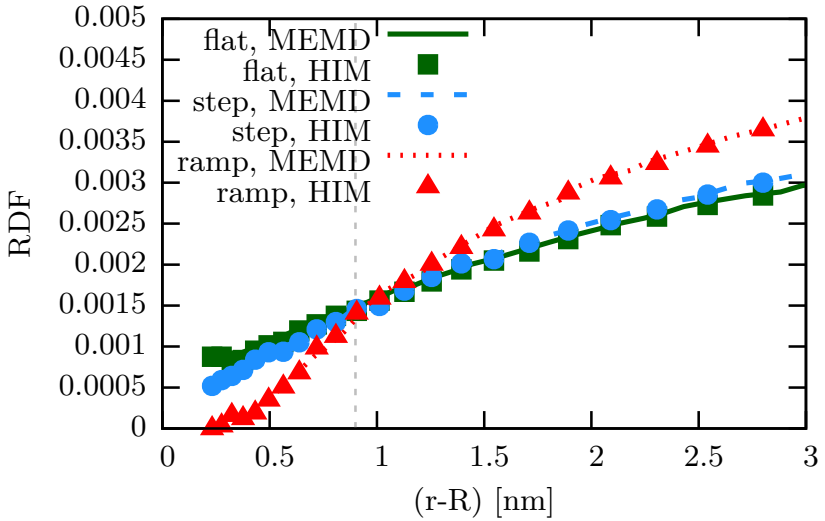



**Figure 6.13.:** Coion RDF curves of 1:1 salt calculated with three different dielectric models. The salt concentration is 20 mM, and the bare surface charge is 30 e. Where coions are able to enter the close vicinity of the colloid surface in the **flat** and **step** models, especially for low bare surface charges, the additional dielectric repulsion in the **ramp** model prevents this, forcing the coion density to zero at the surface. This data has been published in [5].

For dilute monovalent salts, the linearized PB equation holds in the region where the distance to the surface is larger than the Gouy-Chapman length, [162] $\ell_{\mathrm{GC}} = 1/(2\pi Z \ell_B \sigma_S)$ where $\sigma_S = |Q_M|/4\pi R^2$ is the surface charge density. To ensure the validity of the fit, we discarded data points closer to the surface than $2\ell_{\mathrm{GC}}$ and had very stable results from a least squares fit.

The simulation systems have distinct bare charges starting from $Z = -10$ to $-90$, which corresponds to a surface charge density varying from 0.05 to 0.45 e/nm$^2$. The salt concentration is fixed at 20 mM. The renormalized charge is calculated by fitting eq. (6.20) via $\kappa$ and $Z_{\mathrm{eff}}$ for $r - R$ from 1.5 nm. In addition to the residuum of the least squares fit routine, the resulting values of $\kappa$ are a good control of the fit quality when compared to the calculated theoretical value, and they match very well.

The calculated effective charge with fit errors is plotted against the bare colloid charge in Fig. 6.14. For comparison, the theoretical prediction for a constant background permittivity as proposed by Aubouy *et al.* [138] is included in the plot. It follows the formula

$$Z_{\mathrm{eff}} = \frac{R}{\ell_B}\left[4\kappa R t_Q + 2\left(5 - \frac{t_Q^4 + 3}{t_Q^2 + 1}\right) t_Q\right], \tag{6.21}$$

where the bare charge is included via $t_Q = (\sqrt{1+x^2} - 1)/x$ with the substitution $x := Z_{\mathrm{bare}}\ell_B/2R(\kappa R + 1)$.

In the renormalized charge, and therefore the far-field interaction, there is a minor difference between the primitive **flat** model and the **step** model with a sharp dielectric jump. We confirm the findings in ref. [135] that the effective screening of the bare colloidal charge is slightly less effective, since the counterions are repelled from the colloid surface and the EDL is widened. The difference to the newly proposed **ramp** model, however, is far greater. It can already be seen in Fig. 6.10 that the width of its EDL is significantly larger than those of the other two models. While the effective charge for the **flat** and **step** models ranges between 73% and 99% of the bare charge, the **ramp** model predicts much higher values between 86% and 117%. This means that for surface charge densities of less than 0.3 e/nm$^2$, the effective charge of

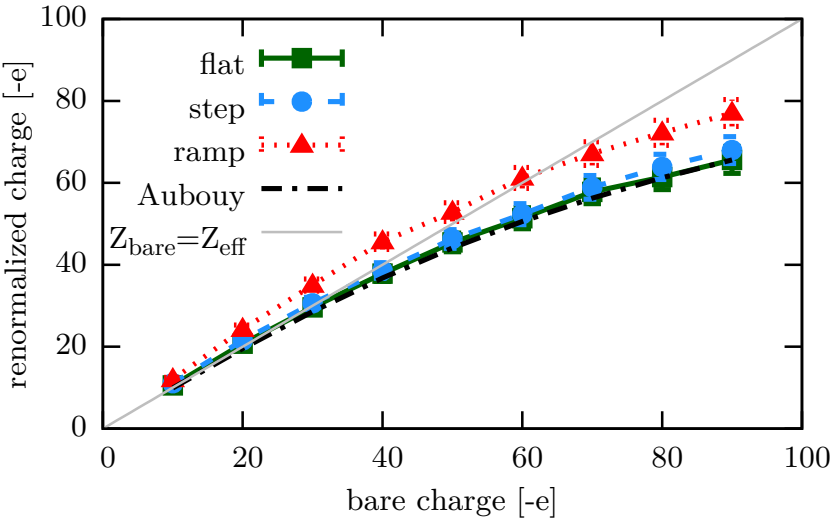



**Figure 6.14.:** Renormalized charge versus the bare colloid charge for the three different dielectric models, as well as the results of a theoretical prediction, for a 1:1 electrolyte of 20 mM. The **flat** and **step** models are very similar, and match the theory in the applicable region. The **ramp** model deviates from these significantly and demonstrates the fundamental influence of dielectric changes within the double layer. The counterions are repelled so strongly that, for low surface charges, the renormalized charge of the colloid is actually increased due the colloid appearing to have a larger effective radius. This data has been published in [5].

the colloid as defined in equation (6.20) is higher than the bare charge. The EDL is widened significantly, and the counterion maximum is as far as 1 nm from the colloid surface, as can be seen in Fig. 6.12a. This means that compared to Debye-Hückel theory, the electrostatic potential actually increases more steeply at far distances, which leads to an increase of the effective charge. This can be explained via an effective colloid radius. It is apparent from equation (6.20) that the fitted effective charge will be larger if the colloid exhibits an effective radius that is larger than $R = 4.225\,\mathrm{nm}$, as inserted into the formula. And with the counterions being pushed out very far, the far field will resemble that of a colloid with an increased effective radius. While effective charges higher than the bare colloid charge are unintuitive at first, they can occur with our definition of the charge renormalization and are meaningful. The **flat** model matches the theoretical curve closely as expected, since the theory does not include dielectric effects. All three models converge towards the line $Z_{\mathrm{bare}} = Z_{\mathrm{eff}}$ for very low surface charges, as expected.

From the data in Fig. 6.14, a shift $\Delta Z_{\mathrm{eff}}$ in comparison to the **flat** model can be calculated and with equation (6.20) interpreted as a change in effective colloid radius. With these calculations, the effective colloid charge corresponds to the radius being widened by 0.37 nm, 0.29 nm, and 0.19 nm, for $Z_{\mathrm{bare}} = 30, 60$, and 90, respectively. This is a reasonable estimate when compared to the EDL structure in Fig. 6.12. This virtual effective colloid size is not an observable change in diameter, but a makeshift parameter that is usually included in the effective charge. In this case however, it explains why the calculated effective charge can be higher than the bare charge of the colloid.

### 6.3.7. Conclusions

The influence of charges in the dielectric background to the effective interaction between macroions has been subject to research for quite some time. Until now, it was not possible to directly include smooth changes of the dielectric permittivity in regions where charges can move freely. With the two recently introduced algorithms presented in this work, the long-range effect of these changes has been investigated. Both

approaches give identical results.

We compared the electric double layer (EDL) structure for a charged colloid in electrolyte solution with three different models of the surrounding dielectric properties. The last of these models, a linear increase of the permittivity from the colloid surface, has not been accessible to coarse-grained simulations up until now for the lack of a suitable electrostatic solver. The impact of this model is significant on the counterion profile within the EDL. It is even more pronounced in the difference between the renormalized charge calculations according to the Alexander prescription. The results indicate that a spatially varying permittivity $\varepsilon(r)$ likely plays a significant role in other biophysical systems, and that the presented algorithms should be applied more widely.

## 6.4. Polyelectrolytes in solution

The results presented in this section are published in [7] and [6], and some graphs and formulations were used in both works.

### 6.4.1. Introduction

Electrophoresis is the directed motion of an object in an aqueous solution subject to an external electric field. Electric fields are often used for the characterization, purification, and manipulation of polymers [119, 242, 287], colloids [26, 34, 61, 196], and cells [120, 243], which all tend to ionize in aqueous solutions. While it is tempting to understand electrophoresis as the balance between the external electric force and hydrodynamic friction, the physics is somewhat more complicated [184]. The reason for this is the aggregation of oppositely charges ions, termed counterions, near the surface of the object. The ions closest to the surface make up the Stern layer, strongly adsorbed ions which co-move with the electrophoresing object, effectively reducing the charge of the object. Ions further away from the surface form the diffuse layer, which has a characteristic size given by the Debye

length, $\lambda_D$:

$$\lambda_D = \left( 4\pi\lambda_B \sum_{j=1}^{N} n_j z_j^2 \right)^{-1/2}, \tag{6.22}$$

where $N$ is the total number of charge species, $n_j$ and $z_j$ are the number concentration and valency of species $j$. The Bjerrum length $\lambda_B$ is given by:

$$\lambda_B = e^2/(4\pi\varepsilon_0\varepsilon_{rs}k_B T), \tag{6.23}$$

where $\varepsilon_0$ and $\varepsilon_{rs}$ are the vacuum and the relative permittivity, respectively.

The ions in the diffuse layer move in the opposite direction of the analyte since they are oppositely charged. The counterions reduce the velocity of the electrophoresing object, through hydrodynamic and electrostatic coupling to the analyte. The plane separating these two layers is called the shear plane, and the two layers are often called electrical double layer (EDL).

Simulations of polyelectrolyte electrophoresis have, for the most part, focused on the electrophoresis of a single chain in bulk solution (free solution electrophoresis). The first results in this regard were lattice-Boltzmann simulations by Grass and Holm [212, 213, 228, 247], and multi-particle collision dynamics simulations by Frank and Winkler [210, 227]. Both methods were able to reproduce the sharp rise in electrophoretic mobility as a function of the polymer length for short chains, followed by a slight decrease for longer chains, as is seen in experiment [108]. In contrast, when long-ranged hydrodynamic interactions were ignored, the simulations showed a monotonic decrease in the mobility in stark contrast to experiment. The reason for the monotonic decline in mobility in the absence of hydrodynamic interactions is that a fraction of the counterions strongly bond to the polymer backbone, co-moving with the chain, and effectively reducing the linear charge density [227, 228, 280]. This phenomenon, termed counterion condensation, was first proposed by Manning [27].

A sharp increase in the mobility with increasing chain length is observed for short chains. This increase can be understood through the

cooperative shearing of the fluid within the Debye layer [287]. Because of this, the mobility starts to flatten out when the chain length becomes comparable to the Debye length [228]. The aforementioned condensation of counterions and subsequent reduction of the effective charge of the chain is responsible for the slight decrease in mobility for long chains [227, 228, 280].

A number of studies have also looked at the electrophoresis of polyelectrolytes through simple channels. Smiatek and Schmid investigate the role of surface slip on the mobility of a polyelectrolyte electrophoresing between parallel plates [253, 271]. More recently, Yan et al looked at the electric field dependent mobility of a polyelectrolyte between parallel plates [290]. The electrophoretic motion of chains through entropic traps [115] has also been studied extensively using computer simulations [195, 208, 209, 244]. A number of studies have also looked at the electrophoresis of DNA in post arrays [240, 242, 254].

Electric fields are also often applied to polyelectrolyte solutions in order to measure their conductivity and characterize the properties of the solution [27, 29, 31, 81, 93, 143, 161, 231, 303]. Theory is able to correctly predict the conductivity under a variety of solvent conditions and salt concentrations [27, 81, 303]. The conductivity normalized by the concentration of charge carriers is called the equivalent conductivity and is often used instead of the raw conductivity. Deviations from this linear scaling, due to both electrostatic and hydrodynamic interactions, causes the variations in the equivalent conductivity with respect to the concentration. In the salt-free case, there are a number of experiments which show an initial decrease in the equivalent conductivity, followed by an increase starting at polyelectrolyte concentrations of approximately $c \approx 0.01M$ [31, 93, 231]. Scaling theories [81, 93] are able to correctly predict the initial decay, but fail to explain the increase in the equivalent conductivity at polyelectrolyte concentrations above $c \approx 0.01M$ [93]. This increase in the equivalent conductivity appears to be independent of the type of polyelectrolyte used [31, 93, 231], the length of the polyelectrolytes [93], and the temperature of the solvent [231].

Generally, the salt concentration has a strong influence on the response of soft matter systems to an external electric field, as it deter-

mines the Debye length [184]. The Debye length not only influences the qualitative behavior of the mobility, but also the quantitative values. The electrophoretic mobility is directly proportional to the electrostatic potential difference between the shear plane and the bulk, called the zeta potential. Decreasing the salt concentration increases the Debye length, causing the diffuse layer to thicken, which in turn leads to a dramatic increase in the zeta potential.

In most theories and simulations on charged systems the presence of water molecules is taken into account as a constant dielectric background. This is most often done by setting a constant relative permittivity of the system $\varepsilon_\mathrm{r} = 78.5$, corresponding to salt-free water at 10 °C. However, it has been shown that the presence of charged salt ions [176, 217] and charged surfaces [261] can considerably alter the local dielectric constant. Recent theoretical work has shown that taking into account the smoothly changing dielectric properties between a spherical colloid and the surrounding fluid causes a depletion in counterions near the surface [296, 5, 305, 308]. This effectively increases the thickness of the Debye layer and in turn leads to an increase in the zeta potential. Presumably, a similar effect occurs for polyelectrolytes, though the magnitude of the effect is, a priori, unclear. Furthermore, the change in the distribution of the ions around the polyelectrolyte will also undoubtedly have an influence on its electrophoretic mobility. Only a few techniques exist for calculating ion distributions in the presence of a varying dielectric constant [255, 281, 293], but it is in general non-trivial to couple this to a hydrodynamic description of the surrounding fluid.

An additional challenge is that the influence of a varying dielectric background consists of three separate contributions: The dielectrically reduced pairwise electrostatic interaction between charges can be described by the aforementioned algorithms, as well as the influence of dielectric enclosures on a point charge by means of virtual boundary or image charges [8, 71, 233, 250, 255, 265], or via a functional approach [239, 281, 293]. But in addition to these two contributions, mobile ions have self energy fluctuations in an inhomogeneous dielectric medium. In uniform dielectric media, the solvation energy of charged particles is invariant and can be absorbed into the chemical poten-

tial. However, this term cannot be disregarded when the solvent has a varying dielectric permittivity. As pointed out in section 6.3, this direct Coulomb term can be described by a modification of the Born energy [11] by introducing the local self energy contribution. The self energy of ion $i$ is given by

$$U_i^{\text{self}} = \frac{q_i^2}{4\pi\varepsilon_0}\left[\frac{G_{\text{pol}}(\mathbf{r}_i,\mathbf{r}_i)}{2} + \frac{1}{2\varepsilon(\mathbf{r}_i)a_i}\right], \tag{6.24}$$

where the first term including $G_{\text{pol}}$ is the contribution from virtual surface or image charges, and the second term relates to the solvation energy within the medium. Here, $q_i$ is the charge and $a_i$ is the Born radius of the ion. If the gradient of the locally varying permittivity $\varepsilon(\mathbf{r})$ is non-zero, this energy leads to a force that is included in the HIM method presented earlier and the MEMD algorithm.

## 6.4.2. Simulation Method

### Molecular Dynamics

We perform Molecular Dynamics (MD) simulations using the Extensible Simulation Package for Research on Soft matter, ESPResSo [181, 1, 315]. There is a purely repulsive Weeks-Chandler-Anderson potential [28] between all particles to represent steric interactions:

$$V_{\text{WCA}}(r) = \begin{cases} 4\varepsilon_{\text{MD}}\left(\left(\frac{\sigma_{\text{LJ}}}{r}\right)^{12} - \left(\frac{\sigma_{\text{LJ}}}{r}\right)^{6} + \frac{1}{4}\right), & r < 2^{1/6}\sigma_{\text{LJ}}, \\ 0, & \text{otherwise}, \end{cases} \tag{6.25}$$

where $r$ is the distance between the particles, $\sigma_{\text{LJ}} = 0.3\,\text{nm}$ is the fundamental MD length scale, and $\varepsilon_{\text{MD}} = k_{\text{B}}T = 4.11 \times 10^{-21} J$ is the fundamental MD energy scale.

$M_{\text{p}}$ polyelectrolytes consist of a linear chain of negatively charged monomers $q_1 = -1e$ (where $e$ is the fundamental MD unit of charge) of length $N$. Adjacent monomers are linked together using a finitely-extensible nonlinear elastic (FENE) potential

$$V_{\text{FENE}}(r) = -\frac{kR_0^2}{2}\ln\left(1 - \left(\frac{r}{R_0}\right)^2\right), \tag{6.26}$$

where $R_0 = 1.5\sigma_{\mathrm{LJ}}$ is the maximum extension of the bond and $k = 30\varepsilon_{\mathrm{MD}}/\sigma_{\mathrm{LJ}}^2$ is the energy scale of the bond. An equal number $M_{\mathrm{p}}N$ of counterions with charge $q = 1e$ are added to the box to render the net charge of the system zero. All particles have a mass $m = 1m_0$, where $m_0$ is the fundamental MD unit of mass. Velocities and positions are updated using the velocity Verlet algorithm with a time step $\Delta t_{\mathrm{MD}} = 0.01\tau$, where $\tau = \sqrt{m_0\sigma_{\mathrm{LJ}}/\epsilon}$ is the MD unit of time. In simulations where we measure static properties, we use a Langevin thermostat with a temperature $k_{\mathrm{B}}T = \varepsilon_{\mathrm{MD}}$ and a friction constant $\Gamma_{\mathrm{Langevin}} = 1m_0/\tau$. All of our simulations used periodic boundary conditions in all directions.

### Hydrodynamic Interactions

In simulations where we calculate dynamic properties, we chose the lattice-Boltzmann (LB) method to model the hydrodynamic interactions, since it is extremely efficient and has been shown to produce physically sound results [196, 213, 306, 309]. Specifically, we use ESPResSo's D3Q19 lattice-Boltzmann (LB) [285] with a kinematic viscosity $\nu = 0.8\sigma_{\mathrm{LJ}}^2/\tau$ and density $\rho_{\mathrm{fluid}} = 1m_0/\sigma_{\mathrm{LJ}}^3$. The LB grid spacing is $a_{\mathrm{LB}} = 0.4\,\mathrm{nm}$ and the time step was set to $\Delta t_{\mathrm{LB}} = 0.01\tau$. These parameters will give the correct hydrodynamic radius of $0.44a \approx 0.15\,\mathrm{nm}$ for counterions while keeping the lattice-Boltzmann algorithm stable. We applied an external electric field of $E_{\mathrm{ext}} = 0.1\varepsilon_{\mathrm{MD}}/\sigma_{\mathrm{LJ}}e$, which has been shown to be low enough to still be in the linear response regime [149, 228], when measuring mobilities and conductivities. The temperature was held constant using an LB thermostat with a temperature $k_{\mathrm{B}}T = \varepsilon_{\mathrm{MD}}$.

### Electrostatics

To compute the electrostatic interactions on a varying dielectric background, we use our MEMD algorithm. We have successfully applied this implementation to a colloid with a smooth radial change of the dielectric permittivity in the vicinity of the colloid surface [5] in section 6.3. We showed that a region of varying dielectric permittivity,

into which charges can enter, gives rise to a strong force in the direction of the permittivity gradient and therefore significantly influences the structure of the electric double layer (EDL) around the colloid. This force is the result of a finite sized object in a permittivity gradient, that from the self-energy. This self-energy term is missing from methods which only include a discrete jump in the dielectric constant and significantly affects the equilibrium distribution of ions. We also compared our implementation to a Monte Carlo code that can deal with radially symmetric and smooth permittivity gradients and found excellent agreement, giving us confidence on the reliability of the algorithm for spatially varying dielectric permittivity. The MEMD grid spacing was set to $a_{\mathrm{MEMD}} = 0.4\,\mathrm{nm}$, the time step was $\Delta t_{\mathrm{MEMD}} = \Delta t_{\mathrm{MD}}$, and the artificial mass was $f_{\mathrm{mass}} = 0.05 m_0$.

### 6.4.3. The influence of charges on dielectric permittivity

In MD simulations of polyelectrolytes, an implicit water model is often used since explicit water molecules would increase the necessary compute time by more than an order of magnitude [234]. To get reliable results in the case of electrophoresis, the implicit water model needs to correctly reproduce two properties of water: the hydrodynamic interactions (which we deal with using lattice-Boltzmann as explained in section 6.4.2) and the screening of electrostatic interactions. Most MD simulations deal with the screening of electrostatics by introducing a bulk dielectric permittivity $\varepsilon$. This shows up as a constant prefactor in the Poisson equation, and since it does not depend on any local parameters, can simply be included in the electrostatic force calculation using the same prefactor.

This electrostatic screening effect originates in the rearrangement of water molecule dipoles in the presence of an electric field $\mathbf{E}$. On average, the dipoles will have a tendency to align with the electric field lines and create an additional field in the opposite direction. The resulting electric field $\mathbf{E}(\mathbf{r})$, at a position $\mathbf{r}$, will be weakened by a factor of $\varepsilon(\mathbf{r})$, which leads to the definition and use of the displacement field $\mathbf{D} = \varepsilon(\mathbf{r})\mathbf{E}$. As implied by the notation $\varepsilon(\mathbf{r})$, this prefactor is not necessarily constant throughout the system but can depend on

the local surroundings. Specifically, if there are charged molecules like polyelectrolyte monomers or salt ions present, the water molecules in the vicinity of these charges will be restricted in their rotational degrees of freedom by the strong electric field and can not react to external electrostatic influences as freely as bulk water molecules. This means that the dielectric response in the presence of salt ions is reduced. Several studies have examined the dependence of the bulk permittivity on the salt concentration in water [96, 174, 176, 177], as well as the variation in the permittivity around a charged object [261].

Most current electrostatics algorithms only allow for a change in the system-wide dielectric permittivity, and this parameter can be adjusted in polyelectrolyte simulations in an attempt to account for salt concentration in the solution, an approach we also investigate in this manuscript. However, it is apparent that the permittivity is not constant in a system that includes a highly charged object surrounded by counterions, and a bulk phase that contains mostly solvent and relatively few co- and counterions. This is particularly true for a dilute polyelectrolyte solution. The local ion concentration will not only vary in the proximity of the polyelectrolyte, but also in time since the polyelectrolyte is mobile. Here, we will apply the MEMD algorithm with temporal and spatial variations in the dielectric permittivity, and examine their influence on the static and dynamic properties of polyelectrolyte solutions.

### 6.4.4. Iterative approach

As a starting point, we consider a polyelectrolyte to be a charged rod, or in our case a linear arrangement of charged beads fixed in space as shown in Fig. 6.15. The rod consists of $N = 80$ monomers which are fixed in place for the duration of the simulation and spaced 0.3 nm apart in a box cubic with a side length of 24 nm with periodic boundary conditions. Inside the cylinder representing the polyelectrolyte, there are very few water molecules and the dielectric permittivity is set to $\varepsilon = 2$[48, 164]. Outside the polyelectrolyte, we expect the dielectric permittivity to depend on the local concentration of counterions, and gradually increases to around $\varepsilon = 78.5$ in the bulk.

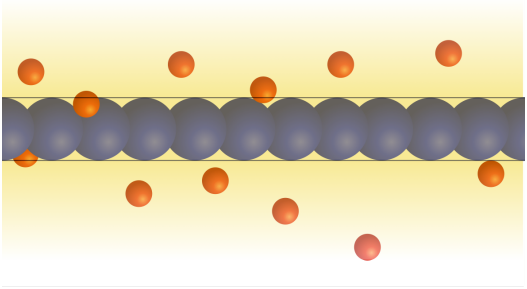


**Figure 6.15.:** Schematic of our first simulation setup: a fixed line of monomers (blue) is placed in the center to form a stiff rod and counterions (red) can move freely around the polyelectrolyte. The permittivity (yellow background) is set to $\varepsilon = 2$ within the rod and is variable outside the rod, eventually reaching $\varepsilon = 78.5$ (white) at longer distances.

If we average the salt concentration over time, we should get a cylindrical ion profile around the rod center because of the system's symmetry. This amounts to a one-dimensional problem to solve, and there are certainly faster and more efficient ways to approach it than an MD simulation. However, we wanted to introduce a verification for our algorithm, and we wanted a point of reference for later comparison when the dielectric constant varies both spatially and temporally. To this end, we will solve for the electrostatic forces using the implementation of MEMD with varying dielectric permittivity in ESPResSo.

To calculate the local permittivity value from the local ion concentration, we require a function that maps a given salt concentration to a permittivity value. For our simulations, we use the empirical function obtained by Hess et al. [176] from atomistic MD simulations of a sodium chloride salt solution:

$$\varepsilon = \frac{78.5}{1 + 0.278 \cdot C} \quad , \tag{6.27}$$

where $C$ is the salt concentration in moles per liter [M].

We self-consistently solve for the counterion distribution and the dielectric constant using an iterative scheme. We start out with a flat permittivity profile, which assumes a constant $\varepsilon = 78.5$ for the sur-

rounding solvent. We then map this salt concentration to a dielectric permittivity via equation (6.27), and interpolate the resulting permittivity distribution to a grid. We then set the permittivity values of the lattice links and, after equilibration, run the simulation until the average radial counterion distribution converged. We use a successive under-relaxation, where the two preceding ion distributions are averaged to stabilize the iterative scheme. The averaged ion distribution is then used to generate new permittivity values and the process is repeated iteratively until both the ion distribution and permittivity converge. The results of the first eight runs are presented in Fig. 6.16.

The simulations clearly converge towards a stable final distribution, and for the simulation shown in Fig. 6.16 it is converged after the eighth iteration. The converged counterion density actually increases in the first two nanometers, before an exponential decrease from two nanometers to around five nanometers from the rod. This roughly corresponds to the region of reduced dielectric permittivity shown in the bottom graph of Fig. 6.16, which also extends approximately five nanometers from the rod.

The counterion distribution in Fig. 6.16 looks similar to our findings for spherical colloids [5] in section 6.3, which is to be expected given that the underlying physics is independent of the specific geometry. Because of these similarities and the very stable final distribution, we are confident that our simulation method produces a realistic counterion distribution for a stiff rod model including varying permittivity effects. However, this method is time consuming and not suitable for dynamic simulations since the dielectric constant does not vary temporally. Nonetheless, it does provide a good point of comparison for the adaptive approach introduced in the next section.

### 6.4.5. Adaptive approach

To gain a more flexible and widely applicable algorithm, we opted to calculate the instantaneous local salt concentration during the simulation. We again simulate a stiff rod consisting of $N = 80$ monomers fixed in space with a distance 0.3 nm between monomers, in a cubic box with a side length of 24 nm with periodic boundary conditions.

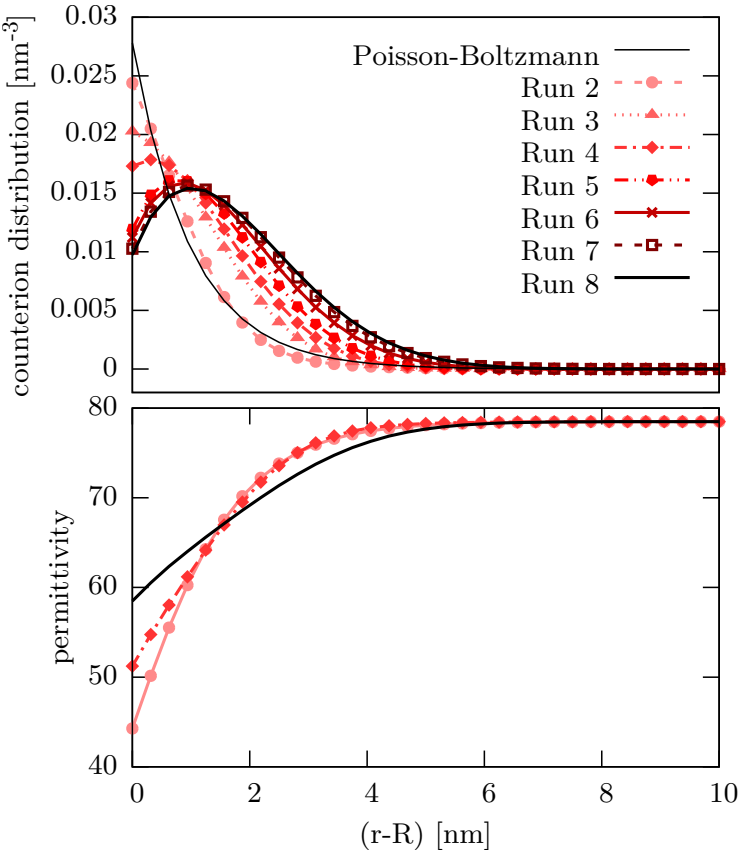



**Figure 6.16.:** Starting from a flat permittivity profile of $\varepsilon = 78.5$, the counterion distribution including a salt concentration dependent permittivity around a charged stiff rod is calculated iteratively. The counterion distribution converges to a stable solution (Run 8, top graph) which is significantly different from the often applied Poisson-Boltzmann solution (thin black line, top graph). The first iterative run expectedly gives a result identical to the Poisson-Boltzmann distribution. The dielectric permittivity, calculated from equation (6.27) including an additional contribution from the polyelectrolyte charge, shows a weaker attraction between the counterions and the rod (bottom graph).

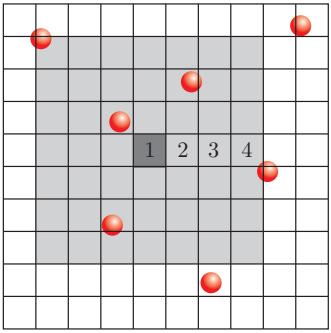



**Figure 6.17.:** To calculate a local salt concentration on the fly, we use a weighted average in the surrounding $7^3$ lattice cells. The weights are equal to the inverse of the shell number, meaning the central cell gets a weight of 1, the second shell a weight of $1/2^2$, the third shell a weight of $1/3^2$ and the outermost cells getting a weight of $1/4^2$. The result can then be mapped to a dielectric permittivity for the center cell using equation (6.27).

For each lattice cell we calculate a weighted average of the ion concentration within a predefined cube, as shown in Fig. 6.17. If the mesh is chosen too coarsely or the screening length too small, there will be large jumps in the permittivity when ions enter or leave the cube. This should be avoided since dielectric jumps are physically unrealistic and represent barriers that block ion flow almost entirely. For our system, we have chosen a cube size of $7^3$ lattice cells. Given the fine mesh of the simulation, this corresponds to a reach of 1.4 nm or $2l_B$ (Bjerrum lengths) in water, which is a realistic screening length and we found it to result in sufficiently smooth permittivity curves. The charges included in the surrounding lattice cells are weighted by the inverse square of the shell number (see Fig. 6.17). This represents a $1/r^2$ influence, in accordance with the assumption that the polarization response is linear to the electric field, which also decays as $1/r^2$.

For verification of this new adaptive scheme to calculate the local charge concentration, we again simulated the system sketched in Fig. 6.15. We did not manually set the dielectric permittivity within the polyelectrolyte and ran the simulation starting with a random dis-

tribution of counterions. The result and comparison to the iterative approach is shown in Fig. 6.18.

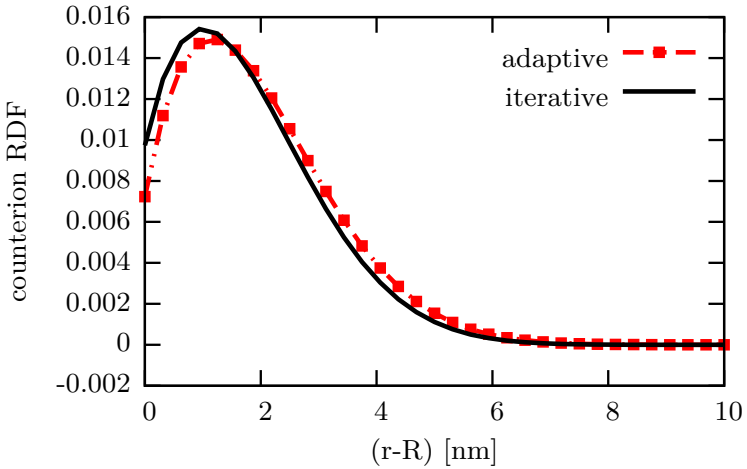



**Figure 6.18.:** Comparison of the newly adopted scheme to the final distribution reached with our iterative approach. The results are almost identical.

The solution provided by our new adaptive scheme results in almost the exact same structure of the electric double layer. Instead of a monotonic exponential decrease, the counterions are pushed away from the polyelectrolyte and reach a stable maximum at around 1.1 nm from the surface. This shows that the counterion distribution is not sensitive to the exact method used to calculate the local dielectric constant, which indicates that our somewhat ad hoc method of calculating the local ion concentration correctly captures the underlying physics.

### 6.4.6. Flexible Polyelectrolyte

With this new adaptive scheme, we are now able to fully simulate a flexible polyelectrolyte in aqueous solution with the local dielectric constant dependent on the local salt concentration. The simulation is still computationally demanding, since the MEMD lattice has to be very fine to accurately represent the changes in dielectric permittivity

around the polyelectrolyte, and because of the calculation of the salt concentration for every cell using the surrounding $7^3$ neighbor lattice sites. It is, however, far less costly than simulations with explicit water representation. Depending on the number of processors, it runs about a factor of 2 to 3.5 slower than simulation with constant background permittivity.

For a flexible polymer, it is not as straight forward to obtain a radial distribution function of counterions. But for comparison, we adopted the calculation presented in Fig. 6.19.

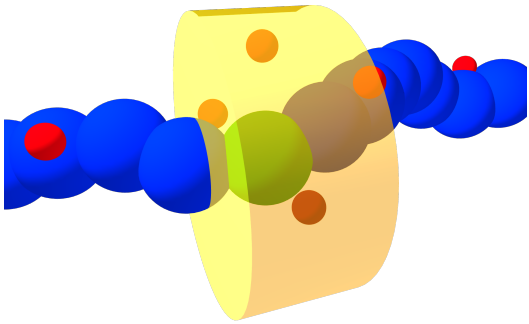


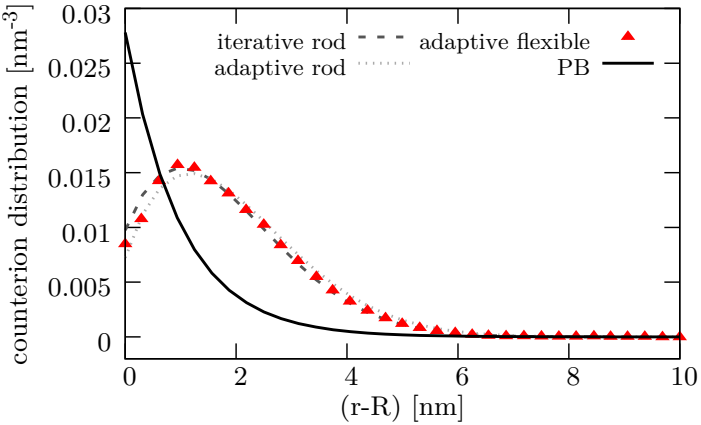
**Figure 6.19.:** To calculate the radial distribution function of counterions for a flexible polyelectrolyte, we form a cylinder around each monomer, the axis of which is parallel to the connection between the two direct neighbor monomers.

For each monomer, we calculate the vector between the centers of the two neighboring monomers to which it is bonded. This vector is then shifted to the monomer's center. Finally, we calculate the radial distribution function in a cylinder of length $2d$, where $d$ is the equilibrium distance between two monomers.

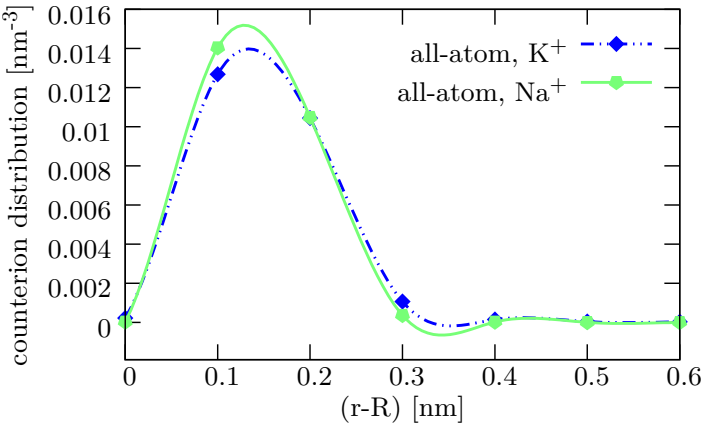
To verify that the scheme works with a flexible polyelectrolyte, we simulated a single chain with 50 monomers, each carrying a charge of $q_m = -1e$. To keep the system neutral we added 50 counterions with charge $q_{ci} = 1e$ but no additional salt ions. We again use a periodic box with side length 24 nm. A comparison of the result for this flexible polyelectrolyte to the schemes applied to the stiff rod, to a simple

Poisson-Boltzmann solution, and to atomistic simulations are presented in Fig. 6.20. The three models that include a salt dependent variation of the dielectric permittivity show quite good agreement. They all differ *qualitatively* from the Poisson-Boltzmann solution with a constant dielectric constant and the simulations with a sharp dielectric jump.

An additional confirmation is that our simulations qualitatively match atomistic simulations, as seen in Fig. 6.20b. The all-atom Molecular Dynamics simulations were performed with the GROMACS 4.5.5 software package [299] at 300 K for a model polyelectrolyte in aqueous solution and in the presence of counterions. The system studied is a sodium chloride (NaCl) solution with a Kirkwood-Buff based force field [264] in combination with the SPC/E water model [52]. Kirkwood-Buff force fields have been shown to reproduce thermodynamic and static properties in good agreement to experimental findings and to avoid the spurious artifacts of other force fields, like the strong occurrence of ion pairs. We constructed a simple fictitious linear polyelectrolyte with 30 'CH2' beads as defined in the GROMOS force field [124] with the corresponding monomeric binding distance of 0.149 nm. The polyelectrolyte can be interpreted as a rod of infinite length by using the periodicity of the simulation box with a cubic side length of 3.804 73 nm in agreement to the approach presented in Ref. [279]. We assigned a charge of $q = \pm 1e$ to each second monomer while the other monomers remain uncharged which gives a line charge density of $l = \pm 3.94\,\mathrm{e/nm}$. The Bjerrum length $\lambda_\mathrm{B} = 0.78\,\mathrm{nm}$ for the SPC/E water model [279] yields a Manning parameter $\zeta = l\lambda_\mathrm{B}/e = 3.08 \gg 1$ which indicates a large fraction of condensed counterions. We randomly inserted $N_c = 15$ counterions in the box to achieve electroneutrality which gives a concentration of $c = 0.45\,\mathrm{M}$. Finally, the positions of the monomers were fixed which avoids the influence of configuration effects on the ion distribution [307]. The largest difference between the coarse-grained and atomistic simulation in Fig. 6.20 is that the counterion density decays much faster in the case of the atomistic simulations. This is because the density of counterions in the box is much higher, which effectively decreases the Debye length $\lambda_\mathrm{D}$ [187].




**(a)** comparison of the three methods




**(b)** atomistic simulations

**Figure 6.20.:** Comparison of the counterion distribution functions for a flexible polyelectrolyte (red triangles), a stiff rod using the iterative (dashed line) and adaptive (dotted line) schemes, and the Poisson-Boltzmann solution that does not take smoothly varying permittivity into account (solid line). The depletion of counterions close to the backbone is present in atomistic simulations as well.

### Radius of Gyration and Diffusion of an Isolated Polyelectrolyte

Despite the relatively large difference in the counterion distribution around the chain when variations in the local dielectric constant are taken into account, most static properties of the polyelectrolyte, like the radii of gyration or end-to-end distance, were surprisingly unaffected by including the local drop of the dielectric constant in the vicinity of the polyelectrolyte. This is clearly seen in the radius of gyration values plotted in Fig. 6.21. Note that here we scaled the box length with the cube root of the chain length $L = 27N^{1/3}$, in order to keep the counterion density constant, and thus the Debye length. Both with and without taking into account variation in the local permittivity constant, the radius of gyration scales as $N^{0.8}$, in good agreement with previous work [212, 227, 304].

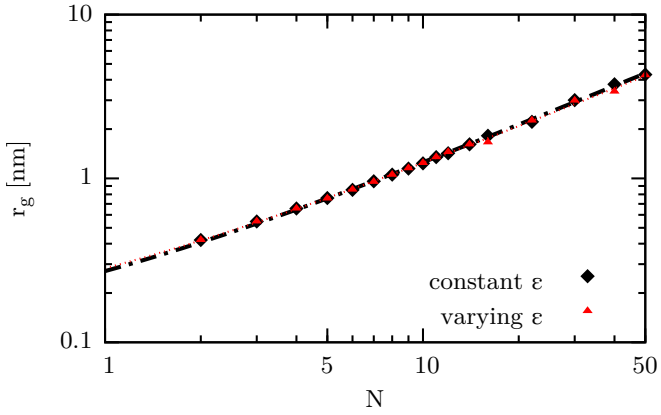



**Figure 6.21.:** The radius of gyration as a function of the chain length with a uniform permittivity (black diamonds) and with varying permittivity (red triangles). We have included exponential fit functions that give a scaling behavior of 0.79 and 0.78 for simulations with and without varying permittivity respectively.

The diffusion of the polyelectrolyte, as shown in Fig. 6.22, leads to slightly different parameters when fitting the power law

$$D = D_0 x^{-m} \tag{6.28}$$

to the data, where the fit parameter $D_0$ represents the diffusion coefficient of a single isolated monomer. The exponent decreases from $m = 1.03(\pm 0.02)$ for constant permittivity to $m = 0.93(\pm 0.03)$ for varying permittivity. This is somewhat surprising since based on Zimm's theory we would expect that the diffusion coefficient would scale with the same exponent as the radius of gyration. The fact that the diffusion coefficient drops off slightly faster is mainly due to the rather small size of the periodic box, which creates large finite-size corrections. Fig. 6.22 shows that the scaling exponent 0.8, obtained from the radius of gyration data, can also reasonably be used to fit the data at large values of $N$.

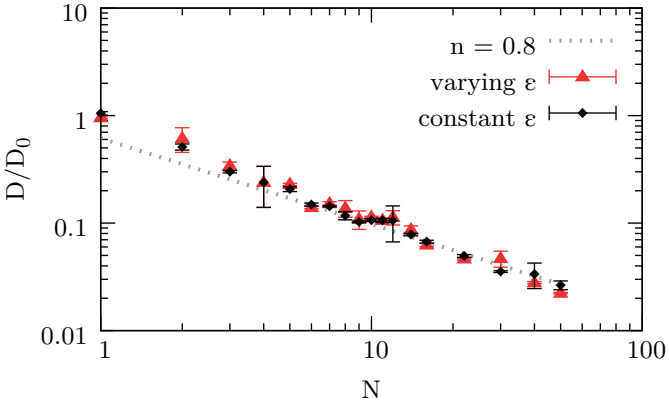



**Figure 6.22.:** Normalized diffusion of a flexible polyelectrolyte with and without varying dielectric permittivity. Both simulations follow the predicted $D = D_0 x^{-m}$ behavior. The scaling parameter changes slightly from $m = 0.93(\pm 0.03)$ for varying to $m = 1.03(\pm 0.02)$ for constant permittivity. But this is mainly due to deviations from the expected $m = 0.8$ behavior at very short polymer lengths, as can be seen by comparing with the dotted line.

### Electrophoresis

In this section, we compare the data of Grass and Holm [213] including hydrodynamic interactions to the same simulation with a Lattice-Boltzmann algorithm including thermalized ghost modes and finally to

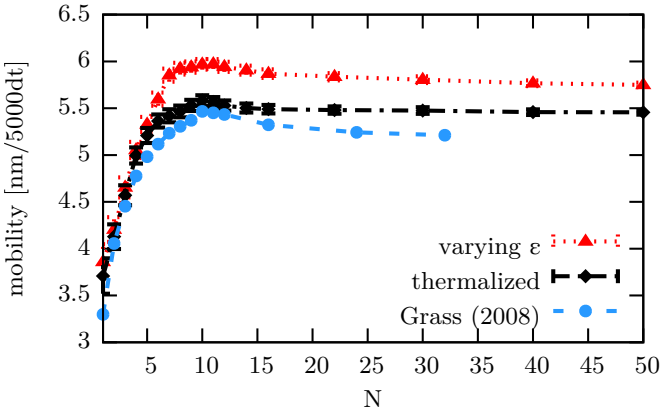



**Figure 6.23.:** Normalized electrophoretic mobility of the polyelectrolyte for constant and varying dielectric background. While both data sets show a similar qualitative behavior, the mobility is significantly increased when variations in the local permittivity are taken into account. For comparison and to judge the influence of ghost mode thermalization, we have included the results obtained by Grass et al. [213].

a new simulation featuring locally and temporally varying permittivity, as calculated by our adaptive approach. Our simulations were carried out using a single polymer of lengths $N$, with one charge $q_m = -1$ per monomer. $N$ counterions with a charge $q_{ci} = 1e$ were also added to the system but no additional salt. The box size was set to $L = 27N^{1/3}$ to maintain a constant counterion density (which also keeps the effective Debye length constant [187]) for all simulations. We measured the velocity of the polymers in the direction of the field and divided by the magnitude of the external field to get the mobility values in Fig. 6.23. The mobility is normalized using $\mu_{\text{red}} = \mu/\mu_0$ with $\mu_0 = e/6\pi\eta l_B$. The inclusion of spatially and temporally varying permittivity shows the same qualitative behavior, but produces mobilities which are significantly larger than the simulations with a uniform dielectric constant.

The mobility values can be interpreted as the ratio of an effective

charge to that of an effective friction coefficient [213, 227, 280, 287]

$$\mu = \frac{Q_{\text{eff}}}{\Gamma_{\text{eff}}}, \tag{6.29}$$

where $Q_{\text{eff}}$ and $\Gamma_{\text{eff}}$ are the effective charge and effective friction of the chain respectively. We can calculate the effective charge per monomer of the polyelectrolyte three different ways: (i) Dynamically, by measuring the mobility in Langevin dynamics in response to an externally applied electric field. (ii) Statically, by determining the integrated charge shift of the moving inflection point according to the ion distribution. And (iii) we can calculate a theoretical prediction with the assumption of Manning condensation,

According to Manning's theory, the predicted value for condensed counterions for a Poisson-Boltzmann distribution is

$$N_{\text{cci}} = 1 - \frac{\sigma_{\text{bond}} l_B}{\approx} 1 - \frac{0.243\,\text{nm}}{0.714\,\text{nm}} \approx 1 - 0.34, \tag{6.30}$$

where $\sigma_{\text{LJ,bond}}$ is the average bond length, and $l_B$ is the Bjerrum length. Manning's theory is known to provide good predictions of the effective charge in the limit of long chains, but fails for very short chains [227, 228, 280].

Regardless of the chain length one can calculate the (static) effective charge directly using the method developed by Belloni [101] and Deserno [112]. We plot the integrated charge density as a function of the logarithm of the distance to the backbone in Fig. 6.24. The counterions closer to the backbone than the inflection point of this graph are considered condensed [112, 213].

In Fig. 6.24 we see that including variations in the local permittivity causes the inflection point to move (green arrow) to longer distances $r$ from the backbone. At the same time, the number of condensed counterions $N_{\text{cci}}$ decreases, increasing the effective charge $Q_{\text{eff}} = (N - N_{\text{cci}})$, resulting in an increase electrophoretic mobility of the polyelectrolyte.

In the case of Langevin dynamics simulations, one can calculate the effective friction $\Gamma_{\text{eff}}$ directly, resulting in:

$$\mu = \frac{Q_{\text{eff}}}{\Gamma_{\text{eff}}} \approx \frac{Q_{\text{eff}}}{N\Gamma}, \tag{6.31}$$

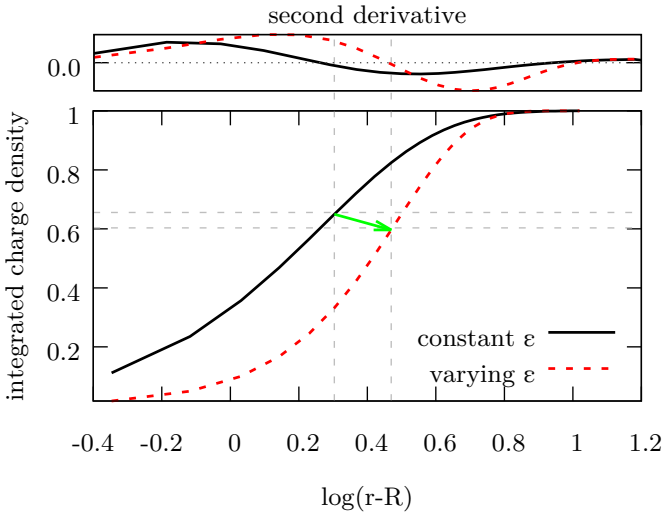
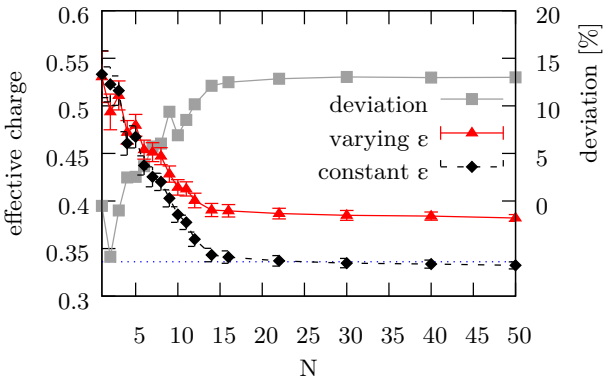


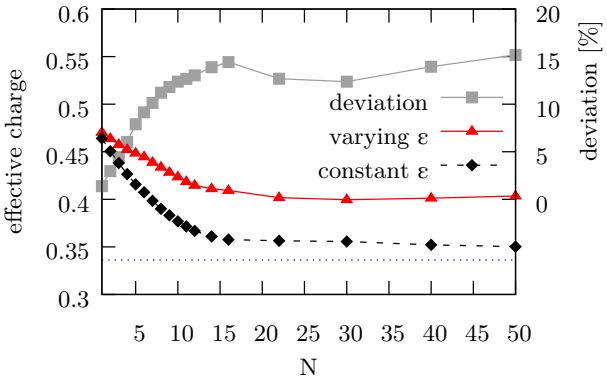

**Figure 6.24.:** Normalized integrated charge density around the polyelectrolyte, plotted against the logarithm of the distance. The inflection point of the graph, pointed out by gray dashed lines, can be determined via a second derivative (upper plot, the dotted line marks zero) and is seen as the boundary between condensed and free counterions. For varying permittivity, this inflection point moves (green arrow) further away from the surface and to a lower integrated charge value, suggesting a higher effective charge and mobility of the polyelectrolyte.

where $N$ is the number of monomers, and $\Gamma = 1m_0/\tau$ is the friction coefficient of an individual MD bead. A rescaled version of the mobility, $Q_{\text{eff}}/N = \Gamma\mu$, is plotted in Fig. 6.25a and matches Manning's prediction in the long chain limit.

Both the results for the Langevin simulations and the static effective charge calculations in Fig. 6.25 yield very similar results, with a difference of less than 3% in the free draining limit. For a constant dielectric background, the effective charge for polyelectrolytes longer than $N = 15$ monomers fits well with Manning's prediction [27]. Our simulations with a flexible varying dielectric permittivity show a significant change compared to our simulations with constant permittivity in




**(a)** dynamic effective charge




**(b)** static effective charge

**Figure 6.25.:** Effective charge of the polyelectrolyte calculated in three different ways. The dashed line in both graphs depicts the effective charge predicted by Manning [27]. (a) shows the dynamic effective charge calculated via the measured mobility with an applied electric field and without hydrodynamic interaction (Langevin dynamics). (b) shows the static effective charge calculated from the integrated charge density as sketched in Fig. 6.24. The simulations with (red triangles) and without (black diamonds) varying dielectric permittivity show very little deviation (gray squares, right axis) at shorter polymer lengths and the difference increases until $N \approx 15$.

effective charge, with the difference being roughly 15% for long chains. This explains why we see an approximately 15% increase in the electrophoretic mobility when taking into account variations in the local dielectric constant.

We have shown that the inclusion of varying dielectric permittivity leads to a significant increase in the electrophoretic mobility as well as the number of free counterions, since less counterions are condensed to the polymer backbone. These two effects should combine in an additive way when looking at the conductivity. In our simulations including an external electric field, we measured the conductivity $\sigma$ using the relation:

$$\sigma = \frac{\mathbf{J}}{\mathbf{E}}, \tag{6.32}$$

where $\mathbf{J}$ is the current density measured via the global dipole moment of the system, and $\mathbf{E}$ is the applied external electric field.

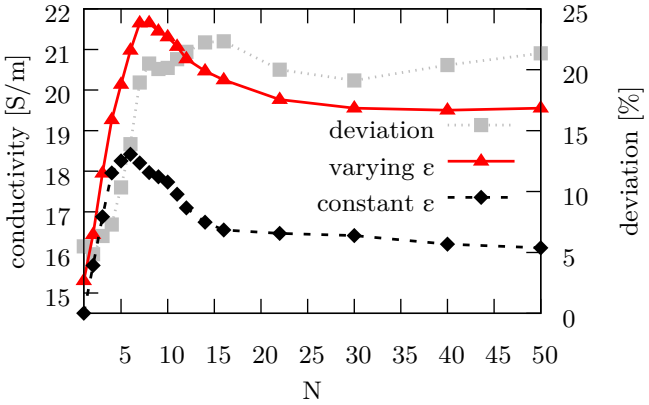



**Figure 6.26.:** The conductivity, calculated via the global dipole moment. The deviation between simulations with and without varying permittivity is even more pronounced, since the higher mobility is accompanied by more uncondensed charges that can contribute to the conductivity.

The conductivities with and without variation in the permittivity

were converted to SI units using the relation

$$\sigma^{\mathrm{SI}} = \sigma^{\mathrm{MD}} \frac{(e^{\mathrm{SI}})^2 (\sigma_{\mathrm{LJ}}^{\mathrm{MD}})^3 \zeta^{\mathrm{SI}}}{(e^{\mathrm{MD}})^2 (\sigma_{\mathrm{LJ}}^{\mathrm{SI}})^3 \zeta^{\mathrm{MD}}}, \tag{6.33}$$

where $\sigma_{\mathrm{LJ}}$ is the length scale, $e$ is the unit charge and $\zeta$ is the friction coefficient of an ion. The superscript denotes the according units system. To calculate the friction coefficient of an ion in SI units we used the Stokes relation $\zeta^{\mathrm{SI}} = 6\pi\eta\sigma^{\mathrm{SI}}$, where $\eta = 8.9 \times 10^{-4}\,\mathrm{Pa\,s}$ is the dynamic viscosity of water. For the friction coefficient $\zeta^{\mathrm{MD}}$ we used the expression from Ahlrichs and Dünweg [106]

$$\zeta^{\mathrm{MD}} = \frac{1}{\Gamma} = \frac{1}{\Gamma_0} + \frac{1}{25\eta a}, \tag{6.34}$$

where $\Gamma_0$ (the bare friction), $\eta$ (the dynamic viscosity), and $a$ (the grid spacing) are the lattice-Boltzmann parameters. The results are shown in Fig. 6.26 and are very comparable for short polymer lengths, but for $N > 12$ taking into account variations in the local dielectric constant results in a 20% increase in the conductivity. To relate this to experimentally accessible results, we are interested in the ratio between the measured conductivity $\sigma^M$ and the ideal conductivity $\sigma^{id}$. The ideal conductivity simply assumes that the mobility of all charge carriers is $\mu = q/6\pi\eta R_{\mathrm{H}}$, which essentially ignores the hydrodynamic and electrostatic coupling between charge carriers. This lets us define a correlation coefficient

$$\Delta = 1 - \frac{\sigma^M}{\sigma^{id}}, \tag{6.35}$$

which is 0 when there is neither hydrodynamic nor electrostatic interactions between the charge carriers. The correlation coefficient quantifies the amount of friction and coupling between the polyelectrolyte and the counterions. The ideal conductivity can be determined with the diffusion constants of both species, via the Stokes-Einstein equation

$$\sigma^{id} = \frac{N e^2 D^+}{k_{\mathrm{B}} T} + \frac{N^2 e^2 D^-}{k_{\mathrm{B}} T}, \tag{6.36}$$

where we can measure the diffusion constant for a single particle $D^0$ and determine the diffusion constant of the polyelectrolyte with

$$D^- = \frac{D^0}{N} + \frac{1}{6\pi\eta R_H} \quad . \tag{6.37}$$

The hydrodynamic radius $R_H$ in this equation is defined as

$$\frac{1}{R_H} = \frac{1}{N^2}\left\langle \sum_{i\neq j} \frac{1}{r_{ij}} \right\rangle \quad , \tag{6.38}$$

which we calculated directly from our simulations. The results for the correlation coefficient $\Delta$ are plotted in Fig. 6.27.

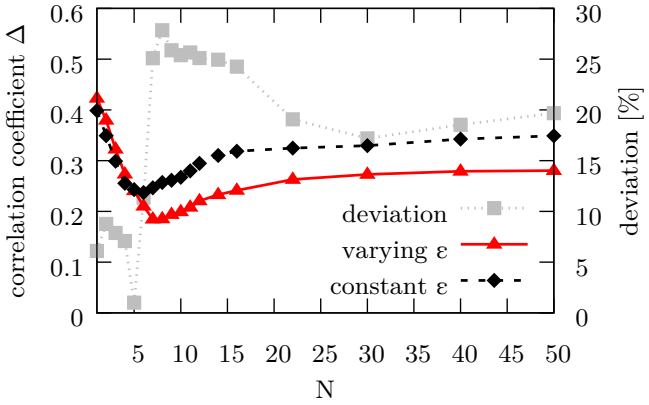



**Figure 6.27.:** The correlation coefficient shows a significant decrease for the simulation with varying dielectric permittivity. This is due to the weaker coupling between the polyelectrolyte and the counterions, and should be observable in experiment.

We observe a significant difference between the approach with and without dielectric variations, exceeding 30% for longer polyelectrolytes. This clearly shows that the coupling between the polyelectrolyte chain and the surrounding counterions is significantly reduced where variations in the permittivity are taken into account due to the structural difference in the EDL. This setup should be more readily accessible

experimentally, and allow for a quantitative comparison to our simulations.

### Conductivity for varying monomer concentration

We now turn to the conductivity of polyelectrolytes in salt-free aqueous solution. In Fig. 6.26, the conductivity plateaus at polymer lengths above 20 monomers. With the same simulation setup as before, we measured the conductivity for different monomer concentrations by keeping the box size constant at $(32\,\mathrm{nm})^3$ and adding polyelectrolyte chains and counterions. We calculated a mean value and standard deviation for the conductivity with lengths $N = 30$, 45, and 60 to make sure that the results are independent of $N$. Renormalizing this data with the monomer concentration gives an equivalent conductivity per monomer $\Lambda$. Through this rescaling $\Lambda$ is independent of both the friction coefficient of the MD beads and the viscosity of the fluid.

Let us first consider the case of constant background permittivity, the blue line in Fig. 6.28. Note that we set the background permittivity using Eq. 6.27, although the results are almost identical to simply using a constant permittivity of $\varepsilon = 78.5$ for all monomer concentrations [6]. The equivalent conductivity drops dramatically until a monomer concentration $C \approx 0.01\,\mathrm{M}$. In this regime, the Debye layer shrinks leading to more and more counterions condensing on the polyelectrolyte backbone as seen in Fig. 6.29. At higher monomer concentrations $C > 0.01\,\mathrm{M}$, both the equivalent conductivity and fraction of condensed counterions continue to increase, but at a much slower rate.

Simulations including variations in the local permittivity also display an initial decrease of the conductivity with increasing polyelectrolyte concentrations, which is again due to an increase in the fraction of condensed counterions in Fig. 6.29. There is a clear minimum around $C = 0.01\,\mathrm{M}$, after which the equivalent conductivity begins to rise. This can be attributed to the decrease in the fraction of condensed counterions in Fig. 6.29. The decrease in the fraction of condensed counterions can in turn be related to the large decrease in the local relative permittivity from around 55 in the dilute limit, to approximately 41 at the highest monomer concentrations (the red line in Fig. 6.29).

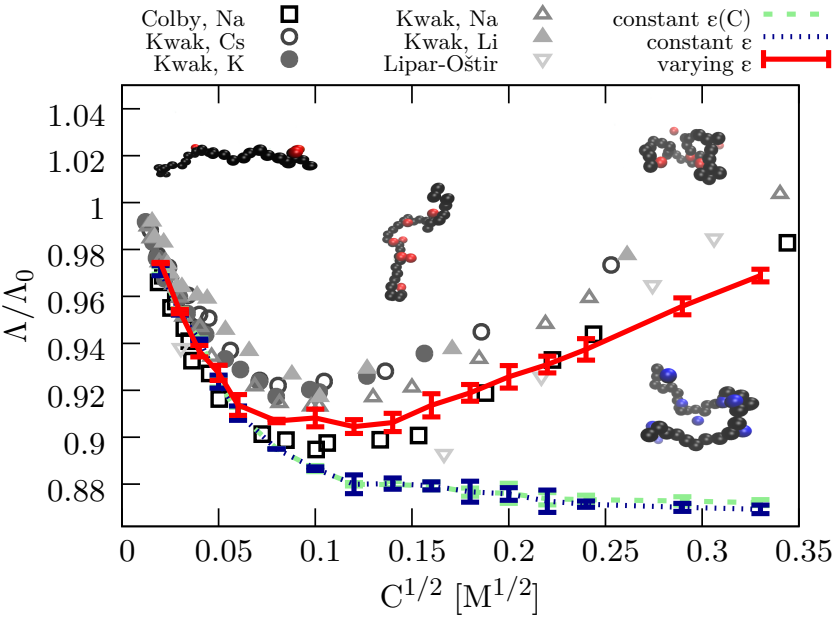



**Figure 6.28.:** The equivalent conductivities $\Lambda/\Lambda_0$ as a function of the square root of molar monomer concentration $C$. The experimentally observed minimum (gray symbols) is reproduced in simulations with varying $\varepsilon(\mathbf{r})$ (red line), while simulations with constant but scaled $\varepsilon(C)$ (blue line) exhibit a different qualitative behavior.

This must be the result of an increase in the local ion concentration, since it alone determines the permittivity in our simulations, see equation 6.27. The reason for this is that the decreasing Debye length causes the polymers to coil significantly as can be seen in figures 6.30 and 6.31.

We also compare our results in Fig. 6.26 to existing experimental data from Kwak and Hayes [31], Colby et al. [93], and Lipar-Oštir et al. [231]. Since the raw results greatly depend on the hydrodynamic radius of the solvated counterions and differ by more than a factor of 2, we normalize all the data to an extrapolated value of $\Lambda(C = 0) = 1$. The experimental data show excellent agreement with

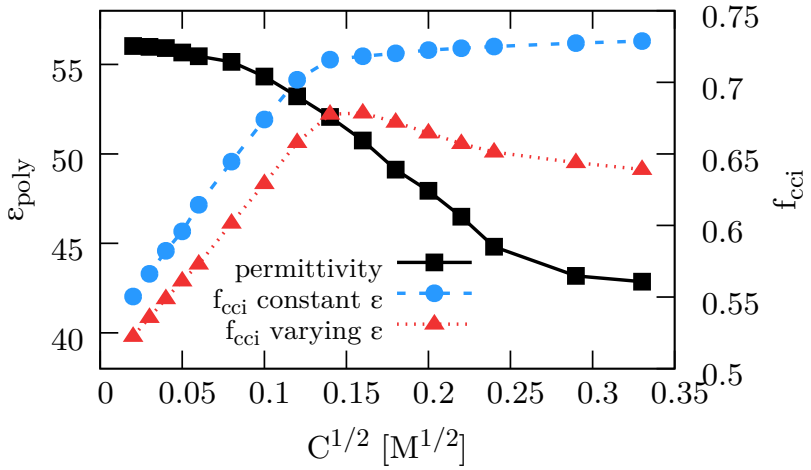
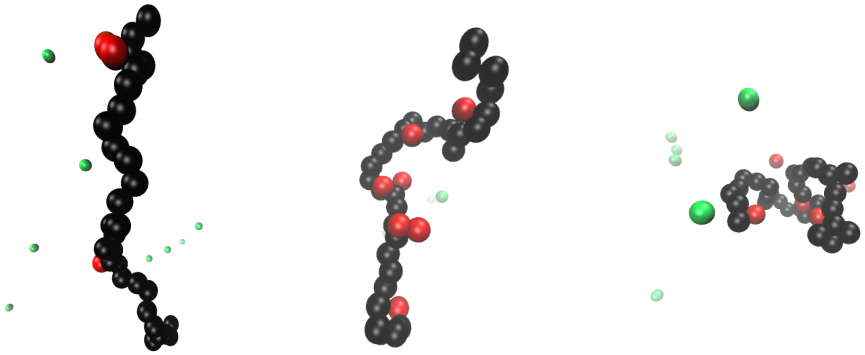



**Figure 6.29.:** Average permittivity in the vicinity of the polyelectrolytes $\varepsilon_{\text{poly}}$ as calculated by the adaptive scheme (black squares, left axis), and the fraction of condensed counterions $f_{\text{cci}}$ (right axis) for constant but scaled $\varepsilon(C)$ (blue circles) and varying $\varepsilon(\mathbf{r})$ (red triangles). $f_{\text{cci}}$ has a maximum for variable permittivity around the conductivity minimum observed before. The salt dependent permittivity keeps decreasing even though the number of condensed counterions almost stagnates at higher concentrations, which is due to the coiling of the polyelectrolyte.

our simulation results. The slight differences are most likely due to enthalpic factors between the counterions and the polyelectrolytes or the specific structure of the polyelectrolyte, however, these are clearly secondary factors.

### 6.4.7. Conclusions

We used our novel implementation of a local electrostatic method to include spatial and temporal changes in the local dielectric permittivity. We then performed simulations of polyelectrolytes solutions. We first looked at a fixed charged straight polymer, and found excellent agreement of the counterion distribution with an additional iterative



**(a)** $C = 0.001\,\mathrm{M}$ **(b)** $C = 0.01\,\mathrm{M}$ **(c)** $C = 0.1\,\mathrm{M}$

**Figure 6.30.:** Snapshots from our simulations. (a) At low monomer concentration, the polyelectrolyte has few condensed counterions (red) and is stretched out because of electrostatic repulsion of the backbone. (b) At higher monomer concentration, the number of condensed counterions is significantly increased. (c) At high monomer concentration, the polyelectrolyte coils. In the simulations with varying dielectric permittivity depending on the local charge concentration, this coiling of highly charged monomers leads to a lower average permittivity around the polymer backbone.

method. Overall, the results were quite similar to findings for the counterion distribution around colloidal particles [5, 305]. We also showed that the inclusion of a dielectric jump in the dielectric permittivity at the surface of the polyelectrolyte was insufficient to produce even qualitatively the same counterion distribution. This is because the dielectric jump fails to take into account the force caused by the gradient in the dielectric permittivity related to the solvation energy of the ions.

We then applied our algorithm to a free, flexible polyelectrolyte. Interestingly, the distribution of counterions around the backbone of the fluctuating polymer are almost identical to the results for the fixed, straight polymer. Despite a qualitatively different structure of the electric double layer around the charged polymer, we observed almost no change in the radius of gyration of the chains when taking into account salt-dependent variations in the local dielectric constant. This demonstrates that the change in the local counterion distribution only has a

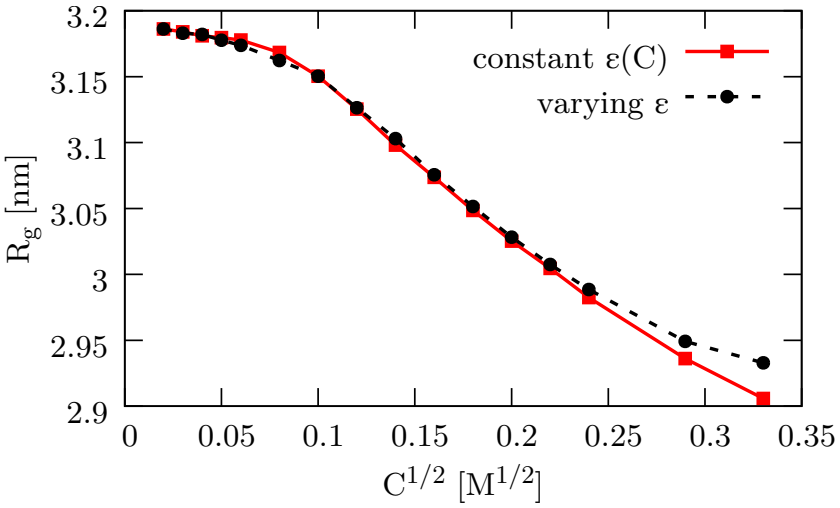



**Figure 6.31.:** Average radius of gyration $R_G$ of the polyelectrolytes as a function of the polyelectrolyte concentration for a uniform background permittivity $\varepsilon(C)$ (red squares, solid line) and a locally varying permittivity $\varepsilon(\mathbf{r})$ (black circles, dashed line).

minor influence on the conformational properties of the polyelectrolyte.

Next, we looked at the role of varying permittivity on the mobility of a polyelectrolyte in aqueous solution subject to an external electric field. We found that the qualitative behavior of earlier simulations [213, 227] is reproduced, in agreement with experimental data [108, 228]. That being said, the difference in the distribution of counterions within the electric double layer had a significant quantitative influence on the electrophoretic mobility. The mobility and effective charge of the polyelectrolytes, in comparison to a simulation without varying permittivity, shows an average increase of around 9% for long polyelectrolytes. We observed a roughly 20% increase in the electric conductivity when taking into account changes in the local permittivity, since both the increased effective charge of the polyelectrolyte and the increased number of uncondensed counterions increase by approximately 10%.

We then investigated the role of monomer concentration on the con-

ductivity of salt-free polyelectrolyte solutions. There was a clear qualitative difference between the simulations with constant dielectric background and the ones using our adaptive scheme for variations in the permittivity, where the latter does not show a continuous decrease of conductivity with rising monomer concentration, but exhibits a distinct minimum and subsequent rise in the equivalent conductivity. The same nonmonotonic behavior has previously been observed in experiments [31, 93, 231], and our simulation results closely match experiment and even exhibit the conductivity minimum at the same monomer concentration.

Our results can be succinctly summarized by saying that the varying dielectric permittivity causes a thickening of the Debye layer, which has only a minor influence on the conformation of the chain. Despite not making dramatic changes to the static properties of the system, the electrophoretic mobility and electric conductivity is significantly larger. Experimental data for the equivalent conductivity at increasing monomer concentration is reproduced using our scheme for varying permittivity, while the simulations assuming a constant dielectric background show qualitatively different behavior. This indicates that correctly accounting for variations in the local permittivity due to the local salt concentration is necessary to achieve quantitative and qualitative agreement with experiment. Most likely taking into account local variations in the dielectric permittivity would lead to similar quantitative differences for other systems such as the electrophoretic mobility of colloids, and electroosmotic flow in microfluidics.

# 7. Conclusion and outlook

In this chapter, I would like to summarize the achievements and results presented in this thesis. I also present some thoughts on possible future work and opinions on my work, as well as express my gratitude towards some of the many people who have helped me during the past years. Since the latter two sections represent my personal opinion and were not discussed with my supervisor or peer reviewed in any way, I will phrase them in first person.

## 7.1. Achievements

This thesis focuses mainly on Molecular Dynamics simulations of biophysical systems. A challenge for such simulations is the inclusion of charges, since their interaction is long-range in nature and thus computationally demanding and hard to parallelize. Many algorithms have been presented to tackle these challenges over the years, and yet long-range interactions still require a large percentage of a simulation's computing power. At the same time, interesting biophysical systems require a very detailed and comprehensive representation of the underlying physical characteristics.

Recently, a new algorithm for long-range interactions has been presented by A. Maggs, and has been adapted for Molecular Dynamics simulations by I. Pasichnyk and J. Rottler in parallel. The algorithm is complex to understand and implement, which is why is has not been implemented very widely. However, it features an advantage over most other electrostatic algorithms: It is intrinsically local. This not only means that parallelization is straight forward, but that the algorithm can theoretically represent local changes in the electrostatic properties of the system, and thereby potentially introduce a more detailed representation of biophysical systems.

In chapter 3, we have extended the existing MEMD algorithm to include spatially and temporally varying dielectric permittivity $\varepsilon(\mathbf{r}, t)$, and we have proven that it remains physically valid and reproduces correct observables in the thermodynamic limit. We have introduced a novel method for the initial solution and the self energy correction, both of which are adapted to work with locally varying dielectric properties. In addition, we have presented and confirmed an error estimate for the algorithm as well as ways of reducing the error. Finally, we have presented a correction for the dipole moment, or "boundary conditions" at infinity to perfectly match the results of the algorithm to existing electrostatics methods. We have discretized the algorithm in space and time, and implemented and parallelized it within the software package ESPResSo and the ScaFaCoS library for Scalable Fast Coulomb Solvers.

We have verified the results of this new algorithm and implementation with various artificial and biophysical setups, some of which were presented in chapter 4. For the verification, two different parts of the algorithm had to be tested, the initial solver as well as the dynamic propagation. Since the extension and implementation mostly focuses on the variation in dielectric properties, we chose systems with sharp as well as smoothly varying permittivity for our comparisons. We also compared the initial solution and the dynamic propagation to analytical results as well as simulations using other available algorithms for dielectric enclosures. Our new algorithm has shown reliable agreement with theoretical predictions and in comparison to other available methods. After a long series of tests and verifications, we can confidently say that the MEMD algorithm with varying dielectric properties gives correct results and reproduces the desired behavior to the given accuracy.

We have also taken a look at some restrictions the algorithm has in comparison to other methods. Three major problems are its speed, especially at systems that are not well suited for its structure. Second, it struggles with setups that feature a systematically increasing dipole moment, like external electric fields. Although we have implemented a correction for this behavior, the algorithm performs very badly in such setups. And third, the algorithm is not suitable for simulations

where high accuracy for electrostatic interactions is required. The way we have implemented the algorithm, it will given correct results up to an accuracy of $10^{-4}$ in the RMS force error, and it can not be tuned to a higher precision.

Our implementation was very well suited for a direct comparison with other, more established electrostatic methods. It is efficiently programmed and parallelized, but most importantly, it was implemented in ESPResSo and ScaFaCoS, two software packages that include a variety of different electrostatics algorithms. Within ESPResSo, we had access to several methods that allow for spatially varying dielectric properties and the ability to compare MEMD results and performance to these methods. On the other hand, ScaFaCoS featured a more exhaustive list of possible solvers for the Poisson equation, all of which were optimized and highly parallelized, making it ideal for efficiency and scalability comparisons. We have shown that MEMD performs quite competitively for such a young method and that it scales very well for highly parallel applications.

After having proven, implemented, validated, and compared our extension of the MEMD algorithm, we applied it to several biophysical systems that are of scientific interest and that feature variations in the dielectric constant that were ignored in previous research with Molecular Dynamics simulations. In ascending complexity, we started with slit pore systems including sharp dielectric contrasts that were accessible to other electrostatic algorithms. We moved on to colloidal systems with smooth changes in the dielectric constant, for which our colleague Zhenli Xu specifically developed a Monte Carlo algorithm to simulate the exact same setup. After these two simulations showed promising results and further grounded our confidence in the algorithm's integrity, we finally simulated a fully flexible polyelectrolyte in solution, with spatially and temporally arbitrarily varying dielectric permittivity dependent on the local charge concentration.

We researched two slit pore systems. The first one was on the electrophoretic behavior of a polyelectrolyte confined by two infinite charged dielectric glass walls at a very short distance of 10 nm, with an external electric field. For the hydrodynamic interactions, large soft particles and a DPD thermostat were applied, and the electrostatic in-

teractions were calculated with MEMD, and for testing purposes with ICMMM2D. We found that the inclusion of dielectric contrasts in the simulation pushes counterions away from the charged channel walls, creating a wider double layer. Since the ions within the double layer move along the external field and the polyelectrolyte moves in the opposite direction, the latter is pushed towards the center. This behavior is more pronounced if dielectric boundaries are included in the simulation and the double layer is widened, which effectively creates a larger friction between the polyelectrolyte and the walls. This is interesting, since this friction is desired in experimental setups to determine the length of polymers within the channel. While our findings in our eyes did not qualify for publication, we still learned that dielectric interfaces can have a measurable impact at short length scales, and that the MEMD algorithm is capable of successfully handling real world simulations, although this particular example might not be perfectly suited.

The second slit pore system had been experimentally investigated by Madhavi Krishnan et al. in 2007 and to this day is not fully understood from a theoretical or simulational viewpoint. They found that negatively charged polyelectrolytes if put in a very narrow nanochannel with negatively charged walls, can spontaneously slip into the corner of the channel and stretch out, and they will be stuck there. This is counterintuitive, since the walls and the polyelectrolyte contain same-sign charges and repel one another, and the polyelectrolyte will additionally have a greatly reduced entropy when stuck to the corner. However, if the electrostatic interactions are strong enough, the asymmetric geometry with a large width and small height of the nanochannel can lead to this effect. This has been studied analytically by Dmitriy Rozhkov and via simulations of static systems by Alexander Schlaich using an Iterative Poisson-Boltzmann Solver (iPBS), but the results remained inconclusive. We repeated the simulations with iPBS and included the correct scaling of the salt concentration, and therefore the Bjerrum length, in the system. In addition, we studied the system with dynamic simulations in ESPResSo with the MEMD algorithm. We were able to explain the behavior observed in experiment, since very close to the corner, we found a strong attractive force on the polyelectrolyte, since

the electrostatic potential dropped rapidly. If one end of the polyelectrolyte randomly moves into this potential well, it will be trapped and slowly pull in the rest of the chain. While these results are exciting, we only saw this behavior once in our parameter sweep, at very narrow channels and for very strong surface charges on the wall. The findings were not consistent enough to qualify for scientific publication.

A colloidal particle in aqueous solution is an object with a very high surface charge density, surrounded by counterions and additional salt ions. It has been shown via atomistic simulations by Stephan Gekle and other groups, that water molecules in the close vicinity of such objects will orient their dipole moments and have a restricted rotational degree of freedom, leading to a lowered effective dielectric permittivity. Our colleague Zhenli Xu developed the HIM algorithm, which is capable of simulating almost arbitrary functions for the dielectric permittivity in radially symmetric systems, which is perfectly suited for a colloid. As a first approximation, we assumed that a more realistic representation of the dielectric bulk permittivity around the colloid is a linear rise from a dielectric constant of 2 inside the colloid to 80 in the bulk. We simulated this system with a Monte Carlo code using the HIM algorithm as well as with ESPResSo and MEMD. We found that the distribution of counterions in close vicinity of the colloid is strongly influenced by this varying dielectric background. The main contribution to this trend is that the gradient in dielectric permittivity exerts a significant force on particles, driving them towards the direction of higher polarizability. This effectively widens the surrounding double layer. Although this is a phenomenon at short distances from the colloid surface, its influence can be seen in the far field as well. We have shown that the effective colloid radius, and the effective charge of the colloid increases by up to 17%. It is also important to mention that the results for MEMD and HIM agree almost perfectly, giving us another validation of our new algorithm.

The final setup we examined were polyelectrolytes in aqueous solution. The goal was to simulate the system with a spatially and temporally varying dielectric permittivity, directly linked to the local charge concentration. Since this has never been attempted before and we did not have any algorithm to confirm our simulations, we approached the

problem step by step. We first simulated a stiff rod, starting with a permittivity function that matched the numerical Poisson-Boltzmann distribution for counterions. We iteratively adjusted the permittivity function to newly gained ion distributions from the last simulation and found that it converges towards a counterion distribution similar to the one we found at colloid surfaces. We then applied a newly developed adaptive scheme that instantly measures the local charge concentration surrounding each lattice cell and sets the permittivity accordingly. The results of this scheme agreed very well with the final iteration we calculated before. Finally, we simulated a flexible polyelectrolyte using the same adaptive scheme and still saw a comparable counterion distribution.

Looking at several other static observables of the system, we quickly found that the inclusion of dielectric variations did not have a big impact on many of them, for example the radius of gyration of the polymer or the diffusion coefficient. We thus opted to look at dynamic properties of the system, most importantly with the application of an external electric field. Since this will exert a force on the polyelectrolyte in one direction and on the counterions in the opposite direction, the difference in counterion distribution can lead to a difference in the correlation between those two and have a major influence on macroscopic observables. We found a significant increase in the electrophoretic mobility of the polyelectrolyte and its interconnected effective charge. The distinction is even more pronounced for the electric conductivity of the system, since the greater mobility and the higher amount of free ions in the system are coupled in an additive way, reaching a more than 20% increase of the conductivity with the inclusion of dielectric variations in the system.

At a certain polymer length, the conductivity reaches the free draining limit and can be expressed by a single number. We looked at the change of this free draining conductivity for systems with different monomer concentrations by keeping a fixed box size and adding polyelectrolytes and counterions. The results are surprising and qualitatively differ with the inclusion of varying dielectrics. When looking at the normalized conductivity – scaled by the number of charges in the system –, it first decreases with increasing salt concentration. This

is expected, since higher salt concentrations will lead to shorter Debye lengths and therefore a higher fraction of counterions that are condensed on the polyelectrolyte backbone. This directly influences both the mobility of the polymer and the amount of free salt in the system, leading to the observed effect. But at salt concentrations higher than 10 mM, the conductivity slowly levels off for a system with constant background permittivity, whereas it rises if we include dielectric variations. We showed that this is due to the coiling of the polyelectrolyte, which directly leads to a higher charge concentration and therefore a lower dielectric permittivity in the close vicinity of the polymer backbone. Counterions are pushed further away and are less tightly bound, leading to a rise in conductivity. This latter behavior is what is observed in experiment and to our knowledge, it has not been reproduced in simulations before or explained even qualitatively. This shows that for certain systems, the introduction of dielectric variation can have a significant influence on the quantitative and even qualitative behavior of observables.

## 7.2. Possible future work

I have worked with the MEMD algorithm as well as several alternatives over the past few years. There are some systems that are sensitive to smooth dielectric variations in areas where charges can move freely, and currently this is the most flexible electrostatic algorithm capable of calculating the resulting forces in such a system. Most significantly, systems that include dynamic processes in the close vicinity of highly charged objects can display a key behavioral difference. This is certainly applicable to the electrophoresis of differently structured systems, like nanochannels with grafted polymers, and can be researched using MEMD. However, the algorithm is not well suited for such setups, since they include a very fine system resolution as well as a constantly increasing dipole moment because of electrophoretic motion. The latter will force MEMD to use the initial solver very frequently, and since this solver scales with $\mathcal{O}(N^3)$, $N$ being the number of lattice cells in one dimension, the simulations are computationally taxing. In future

implementations, this could be avoided by either correcting the dipole moment via a different approach such as treating it as an additional degree of freedom in the system. Or by simulating such setups without an explicit external electric field using a Green-Kubo relation to determine the electrophoretic transport coefficients. Although it has proven very difficult to obtain a meaningful signal to noise ratio with this particular setup.

In addition, the MEMD algorithm, while is has certainly come a long way, is complicated to grasp and apply. The artificial dynamics underlying the force calculation are hard to intuitively comprehend and if some properties are obviously wrong, it is time consuming, if not sometimes impossible, to determine the source of the problem. The algorithm is certainly capable of producing the correct behavior if used with care, but especially when dealing with varying dielectric backgrounds, it can easily lead to unphysical behavior. This is why, in every single setup of this thesis, I took very small steps towards the final goal, always comparing against known results or alternative methods, since I am very hesitant to blindly trust the results of this algorithm entirely. Imagine an example where you have quick and harsh dielectric changes in one lattice cube of the system. These changes will dampen the electric field within this cell very quickly, and the lost energy might not be recovered by the resulting electromagnetic waves created from the change. A similar problem arises if sharp dielectric contrasts move through the system. An analogy in the Lattice-Boltzmann algorithm would be an object made out of hydrodynamic walls moving through the system. Two questions arise immediately: What is the density and the velocities at the newly freed cells *behind* the object, and what happens to the populations and kinetic properties of the cells being "eaten up" *in front* of the object? This is the reason why when I simulated the adaptive scheme of the flexible polyelectrolyte, I used $7^3$ cells with a weighted sum to calculate the local salt concentration. Apart from the physical meaning that distant particles will have a smaller influence, the jump in dielectric permittivity needed to be small if one or two ions enter or exit the area of averaging, or else I would have expected unphysical behavior.

A similar problem arises when a sharp dielectric interface, for exam-

ple between water and oil, is included and particles are allowed to move through it. While the kinetic energy needed for this should reflect the difference in solvation energy between the two media, the force on any particle approaching such an interface will diverge. While it can be appreciated that sharp changes in the dielectric constant, as well as sharp boundaries and infinitely thin walls are not present in real physical systems, the modeling for such a simulation has to be done very carefully and with these effects in mind. I would recommend, for any systems that do not necessarily have to include smooth variation in the dielectric constant, to use one of the alternative algorithms present in ESPResSo, particularly ICC$\star$ in combination with P3M. There are far less pitfalls and potential problems, and the algorithm does not need to be completely understood to be usable, nor is it as complex as MEMD to understand.

For standard electrostatic force calculations on a CPU, the implementation of P3M in ESPResSo and ScaFaCoS is superior to MEMD. There is, however, a possible use for the algorithm when graphics processing units (GPUs) are considered. Structurally, the implementation of this algorithm is very similar to Lattice-Boltzmann, which has shown an increase in speed by almost two orders of magnitude – compared to a single CPU – when ported to CUDA on the GPU. To allow for high precision calculations, I would recommend a higher order interpolation scheme when implementing the MEMD algorithm for graphic cards, but even with this restriction, it should easily speed up the calculation of long-range interactions by a factor of 10. While this might be possible to reach by implementing algorithms like P3M in CUDA, MEMD will most likely perform quite competitively in this scenario. This is, in my opinion, the prime application that needs to be considered for this algorithm.

To conclude, after having examined several biophysical systems and having worked with the MEMD algorithm for several years, I do not consider it a real alternative to FFT-based algorithms. The inclusion of smooth dielectric backgrounds has put a lot of restrictions on the existing correction schemes, and is the one main advantage of MEMD over more established methods. However, for most biophysical systems and the observables associated with them, such smooth variations either

do not exist, or have only a small or no influence on the quantitative outcome. During my research, I looked at a long list of suitable systems where the inclusion of dielectrics might make a significant difference, some of which were presented in this thesis. The only system that exhibited major changes was the conductivity of polyelectrolytes in aqueous solutions, and even for that system, other properties like the electrophoretic mobility merely included some quantitative shifts, while the qualitative trends remained unchanged. It is very advantageous to know for future simulations that dielectric variations in the system will most likely not have any major impact on the observables and can be neglected. But I am unsure if further research in this direction is justified.

## 7.3. Acknowledgements

The research presented in this thesis would not have been possible without the help and support of other people. First and foremost, I would like to cordially thank my family. My wife Theresa has always supported me and has shown great understanding and patience towards the problems and setbacks I faced during my time in Stuttgart. My two children, Leo and Silva, have given me great joy and accepted when I was not able to be with them as much as I would have liked to. My parents and siblings have supported me and my choices throughout the entire time. I am eternally grateful to have such a kind and loving family and I could not have done it without them.

Research can only blossom in a nurturing environment. I enjoyed working at the Institute for Computational Physics very much and have met colleagues and friends here. I have had fruitful and friendly discussions with many of my coworkers, and would like to name a few that were most helpful during my stay. Stefan Kesselheim started his PhD at the same time, and he has accompanied me and my research for the entire way. He was the person who was involved most closely with the development of the algorithmic details and was a source of helpful feedback whenever I turned to him. Olaf Lenz did not only support and motivate me during the writing of this thesis, but he was of great help

during the implementation of the MEMD algorithm for the ScaFaCoS library, for which I am very grateful. In the simulational part of my research, Owen Hickey has always been a source of great feedback and inspirational ideas. He has helped me come up with interesting physical systems and discussed the results with me at length. I consider all three of them newly gained and dear friends and will certainly stay in contact with them.

Finally, I would like to thank Christian Holm, my supervisor. Not only has he provided the kickoff idea for the entire project, but he has guided me throughout the entire time and was always available for questions and discussions. He was supportive when I founded my new family during my stay at the institute and took my parental leave, and he allowed me to follow the research path I deemed the most interesting. Last but not least, he provided the aforementioned welcoming and friendly atmosphere at the Institute for Computational Physics, which in my experience is a unique and very fruitful work environment.

# 8. Bibliography


8. Neumann, C. in, 279–282 (Teubner, Leipzig, 1883).
9. Gouy, G. L. *J. de Phys.* **9,** 457 (1910).
10. Chapman, D. L. *Philosophical Magazine* **25,** 475 (1913).
11. Born, M. Volumes and heats of hydration of ions. *Z. Phys.* **1,** 45–48 (1920).
12. Ewald, P. P. Die Berechnung optischer und elektrostatischer Gitterpotentiale. *Annales de Physique* **369,** 253–287 (1921).
13. Debye, P. & Hückel, E. Zur Theorie der Elektrolyte. I. Gefrierpunktserniedrigung und verwandte Erscheinungen. *Phys. Z.* **24,** 185 (1923).
14. Lennard-Jones, J. E. On the Determination of Molecular Fields. II. From the Equation of State of a Gas. *Royal Society of London Proceedings Series A* **106,** 463–477 (Oct. 1924).
15. Derjaguin, B. V. & Landau, L. D. Theory of the stability of strongly charged lyophobic sols and of the adhesion of strongly charged particles in solutions of electrolytes. *Acta Physicochim. (USSR)* **14,** 633–650 (1941).
16. McMillan, W. G. & Mayer, J. E. The Statistical Thermodynamics of Multicomponent Systems. *J. Chem. Phys.* **13,** 276 (1945).
17. Hasted, J. B., Ritson, D. M. & Collie, C. H. Dielectric Properties of Aqueous Ionic Solutions. Parts I and II. *The Journal of Chemical Physics* **16** (1948).
18. Verwey, E. J. & Overbeek, J. T. G. *Theory of the stability of Lyophobic Colloids* (Elsevier, Amsterdam, 1948).

19. Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. & Teller, E. Equation of State Calculations by Fast Computing Machines. *J. Chem. Phys.* **21.** The original paper of the Metropolis algorithm., 1087 (1953).

20. Bhatnagar, P. L., Gross, E. P. & Krook, M. A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems. *Physical Review* **94,** 511 (1954).

21. Nijboer, B. & Wette, F. de. On the calculation of lattice sums. *Physica* **23,** 309–321 (1957).

22. Allen, L. C. & Karo, A. M. Basis functions for ab initio calculations. *Revs. Modern Phys.* **32** (1960).

23. Hohenberg, P. & Kohn, W. Inhomogeneous electron gas. *Physical Review* **136,** 864–871 (1964).

24. Kohn, W. & Sham, L. J. Self-Consistent Equations Including Exchange and Correlation Effects. *Physical Review* **140,** A1133– (Nov. 1965).

25. Moore, G. E. *et al.* Cramming more components onto integrated circuits. *Electronics,* 114–117 (1965).

26. Wiersema, P. H., Loeb, A. L. & Overbeek, J. T. Calculation of Electrophoretic Mobility of a spherical Colloid Particle. *Journal of Colloid and Interface Science* **22,** 78 (1966).

27. Manning, G. Limiting Laws and Counterion Condensation in Polyelectrolyte Solutions I. Colligative Properties. *Journal of Chemical Physics* **51,** 924–933 (1969).

28. Weeks, J. D., Chandler, D. & Andersen, H. C. Role of Repulsive Forces in Determining the Equilibrium Structure of Simple Liquids. *Journal of Chemical Physics* **54,** 5237 (1971).

29. Schmitt, A. & Varoqui, R. Electrical and mass transport in salt-free polyelectrolyte solutions. *Journal of the Chemical Society, Faraday Transactions 2* **69,** 1087–1103 (1973).

30. Schoenberg, I. J. *Cardinal Spline Interpolation* (Society for Industrial and Applied Mathematics, Philadelphia, 1973).

31. Kwak, J. C. T. & Hayes, R. C. Electrical conductivity of aqueous solutions of salts of polystyrenesulfonic acid with univalent and divalent counterions. *The Journal of Physical Chemistry* **79,** 265–269 (1975).

32. Ermak, D. L. & McCammon, J. Brownian dynamics with hydrodynamic interactions. *The Journal of chemical physics* **69,** 1352 (1978).

33. Ladd, A. J. C. Long-range dipolar interactions in computer simulations of polar liquids. *Mol. Phys.* **36,** 463–474 (1978).

34. O'Brien, R. W. & White, L. R. Electrophoretic Mobility of a spherical colloidal Particle. *Journal of the Chemical Society, Faraday Transactions 2* **74,** 1607 (1978).

35. Drift, W. P. J. T. V. D., Keizer, A. D. & Overbeek, J. T. G. Electrophoretic mobility of a cylinder with high surface charge density. *Journal of Colloid and Interface Science* **71,** 67–78 (1979).

36. Leeuw, S. W. de, Perram, J. W. & Smith, E. R. Simulation of Electrostatic Systems in Periodic Boundary Conditions. I. Lattice Sums and Dielectric Constants. *Proceedings of the Royal Society of London. Series A: Mathematics and Physical Sciences* **373,** 27–56 (Oct. 1980).

37. Leeuw, S. W. de, Perram, J. W. & Smith, E. R. Simulation of Electrostatic Systems in Periodic Boundary Conditions. II. Equivalence of Boundary Conditions. *P. Roy. Soc. Lond. A: Mat.* **373,** 57–66 (1980).

38. Berendsen, H. J. C., Postma, J. P. M., Gunsteren, W. F. & Hermans, J. in *Intermolecular Forces* (ed Pullman, B.) 331–342 (Springer Netherlands, 1981).

39. Smith, E. R. Electrostatic energy in ionic crystals. *P. Roy. Soc. Lond. A: Mat.* **375,** 475–505 (1981).

40. Swope, W. C., Andersen, H. C., Berens, P. H. & Wilson, K. R. A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters. *Journal of Chemical Physics* **76,** 637–649 (1982).

41. Torrie, G. M., Valleau, J. P. & Patey, G. N. Electrical double layers. II. Monte Carlo and HNC studies of image effects. *J. Chem. Phys.* **76,** 4615–4622 (1982).

42. Jorgensen, W. L., Chandrasekhar, J., Madura, J. D., Impey, R. W. & Klein, M. L. Comparison of simple potential functions for simulating liquid water. *Journal of Chemical Physics* **79,** 926–935 (1983).

43. Neumann, M. Dipole moment fluctuation formulas in computer simulations of polar systems. *Molecular Physics* **50,** 841–858 (1983).

44. Alexander, S., Chaikin, P. M., Grant, P., Morales, G. J., Pincus, P. & Hone, D. Charge Renormalization, Osmotic-Pressure, And Bulk Modulus Of Colloidal Crystals - Theory. *J. Chem. Phys.* **80,** 5776–5781 (1984).

45. Kjellander, R. & Marcelja, S. Correlation and image charge Effects in electric Double-Layers. *Chem. Phys. Lett.* **112,** 49 (1984).

46. Torrie, G. M., Valleau, J. P. & Outhwaite, C. W. Electrical double layers. VI. Image effects for divalent ions. *J. Chem. Phys.* **81,** 6296–6300 (1984).

47. Car, R. & Parrinello, M. Unified Approach For Molecular Dynamics and Density Functional Theory. *Physical Review Letters* **55,** 2471 (1985).

48. Pethig, R. Dielectric and electrical properties of biological materials. *Journal of Bioelectricity* **4,** vii–ix (1985).

49. Barnes, J. & Hut, P. A hierarchical $\mathcal{O}(N \log N)$ force-calculation algorithm. *Nature* **324,** 446–449 (1986).

50. Linse, P. Image charge effects in spherical symmetry with applications to micellar systems. *Journal of Physical Chemistry* **90,** 6821–6828 (1986).

51. Allen, M. P. & Tildesley, D. J. *Computer Simulation of Liquids* 1st ed. (Clarendon Press, Oxford, 1987).

52. Berendsen, H. J. C., Grigera, J. R. & Straatsma, T. P. The missing term in effective pair potentials. *The Journal of Physical Chemistry* **91,** 6269–6271 (Nov. 1987).

53. Greengard, L. & Rokhlin, V. A fast algorithm for particle simulations. *J. Comput. Phys.* **73,** 325–348 (1987).

54. Atkinson, K. A. *Introduction to Numerical Analysis* (1988).

55. Hockney, R. W. & Eastwood, J. W. *Computer simulation using particles* (Taylor & Francis, Inc., Bristol, PA, USA, 1988).

56. McNamara, G. R. & Zanetti, G. Use of the Boltzmann-equation to simulate lattice-gas automata. *Physical Review Letters* **61,** 2332–2335 (1988).

57. Perram, J. W., Petersen, H. G. & Leeuw, S. W. de. An algorithm for the simulation of condensed matter which grows as the 3/2 power of the number of particles. **65,** 875–893 (1988).

58. Arnold, V. I. *Mathematical methods of classical mechanics* (Springer Science & Business Media, 1989).

59. Kaatze, U. Complex permittivity of water as a function of frequency and temperature. *Journal of Chemical and Engineering Data* **34,** 371–374 (1989).

60. Parr, R. G. & Yang, R. G. P. W. *Density-functional theory of atoms and molecules* (Oxford university press, 1989).

61. Russel, W. B., Saville, D. A. & Schowalter, W. R. *Colloidal Dispersions* (Cambridge University Press, Cambridge, UK, 1989).

62. Beest, B. W. H. van, Kramer, G. J. & Santen, R. A. van. Force fields for silicas and aluminophosphates based on ab initio calculations. *Phys. Rev. Lett.* **64,** 1955–1958 (1990).

63. Lekner, J. Summation of Coulomb fields in computer simulated disordered systems. *Physica A* **176,** 485 (1991).

64. Hoogerbrugge, P. J. & Koelman, J. M. V. A. Simulating Microscopic Hydrodynamic Phenomena with Dissipative Particle Dynamics. *Europhysics Letters* **19,** 155–160 (1992).

65. Kolafa, J. & Perram, J. W. Cutoff errors in the Ewald summation formulae for point charge systems. *Molec. Sim.* **9,** 351–368 (1992).

66. Lyapunov, A. M. The general problem of the stability of motion. *International Journal of Control* **55,** 531–534 (1992).

67. Press, W. H., Teukolsky, S. A., Vetterling, W. T. & Flannery, B. P. *Numerical Recipes: The Art of Scientific Computing* 2nd ed. (Cambridge University Press, Cambridge, 1992).

68. Rycaerz, Z. A. & Jacobs, P. W. M. Ewald summation in the molecular dynamics simulation of large ionic systems: the cohesive energy. *Molec. Sim.* **8,** 197–213 (1992).

69. Schrödinger, E. *What is life?: With mind and matter and autobiographical sketches* (Cambridge University Press, 1992).

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

71. Lindell, I. V. in *The Review of Radio Science 1990-1992* (ed Stone, W. R.) 107–126 (Oxford University Press, Oxford, 1993).

72. Sperb, R. *Extension and simple proof of Lekner's summation formula for Coulomb forces* tech. rep. 93-06 (Math. Sem. ETH Zürich, 1993).

73. Warren, M. S. & Salmon, J. K. *A parallel hashed Oct-Tree N-body algorithm* in *Proceedings of the 1993 ACM/IEEE conference on Supercomputing* (ACM, Portland, Oregon, United States, 1993), 12–21.

74. Caillol, J.-M. Comments on the numerical simulation of electrolytes in periodic boundary conditions. *Journal of Chemical Physics* **101,** 6080–6090 (Oct. 1994).

75. Fincham, D. Optimisation of the Ewald sum for large systems. *Mol. Simul.* **13,** 1–9 (1994).

76. Luty, B., Davis, M., Tironi, I. & Gunsteren, W. van. A comparison of particle-particle, particle-mesh and Ewald methods for calculating electrostatic interactions in periodic molecular systems. *Mol. Sim.* **14,** 11–20 (1994).

77. Sperb, R. *An alternative to Ewald sums* tech. rep. 94-01 (Math. Sem. ETH Zürich, 1994).

78. White, C. A. & Head-Gordon, M. Derivation and efficient implementation of the fast multipole method. *J. Chem. Phys.* **101,** 6593–6605 (Oct. 1994).

79. York, D. & Yang, W. The Fast Fourier Poisson method for calculating Ewald sums. *J. Chem. Phys.* **101,** 3298 (1994).

80. Ding, H. Q., Ferraro, R. D. & Gennery, D. B. *A Portable 3D FFT Package for Distributed-Memory Parallel Architectures* in *Proceedings of the 7th SIAM Conference on Parallel Processing* (SIAM, Philadelphia, 1995), 70 –71.

81. Dobrynin, A. V., Colby, R. H. & Rubinstein, M. Scaling theory of polyelctrolyte solutions. *Macromolecules* **28,** 1859–1871 (1995).

82. Español, P. & Warren, P. Statistical mechanics of Dissipative Particle Dynamics. *Europhysics Letters* **30,** 191 (1995).

83. Esselink, K. A comparison of algorithms for long-range interactions. *Comput. Phys. Commun.* **87,** 375–395 (1995).

84. Essmann, U., Perera, L., Berkowitz, M. L., Darden, T., Lee, H. & Pedersen, L. G. A smooth Particle Mesh Ewald method. *J. Chem. Phys.* **103,** 8577 –8593 (1995).

85. Petersen, H. G. Accuracy and efficiency of the particle mesh Ewald method. **103,** 3668–3679 (1995).

86. Plimpton, S. J. Fast Parallel Algorithms for Short-Range Molecular Dynamics. *J. Comput. Phys.* **117,** 1–19 (1995).

87. Pollock, E. L. & Glosli, J. Comments on $P^3M$, FMM, and the Ewald method for large periodic Coulombic systems. **95,** 93–110 (1996).

88. Spotz, W. & Carey, G. A High-Order Compact Formulation for the 3D Poisson Equation. *Numer. Meth. Part. D. E.* **12,** 235–243 (1996).

89. Toukmaji, A. Y. & Jr., J. A. B. Ewald summation techniques in perspective: a survey. **95,** 73–92 (1996).

90. Toukmaji, A. Y. & Board Jr., J. A. Ewald summation techniques in perspective: a survey. *Comput. Phys. Commun.* **95,** 73–92 (1996).

91. White, C. A. & Head-Gordon, M. Rotating around the quartic angular momentum barrier in fast multipole method calculations. *J. Chem. Phys.* **105,** 5061–5067 (1996).

92. Blelloch, G. & Narlikar, G. *A practical comparison of N-body algorithms* in *Parallel Algorithms* **30** (1997), 1–16.

93. Colby, R. H., Boris, D. C., Krause, W. E. & Tan, J. S. Polyelectrolyte conductivity. *Journal of Polymer Science Part B: Polymer Physics* **35,** 2951–2960 (1997).

94. Greengard, L. & Rokhlin, V. A new version of the Fast Multipole Method for the Laplace equation in three dimensions. *Acta Numer.* **6,** 229–269 (1997).

95. Kaatze, U. The dielectric properties of water in its different states of interaction. *Journal of Solution Chemistry* **26,** 1049–1112 (1997).

96. Lamm, G. & Pack, G. R. Calculation of dielectric constants near polyelectrolytes in solution. *The Journal of Physical Chemistry B* **101,** 959–965 (1997).

97. Stadler, J., Mikulla, R. & Trebin, H.-R. IMD: a Software Package for Molecular Dynamics Studies on Parallel Computers. *Int. J. Mod. Phys. C.* **8,** 1131–1140 (1997).

98. Warnick, K. F. *A differential forms approach to electromagnetics in anisotropic media* PhD thesis (Brigham Young University, 1997).

99. Widmann, A. H. & Adolf, D. B. A comparison of Ewald summation techniques for planar surfaces. *Computer Physics Communications* **107,** 167–186 (1997).

100. Arora, A. K. & Tata, B. Interactions, structural ordering and phase transitions in colloidal dispersions. *Advances in Colloid and Interface Science* **78,** 49–97 (1998).

101. Belloni, L. Ionic condensation and charge renormalization in colloidal suspensions. *Colloids and Surfaces* **A 140,** 227 (1998).

102. Deserno, M. & Holm, C. How to mesh up Ewald sums. I. A theoretical and numerical comparison of various particle mesh routines. *J. Chem. Phys.* **109,** 7678 –7693 (1998).

103. Deserno, M. & Holm, C. How to mesh up Ewald sums. II. An accurate error estimate for the Particle-Particle-Particle-Mesh algorithm. *J. Chem. Phys.* **109,** 7694 –7701 (1998).

104. Im, W., Beglov, D. & Roux, B. Continuum Solvation Model: Computation of electrostatic forces from numerical solutions to the Poisson-Boltzmann equation. *Comput. Phys. Commun.* **111,** 59–75 (1998).

105. Sperb, R. *An alternative to Ewald sums, Part 2: the Coulomb potential in a periodic system* tech. rep. 98-04 (Math. Sem. ETH Zürich, 1998).

106. Ahlrichs, P. & Dünweg, B. Simulation of a single polymer chain in solution by combining lattice Boltzmann and molecular dynamics. *Journal of Chemical Physics* **111,** 8225–8239 (1999).

107. Evans, D. F. & Wennerström, H. *The Colloidal Domain* (Wiley-VCH, New York, 1999).

108. Hoagland, D. A., Arvanitidou, E. & Welch, C. Capillary electrophoresis measurements of the free solution mobility for several model polyelectrolyte systems. *Macromolecules* **32,** 6180–6190 (Sept. 1999).

109. Hünenberger, P. H. & McCammon, J. A. Ewald artifacts in computer simulations of ionic solvation and ion–ion interaction: A continuum electrostatics study. *J. Chem. Phys.* **110,** 1856–72 (1999).

110. Jackson, J. D. *Classical Electrodynamics* (Wiley, New York, 3rd edition, 1999).

111. Sagui, C. & Darden, T. A. *P3M and PME: A comparison of the two methods* in *Simulation and theory of electrostatic interactions in solution* (eds Pratt, L. R. & Hummer, G.) (AIP, Sante Fe, New Mexico (USA), 1999), 104–113.

112. Deserno, M., Holm, C. & May, S. The fraction of condensed counterions around a charged rod: Comparison of Poisson-Boltzmann theory and computer simulations. *Macromolecules* **33,** 199–206 (2000).

113. Glasser, M. & Boersma, J. Exact values for the cubic lattice Green functions. *Journal of Physics A-Mathematical and General* **33,** 5017–5023 (July 2000).

114. Grzybowski, A., ź, E. Gwóźd & Bródka, A. Ewald summation of electrostatic interactions in molecular dynamics of a three-dimensional system with periodicity in two directions. *Phys. Rev. B* **61,** 6706–6712 (10 2000).

115. Han, J. & Craighead, H. G. Separation of long DNA molecules in a microfabricated entropic trap array. *Science* **288,** 1026–1029 (May 2000).

116. Hünenberger, P. Optimal charge-shaping functions for the particle-particle particle-mesh ($P^3M$) method for computing electrostatic interactions in molecular simulations. **113,** 10464–10476 (2000).

117. Porto, M. Ewald summation of electrostatic interactions of systems with finite extent in two of three dimensions. *J. Phys. A: Math. Gen.* **33,** 6211–6218 (2000).

118. Trottenberg, U, Oosterlee, C. W. & Schuller, A. *Multigrid* (2000).

119. Viovy, J. L. Electrophoresis of DNA and other polyelectrolytes: Physical mechanisms. *Reviews of Modern Physics* **72,** 813–872 (2000).

120. Hayashi, H., Tsuneda, S., Hirata, A. & Sasaki, H. Soft particle analysis of bacterial cells and its interpretation of cell adhesion behaviors in terms of DLVO theory. *Colloids and Surfaces B: Biointerfaces* **22,** 149 –157 (2001).

121. Long, D. & Ajdari, A. A note on the screening of hydrodynamic interactions, in electrophoresis, and in porous media. *European Physical Journal E* **4,** 29–32 (Jan. 2001).

122. Messina, R., Holm, C. & Kremer, K. Effect of colloidal charge discretization in the primitive model. *Euro. Phys. J. E.* **4,** 363–370 (2001).

123. Potts, D., Steidl, G. & Tasche, M. *Fast Fourier transforms for nonequispaced data: A tutorial* in *Modern Sampling Theory: Mathematics and Applications* (eds Benedetto, J. J. & Ferreira, P. J. S. G.) (Birkhäuser, Boston, MA, USA, 2001), 247 –270.

124. Schuler, L. D., Daura, X. & Van Gunsteren, W. F. An improved GROMOS96 force field for aliphatic hydrocarbons in the condensed phase. *Journal of Computational Chemistry* **22,** 1205–1218 (2001).

125. Arnold, A. & Holm, C. A novel method for calculating electrostatic interactions in 2D periodic slab geometries. *Chemical Physics Letters* **354,** 324–330 (2002).

126. Arnold, A. & Holm, C. MMM2D: A fast and accurate summation method for electrostatic interactions in 2D slab geometries. *Comput. Phys. Commun.* **148,** 327–348 (2002).

127. Frenkel, D. & Smit, B. *Understanding Molecular Simulation* Second (Academic Press, San Diego, 2002).

128. Grosberg, A. Y., Nguyen, T. T. & Shklovskii, B. I. Colloquium: The physics of charge inversion in chemical and biological systems. *Rev. Mod. Phys* **74,** 329 (2002).

129. Hairer, E., Lubich, C. & Wanner, G. *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations* (Springer, Heidelberg, 2002).

130. Katsikadelis, J. T. *Boundary Elements: Theory and Applications: Theory and Applications* (Elsevier, 2002).

131. Levin, Y. Electrostatic correlations: from plasma to biology. *Rep. Prog. Phys.* **65,** 1577–1632 (2002).

132. Maggs, A. C. Dynamics of a local algorithm for simulating Coulomb interactions. *Journal of Chemical Physics* **117,** 1975 (2002).

133. Maggs, A. C. & Rosseto, V. Local Simulation Algorithms for Coulombic interactions. *Physical Review Letters* **88,** 196402 (2002).

134. Messina, R. Image charges in spherical geometry: Application to colloidal systems. *Journal of Chemical Physics* **117,** 11062 (2002).

135. Messina, R. Spherical Colloids: Effect of Discrete Macroion Charge Distribution and Counterion Valence. *Physica A* **308,** 59–79 (2002).

136. Zheng, J. & Yeung, E. S. Anomalous radial migration of single DNA molecules in capillary electrophoresis. *Analytical chemistry* **74,** 4536–4547 (2002).

137. Aguado, A. & Madden, P. Ewald summation of electrostatic multipole interactions up to the quadrupolar level. *J. Chem. Phys.* **119,** 7471–7483 (2003).

138. Aubouy, M., Trizac, E. & Bocquet, L. Effective charge versus bare charge: an analytical estimate for colloids in the infinite dilution limit. *J. Phys. A: Math. Gen.* **36,** 5835–5840 (2003).

139. Gibbon, P. *PEPC : Pretty Efficient Parallel Coulomb-solver* Internal Report ZAM-IB-2003-05 (ZAM, Jülich, Forschungszentrum, 2003).

140. Potts, D. & Steidl, G. Fast Summation at nonequispaced knots by NFFTs. *SIAM J. Sci. Comput.* **24,** 2013 –2037 (2003).

141. Zheng, J. & Yeung, E. S. Mechanism for the separation of large molecules based on radial migration in capillary electrophoresis. *Analytical chemistry* **75,** 3675–3680 (2003).

142. Boda, D., Gillespie, D., Nonner, W., Henderson, D. & Eisenberg, B. Computing induced charges in inhomogeneous dielectric media: Application in a Monte Carlo simulation of complex ionic system. *Phys. Rev. E* **69,** 046702 (2004).

143. Bordi, F, Cametti, C & Colby, R. H. Dielectric spectroscopy and conductivity of polyelectrolyte solutions. *Journal of Physics: Condensed Matter* **16,** R1423 (2004).

144. Fenn, M. & Steidl, G. Fast NFFT based summation of radial functions. *Sampl. Theory Signal Image Process.* **3,** 1 –28 (2004).

145. Gibbon, P., Beg, F., Clark, E., Evans, R. & Zepf, M. Tree-code simulations of proton acceleration from laser-irradiated wire targets. *Physics of Plasmas* **11.** cited By (since 1996) 20, 4032–4040 (2004).

146. Horn, H. W., Swope, W. C., Pitera, J. W., Madura, J. D., Dick, T. J., Hura, G. L. & Head-Gordon, T. Development of an improved four-site water model for biomolecular simulations: TIP4P-Ew. *The Journal of chemical physics* **120,** 9665–9678 (2004).

147. Hutter, K & Jöhnk, K. *Continuum methods of physical modelling–Continuum mechanics, dimensional analysis, turbulence* 2004.

148. Kudin, K. N. & Scuseria, G. E. Revisiting infinite lattice sums with the periodic fast multipole method. *J. Chem. Phys.* **121,** 2886–2890 (2004).

149. Lobaskin, V., Dünweg, B. & Holm, C. Electrophoretic mobility of a charged colloidal particle: A computer simulation study. *Journal of Physics: Condensed Matter* **16,** S4063–S4073 (Sept. 2004).

150. Maggs, A. C. Auxilary field Monte Carlo for charged particles. *Journal of Chemical Physics* **120,** 3108 (2004).

151. Pasichnyk, I. & Dünweg, B. Coulomb interactions via local dynamics: A molecular-dynamics algorithm. *J. Phys.: Condens. Mat.* **16,** 3999–4020 (Sept. 2004).

152. Pasichnyk, I. *Novel simulation methods for Coulomb and hydrodynamic interactions* PhD thesis (Johannes-Gutenberg-Universität Mainz, 2004).

153. Potts, D., Steidl, G. & Nieslony, A. Fast convolution with radial kernels at nonequispaced knots. *Numer. Math.* **98,** 329 –351 (2004).

154. Rekvig, L., Hafskjold, B. & Smit, B. Molecular simulations of surface forces and film rupture in oil/water/surfactant systems. *Langmuir* **20,** 11583–11593 (2004).

155. Rottler, J. & Maggs, A. C. Local molecular dynamics with Coulombic interactions. *Physical Review Letters* **93,** 170201 (2004).

156. Rottler, J. & Maggs, A. C. A continuum, O(N) Monte Carlo algorithm for charged particles. *Journal of Chemical Physics* **120,** 3119 (2004).

157. Zheng, J., Li, H.-W. & Yeung, E. S. Manipulation of single DNA molecules via lateral focusing in a PDMS/glass microchannel. *The Journal of Physical Chemistry B* **108,** 10357–10362 (2004).

158. Adhikari, R., Stratford, K., Cates, M. & Wagner, A. Fluctuating Lattice Boltzmann. *Europhysics Letters* **71,** 473 (2005).

159. Arnold, A. & Holm, C. in *Advanced Computer Simulation Approaches for Soft Matter Sciences II* (eds Holm, C. & Kremer, K.) 59–109 (Springer, Berlin, 2005).

160. Arnold, A. & Holm, C. MMM1D: A method for calculating electrostatic interactions in one-dimensional periodic geometries. *J. Chem. Phys.* **123,** 144103 (2005).

161. Bordi, F., Cametti, C., Gili, T., Sennato, S., Zuzzi, S., Dou, S. & Colby, R. H. Solvent quality influence on the dielectric properties of polyelectrolyte solutions: A scaling approach. *Physical Review E* **72,** 031806 (3 2005).

162. Boroudjerdi, H., Kim, Y.-W., Naji, A., Netz, R. R., Schlagberger, X. & Serr, A. Statics and dynamics of strongly charged soft matter. *Physics Reports* **416,** 129–199 (2005).

163. Dehez, F., Martins-Costa, M., Rinaldi, D. & Millot, C. Long-range electrostatic interactions in hybrid quantum and molecular mechanical dynamics using a lattice summation approach. **122,** 234503 (2005).

164. Dobrynin, A. V. & Rubinstein, M. Theory of polyelectrolytes in solutions and at surfaces. *Progress in Polymer Science* **30,** 1049–1118 (2005).

165. Duncan, A., Sedgewick, R. D. & Coalson, R. D. Improved local lattice approach for Coulombic simulations. *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics* **71** (2005).

166. Gumerov, N. A. & Duraiswami, R. *Fast Multipole Methods for the Helmholtz Equation in Three Dimensions* (Elsevier, 2005).

167. Linse, P. Simulation of charged colloids in solution. *Adv. Polym. Sci.* **185,** 111–162 (2005).

168. Maggs, A. C. & Rottler, J. Auxiliary field simulation and Coulomb's law. *Computer Physics Communications* **169,** 160–165 (2005).

169. Phillips, J. C., Braun, R., Wang, W., Gumbart, J., Tajkhorshid, E., Villa, E., Chipot, C., Skeel, R. D., Kalé, L. & Schulten, K. Scalable molecular dynamics with NAMD. *J. Comput. Chem.* **26,** 1781–1802 (2005).

170. Siwy, Z., Kosinska, I. D., Fluinski, A. & Martin, C. R. Asymmetric Diffusion through Synthetic Nanopores. *Physical Review Letters* **94,** 048102 (2005).

171. Usta, O. B., Ladd, A. J. C. & Butler, J. E. Lattice Boltzmann simulations of the dynamics of polymer solutions in periodic and confined geometries. *Journal of Chemical Physics* **122,** 094902 (2005).

172. Bastian, P., Blatt, M., Dedner, A., Engwer, C., Klöfkorn, R., Kuttanikkad, S., Ohlberger, M. & Sander, O. *The distributed and unified numerics environment (DUNE)* in *19th Symposium on Simulatin Technique in Hannover, September 12-14* (2006).

173. Duncan, A., Sedgewick, R. D. & Coalson, R. D. Local simulation algorithms for Coulomb gases with dynamical dielectric effects. *Phys. Rev. E (Statistical, Nonlinear, and Soft Matter Physics)* **73,** 016705 (2006).

174. Gavryushov, S. & Linse, P. Effective Interaction Potentials for Alkali and Alkaline Earth Metal Ions in SPC/E Water and Prediction of Mean Ion Activity Coefficients. *Journal of Physical Chemistry B* **110,** 10878–10887 (2006).

175. Hedman, F. & Laaksonen, A. Ewald summation based on nonuniform fast Fourier transform. *Chem. Phys. Lett.* **425,** 142 –147 (2006).

176. Hess, B., Holm, C. & Vegt, N. van der. Modeling multi-body effects in ionic solutions with a concentration dependent dielectric permittivity. *Physical Review Letters* **96,** 147801 (2006).

177. Hess, B., Holm, C. & Vegt, N. van der. Osmotic Coeffcients of atomistic NACl (aq) force-fields. *Journal of Chemical Physics* **124,** 164509 (2006).

178. Keyser, U., Does, J. van der, Dekker, C. & Dekker, N. Optical tweezers for force measurements on DNA in nanopores. *Review of Scientific Instruments* **77,** 105105 (2006).

179. Kreer, T., Horbach, J. & Chatterji, A. Nonlinear effects in charge stabilized colloidal suspensions. English. *Phys. Rev. E* **74** (Aug. 2006).

180. Kunis, S., Potts, D. & Steidl, G. Fast Gauss transform with complex parameters using NFFTs. *J. Numer. Math.* **14,** 295 – 303 (2006).

181. Limbach, H. J., Arnold, A., Mann, B. A. & Holm, C. ESPResSo – An Extensible Simulation Package for Research on Soft Matter Systems. *Computer Physics Communications* **174,** 704–727 (May 2006).

182. Maggs, A. C. Multiscale Monte Carlo Algorithm for Simple Fluids. *Physical Review Letters* **97,** 197802–197804 (Nov. 2006).

183. Maggs, A. C. & Everaers, R. Simulating Nanoscale Dielectric Response. *Physical Review Letters* **96,** 230603–230604 (June 2006).

184. Ohshima, H. *Theory of colloid and interfacial electric phenomena* (Academic Press, 2006).

185. Smeets, R. M. M., Keyser, U. F., Krapf, D., Wu, M.-Y., Dekker, N. H. & Dekker, C. Salt dependence of ion transport and DNA translocation through solid-state nanopores. *Nano Letters* **6,** 89–95 (2006).

186. Sutmann, G. & Steffen, B. High-order compact solvers for the three dimensional Poisson equation. *J. Comp. Appl. Math.* **187,** 142–170 (2006).

187. Tessier, F. & Slater, G. W. Effective Debye length in closed nanoscopic systems: A competition between two length scales. *Electrophoresis* **27,** 686–693 (2006).

188. Butler, J. E., Usta, O. B., Kekre, R. & Ladd, A. J. Kinetic theory of a confined polymer driven by an external force and pressure-driven flow. *Physics of Fluids (1994-present)* **19,** 113101 (2007).

189. Cross, J. D., Strychalski, E. A. & Craighead, H. Size-dependent DNA mobility in nanochannels. *Journal of Applied Physics* **102,** 024701–024701 (2007).

190. Dachsel, H., Hofmann, M. & Rünger, G. *Library Support for Parallel Sorting in Scientific Computations* in *Proc. of the 13th International Euro-Par Conference* **4641** (Springer, 2007), 695–704.

191. Dekker, C. Solid-state nanopores. *Nature Nanotech* **2,** 209–215 (2007).

192. Dünweg, B., Schiller, U. & Ladd, A. J. C. Statistical mechanics of the fluctuating Lattice Boltzmann equation. *Phys. Rev. E* **76,** 36704 (2007).

193. Griebel, M., Knapek, S. & Zumbusch, G. *Numerical Simulation in Molecular Dynamics – Numerics, Algorithms, Parallelization, Applications* 1st, 1–431 (Springer-Verlag, Heidelberg, 2007).

194. Krishnan, M., Monch, I. & Schwille, P. Spontaneous Stretching of DNA in a Two-Dimensional Nanoslit. *Nano Letters* **7,** 1270–1275 (2007).

195. Laachi, N., Declet, C., Matson, C. & Dorfman, K. D. Nonequilibrium Transport of Rigid Macromolecules in Periodically Constricted Geometries. *Physical Review Letters* **98,** 98106 (2007).

196. Lobaskin, V., Dünweg, B., Medebach, M., Palberg, T. & Holm, C. Electrophoresis of Colloidal Dispersions in the Low-Salt Regime. *Physical Review Letters* **98,** 176105 (Apr. 2007).

197. Rekvig, L. & Frenkel, D. Molecular simulations of droplet coalescence in oil/water/surfactant systems. *J. Chem. Phys.* **127,** 134701 (2007).

198. Rottler, J. Local electrostatics algorithm for classical molecular dynamics simulations. *Journal of Chemical Physics* **127,** 134104–134109 (Oct. 2007).

199. Skeel, R. D., Hardy, D. & Phillips, J. Correcting mesh-based force calculations to conserve both energy and momentum in molecular dynamics simulations. *J. Comp. Phys.* **225,** 1–5 (2007).

200. Strogatz, S. *The end of insight* (ed Brockman, J.) 130–131 (Harper Perennial, New York, 2007).

201. Tyagi, S., Arnold, A. & Holm, C. ICMMM2D: An accurate method to include planar dielectric interfaces via image charge summation. *Journal of Chemical Physics* **127,** 154723 (2007).

202. Usta, O. B., Butler, J. E. & Ladd, A. J. C. Transverse Migration of a Confined Polymer Driven by an External Force. *Physical Review Letters* **98,** 98301 (2007).

203. Zhao, Z., Kovvali, N., Lin, W., Ahn, C.-H., Couchman, L. & Carin, L. Volumetric fast multipole method for modeling Schrödinger's equation. *J. Comput. Phys.* **224,** 941–955 (2007).

204. Ballenegger, V., Cerdà, J. J., Lenz, O. & Holm, C. The Optimal P3M Algorithm for Computing Electrostatic Energies in Periodic Systems. *J. Chem. Phys.* **128,** 034109 (2008).

205. Bastian, P., Blatt, M., Dedner, A., Engwer, C., Klöfkorn, R., Kornhuber, R., Ohlberger, M. & Sander, O. A Generic Grid Interface for Parallel and Adaptive Scientific Computing. Part II: Implementation and Tests in DUNE. *Computing* **82,** 121–138 (2008).

206. Bolten, M. *Multigrid methods for structured grids and their application in particle simulation* PhD thesis (Bergische Universität Wuppertal, Wuppertal, 2008).

207. Cerdà, J. J., Ballenegger, V., Lenz, O. & Holm, C. P3M Algorithm for Dipolar Interactions. *J. Chem. Phys.* **129,** 234104 (2008).

208. Duong-Hong, D., Wang, J.-S., Liu, G., Chen, Y., Han, J. & Hadjiconstantinou, N. Dissipative particle dynamics simulations of electroosmotic flow in nano-fluidic devices. English. *Microfluidics and Nanofluidics* **4,** 219–225 (2008).

209. Duong-Hong, D., Han, J., Wang, J.-S., Hadjiconstantinou, N. G., Chen, Y. Z. & Liu, G.-R. Realistic simulations of combined DNA electrophoretic flow and EOF in nano-fluidic devices. *Electrophoresis* **29,** 4880–4886 (2008).

210. Frank, S. & Winkler, R. G. Polyelectrolyte electrophoresis: field effects and hydrodynamic interactions. *Europhysics Letters* **83,** 38004 (2008).

211. Graaf, J. de, Zwanikken, J., Bier, M., Baarsma, A., Oloumi, Y., Spelt, M. & Roij, R. van. Spontaneous charging and crystallization of water droplets in oil. *J. Chem. Phys.* **129,** 194701 (2008).

212. Grass, K. & Holm, C. On the importance of hydrodynamic interactions in polyelectrolyte electrophoresis. *Journal of Physics: Condensed Matter* **20,** 494217 (2008).

213. Grass, K., Böhme, U., Scheler, U., Cottet, H. & Holm, C. Importance of Hydrodynamic Shielding for the Dynamic Behavior of Short Polyelectrolyte Chains. *Physical Review Letters* **100,** 096104 (2008).

214. Hess, B., Kutzner, C., van der Spoel, D. & Lindahl, E. GROMACS 4: Algorithms for Highly Efficient, Load-Balanced, and Scalable Molecular Simulation. *J. Chem. Theory Comput.* **4,** 435–447 (2008).

215. Krishnan, M., Petrek, Z., Mnch, I. & Schwille, P. Electrostatic Self-Assembly of Charged Colloids and Macromolecules in a Fluidic Nanoslit. *Small* **4,** 1900–1906 (2008).

216. Levrel, L. & Maggs, A. C. Boundary conditions in local electrostatics algorithms. *Journal of Chemical Physics* **128** (2008).

217. Linse, P. Electrostatics in the presence of spherical dielectric discontinuities. *Journal of Chemical Physics* **128,** 214505 (2008).

218. Lu, B. Z., Zhou, Y. C., Holst, M. J. & McCammon, J. A. Recent progress in numerical methods for the Poisson-Boltzmann equation in biophysical applications. *Commun. Comput. Phys.* **3,** 973–1009 (May 2008).

219. Pasquali, S. & Maggs, A. C. Fluctuation-induced interactions between dielectrics in general geometries. *Journal of Chemical Physics* **129** (2008).

220. Pasquali, S., Nitti, F. & Maggs, A. C. Numerical methods for fluctuation-driven interactions between dielectrics. *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics* **77** (2008).

221. Reščič, J. & Linse, P. Potential of mean force between charged colloids: Effect of dielectric discontinuities. *Journal of Chemical Physics* **129,** 114505 (2008).

222. Strychalski, E. A., Levy, S. L. & Craighead, H. G. Diffusion of DNA in Nanoslits. *Macromolecules* **41,** 7716–7721 (2008).

223. Thompson, D. & Rottler, J. Local Monte Carlo for electrostatics in anisotropic and nonperiodic geometries. *Journal of Chemical Physics* **128** (2008).

224. Tyagi, S., Arnold, A. & Holm, C. Electrostatic layer correction with image charges: A linear scaling method to treat slab 2D + h systems with dielectric interfaces. *Journal of Chemical Physics* **129,** 204102 (2008).

225. Ballenegger, V., Arnold, A. & Cerda, J. J. Simulations of non-neutral slab systems with long-range electrostatic interactions in two-dimensional periodic boundary conditions. *Journal of Chemical Physics* **131,** 094107 (2009).

226. BerkáUsta, O *et al.* Flow injection of polymers into nanopores. *Soft Matter* **5,** 4575–4579 (2009).

227. Frank, S. & Winkler, R. G. Mesoscale hydrodynamic simulation of short polyelectrolytes in electric fields. English. *Journal of Chemical Physics* **131** (2009).

228. Grass, K. & Holm, C. Polyelectrolytes in electric fields: Measuring the dynamical effective charge and effective friction. *Soft Matter* **5,** 2079–2092 (2009).

229. Keiner, J., Kunis, S. & Potts, D. Using NFFT3 - a Software Library for Various Nonequispaced Fast Fourier Transforms. *ACM Trans. Math. Software* **36,** Article 19, 1 –30 (2009).

230. Lin, P.-K., Lin, K.-h., Fu, C.-C., Lee, K.-C., Wei, P.-K., Pai, W.-W., Tsao, P.-H., Chen, Y.-L. & Fann, W. One-dimensional dynamics and transport of DNA molecules in a quasi-two-dimensional nanoslit. *Macromolecules* **42,** 1770–1774 (2009).

231. Lipar-Oštir, I., Zalar, P., Bešter-Rogač, M., Pohar, C. & Vlachy, V. Electric Conductivity of Aqueous Solutions of Poly(anetholesulfonic acid) and Its Alkaline Salts. *The Journal of Physical Chemistry B* **113,** 2705–2711 (2009).

232. Pasquali, S. & Maggs, A. C. Numerical studies of Lifshitz interactions between dielectrics. *Physical Review A: Atomic, Molecular, and Optical Physics* **79** (2009).

233. Qin, P., Xu, Z., Cai, W. & Jacobs, D. Image charge methods for a three-dielectric-layer hybrid solvation model of biomolecules. *Commun. Comput. Phys.* **6,** 955–977 (2009).

234. Slater, G. W., Holm, C., Chubynsky, M. V., Haan, H. W. de, Dubé, A., Grass, K., Hickey, O. A., Kingsburry, C., Sean, D., Shendruk, T. N. & Zhan, L. Modeling the separation of macromolecules: A review of current computer simulation methods. *Electrophoresis* **30,** 792–818 (Mar. 2009).

235. Slater, G. W. DNA gel electrophoresis: The reptation model(s). *Electrophoresis* **30,** S181–S187 (2009).

236. (eds Sutmann, G., Gibbon, P. & Lippert, T.) *Fast Methods for Long Range Interactions in Complex Systems* (IAS, Jülich, 2009).

237. Wang, Z. Y. & Ma, Y. Q. Monte Carlo determination of mixed electrolytes next to a planar dielectric interface with different surface charge distributions. *J. Chem. Phys.* **131,** 244715 (2009).

238. Wanunu, M., Morrison, W., Rabin, Y., Grosberg, A. & Meller, A. Electrostatic focusing of unlabelled DNA into nanoscale pores using a salt gradient. *Nature nanotechnology* **5,** 160–165 (2009).

239. Bichoutskaia, E., Boatwright, A. L., Khachatourian, A. & Stace, A. J. Electrostatic analysis of the interactions between charged particles of dielectric materials. *J. Chem. Phys.* **133,** 024105 (2010).

240. Cho, J. & Dorfman, K. D. Brownian dynamics simulations of electrophoretic {DNA} separations in a sparse ordered post array. *Journal of Chromatography A* **1217,** 5522 –5528 (2010).

241. Dachsel, H. An Error-Controlled Fast Multipole Method. *J. Chem. Phys.* **132,** 119901 (2010).

242. Dorfman, K. D. DNA electrophoresis in microfabricated devices. *Rev. Mod. Phys.* **82,** 2903–2947 (4 2010).

243. Duval, J. F. L. & Gaboriaud, F. Progress in electrohydrodynamics of soft microbial particle interphases. *Current Opinion in Colloid & Interface Science* **15,** 184–195 (June 2010).

244. Fayad, G. & Hadjiconstantinou, N. Realistic Brownian Dynamics simulations of biological molecule separation in nanofluidic devices. English. *Microfluidics and Nanofluidics* **8,** 521–529 (2010).

245. French, R. H. *et al.* Long range interactions in nanoscale science. *Rev. Mod. Phys.* **82,** 1887–1944 (2010).

246. Gibbon, P., Speck, R., Karmakar, A., Arnold, L., Frings, W., Berberich, B., Reiter, D. & Masek, M. Progress in mesh-free plasma simulation with parallel tree codes. *IEEE Transactions on Plasma Science* **38.** cited By (since 1996) 3, 2367–2376 (2010).

247. Grass, K & Holm, C. Mesoscale modelling of polyelectrolyte electrophoresis. *Faraday Discussions* **144,** 57–70 (2010).

248. Holdren, J. P., Lander, E., Varmus, H., Bierbaum, R., Cassel, C., Chyba, C., S. James Gates, J., Jackson, S. A., Levin, R. C., Mirkin, C., Molina, M., Moniz, E. J. & Mundie, C. *Designing a Digital Future: Federally Funded Research and Development Networking and Information Technology* 2010.

249. Kekre, R., Butler, J. E. & Ladd, A. J. Role of hydrodynamic interactions in the migration of polyelectrolytes driven by a pressure gradient and an electric field. *Physical Review E* **82,** 050803 (2010).

250. Kesselheim, S., Sega, M. & Holm, C. The ICC* Algorithm: A fast way to include dielectric boundary effects into molecular dynamics simulations. *arXiv preprint arXiv:1003.1271* (2010).

251. Krishnan, M., Mojarad, N., Kukura, P. & Sandoghdar, V. Geometry-induced electrostatic trapping of nanometric objects in a fluid. *Nature* **467,** 692–695 (2010).

252. Neelov, A. & Holm, C. Interlaced P3M Algorithm with Analytical and ik-Differentiation. *J. Chem. Phys.* **132** (2010).

253. Smiatek, J. & Schmid, F. Polyelectrolyte Electrophoresis in Nanochannels: A Dissipative Particle Dynamics Simulation. *Journal of Physical Chemistry B* **114,** 6266–6272 (2010).

254. Trahan, D. W. & Doyle, P. S. DNA Collisions with a Large, Conducting Post. *Macromolecules* **43,** 5424–5432 (2010).

255. Tyagi, C., Süzen, M., Sega, M., Barbosa, M., Kantorovich, S. & Holm, C. An iterative, fast, linear-scaling method for computing induced charges on arbitrary dielectric boundaries. *J. Chem. Phys.* **132** (2010).

256. Tyagi, C., Süzen, M., Sega, M., Barbosa, M., Kantorovich, S. & Holm, C. An iterative, fast, linear-scaling method for computing induced charges on arbitrary dielectric boundaries. *Journal of Chemical Physics* **132,** 1154112 (2010).

257. Wang, Z. G. Fluctuation in electrolyte solutions: The self energy. *Phys. Rev. E* **81,** 021501 (2010).

258. Arnold, A. *Fourier Transformed-Based Methods for Long-Range Interactions: Ewald, $P^3M$ and More* in *Fast Methods for Long-Range Interactions in Complex Systems* (eds Sutmann, G., Gibbon, P. & Lippert, T.) **6** (FZ Jülich, 2011), 59–109.

259. Arnold, A., Lenz, O. & Holm, C. *Simulating Charged Systems with* ESPResSo in *Fast Methods for Long-Range Interactions in Complex Systems* (eds Sutmann, G., Gibbon, P. & Lippert, T.) **6** (FZ Jülich, 2011), 159–166.

260. Ballenegger, V., Cerdà, J. J. & Holm, C. Removal of spurious self-interactions in particle-mesh methods. *Comput. Phys. Commun.* **182,** 1919–1923 (Sept. 2011).

261. Bonthuis, D. J., Gekle, S. & Netz, R. R. Dielectric Profile of Interfacial Water and its Effect on Double-Layer Capacitance. *Physical Review Letters* **107,** 166102 (16 2011).

262. Cerda, J. J., Ballenegger, V. & Holm, C. Particle-Particle Particle-Mesh Method for Dipolar Interactions: On Error Estimates and Efficiency of Schemes with Analytical Differentiation and Mesh Interlacing. *J. Chem. Phys.* **135** (2011).

263. Cerdà, J. J., Ballenegger, V. & Holm, C. Particle-particle particle-mesh method for dipolar interactions: On error estimates and efficiency of schemes with analytical differentiation and mesh interlacing. *J. Chem . Phys.* **135,** 184110 (2011).

264. Gee, M. B., Cox, N. R., Jiao, Y., Bentenitis, N., Weerasinghe, S. & Smith, P. E. A Kirkwood-Buff derived force field for aqueous alkali halides. *Journal of Chemical Theory and Computation* **7,** 1369–1380 (2011).

265. Kesselheim, S., Sega, M. & Holm, C. Applying ICC* to DNA translocation. Effect of dielectric boundaries. *Computer Physics Communications* **182,** 33–35 (Jan. 2011).

266. Pippig, M. *PFFT, Parallel FFT subroutine library* 2011.

267. Pippig, M. *PNFFT, Parallel Nonequispaced FFT subroutine library* 2011.

268. Pippig, M. & Potts, D. *Particle simulation based on nonequispaced fast Fourier transforms* in *Fast Methods for Long-Range Interactions in Complex Systems* (eds Sutmann, G., Gibbon, P. & Lippert, T.) (Forschungszentrum Jülich, Jülich, 2011), 131 – 158.

269. Rottler, J. & Maggs, A. C. Long-ranged Electrostatics from Local Algorithms. *Soft Matter* **7,** 3260–3267 (2011).

270. Schlaich, A. *An iterative Poisson Boltzmann solver for regions with dielectric mismatch* PhD thesis (University of Stuttgart, 2011).

271. Smiatek, J. & Schmid, F. Mesoscopic simulations of electroosmotic flow and electrophoresis in nanochannels. *Comp. Phys. Comm.* **182.** Computer Physics Communications Special Edition for Conference on Computational Physics Trondheim, Norway, June 23-26, 2010, 1941–1944 (2011).

272. Speck, R., Arnold, L. & Gibbon, P. Towards a petascale tree code: Scaling and efficiency of the PEPC library. *J. Comput. Sci.* **2.** Simulation Software for Supercomputers, 138 –143 (2011).

273. Walker, D. A., Kowalczyk, B., La Cruz, M. O. de & Grzybowski, B. A. Electrostatics at the nanoscale. *Nanoscale* **3,** 1316–1344 (2011).

274. Weik, F. *Implementation of an improved P3M algorithm* MA thesis (Universität Stuttgart, 2011).

275. dos Santos, A. P., Bakhshandeh, A. & Levin, Y. Effects of the dielectric discontinuity on the counterion distribution in a colloidal suspension. *J. Chem. Phys.* **135,** 044124 (2011).

276. Ballenegger, V., Cerdà, J. J. & Holm, C. How to Convert SPME to P3M: Influence Functions and Error Estimates. *J. Chem. Theory Comput.* **8,** 936–947 (2012).

277. Bonthuis, D. J., Gekle, S. & Netz, R. R. Profile of the Static Permittivity Tensor of Water at Interfaces: Consequences for Capacitance, Hydration Interaction and Ion Adsorption. *Langmuir* **28,** 7679–7694 (2012).

278. Gan, Z., Xing, X. & Xu, Z. Effects of image charges, interfacial charge discreteness, and surface roughness on the zeta potential of spherical electric double layers. *J. Chem. Phys.* **137,** 034708 (2012).

279. Heyda, J. & Dzubiella, J. Ion-specific counterion condensation on charged peptides: Poisson–Boltzmann vs. atomistic simulations. *Soft Matter* **8,** 9338–9344 (2012).

280. Hickey, O. A., Shendruk, T. N., Harden, J. L. & Slater, G. W. Simulations of Free-Solution Electrophoresis of Polyelectrolytes with a Finite Debye Length Using the Debye-Hückel Approximation. *Physical Review Letters* **109,** 098302 (9 2012).

281. Jadhao, V., Solis, F. J. & Cruz, M. O. de la. Simulation of charged systems in heterogeneous dielectric media via a true energy functional. *Physical Review Letters* **109,** 223905 (2012).

282. Jülich Supercomputing Centre, F. J. *PEPC – The Pretty Efficient Coulomb Solver* `http://www.fz-juelich.de/ias/jsc/pepc`. [Online; accessed 01-October-2012]. 2012.

283. Kesselheim, S., Sega, M. & Holm, C. Effects of dielectric mismatch and chain flexibility on the translocation barriers of charged macromolecules through solid state nanopores. *Soft Matter* **8,** 9480–9486 (2012).

284. Michel, A. *Eine Analyse elektrostatischer Algorithmen in der Scafacos-Bibliothek* Eingereicht. MA thesis (Universität Stuttgart, 2012).

285. Roehm, D. & Arnold, A. Lattice Boltzmann simulations on GPUs with ESPResSo. *The European Physical Journal Special Topics* **210,** 89–100 (2012).

286. Röhm, D. & Arnold, A. Lattice Boltzmann simulations on GPUs with ESPResSo. *The European Physical Journal-Special Topics* **210,** 89–100 (2012).

287. Shendruk, T. N., Hickey, O. A., Slater, G. W. & Harden, J. L. Electrophoresis: When hydrodynamics matter. *Curr. Opin. Colloid In.* **17,** 74 –82 (2012).

288. Speck, R., Ruprecht, D., Krause, R., Emmett, M., Minion, M., Winkel, M. & Gibbon, P. A massively space-time parallel N-body solver. *Proceedings of "The International Conference for High-Performance Computing, Networking, Storage and Analysis 2012 (SC'12)"* (2012).

289. Winkel, M., Speck, R., Hübner, H., Arnold, L., Krause, R. & Gibbon, P. A massively parallel, multi-disciplinary Barnes-Hut tree code for extreme-scale N-body simulations. *Computer Physics Communications* **183,** 880 –889 (2012).

290. Yan, K., Chen, Y.-Z., Han, J., Liu, G.-R., Wang, J.-S. & Hadjiconstantinou, N. Dissipative particle dynamics simulation of field-dependent DNA mobility in nanoslits. *Microfl. Nanofl.* **12,** 157–163 (1 2012).

291. Bolten, M., Fahrenberger, F., Gähler, F., Halver, R., Heber, F., Hofmann, M., Kabadshow, I., Lenz, O. & Pippig, M. *ScaFaCoS Manual* (`http://www.scafacos.de`, 2013).

292. Cisneros, G. A., Babin, V. & Sagui, C. in *Biomolecular Simulations* 243–270 (Springer, 2013).

293. Jadhao, V., Solis, F. J. & Cruz, M. O. de la. A variational formulation of electrostatics in a medium with spatially varying dielectric permittivity. *J. Chem. Phys.* **138,** 054119 (2013).

294. Krishnan, M. Electrostatic free energy for a confined nanoscale object in a fluid. *The Journal of Chemical Physics* **138,** – (2013).

295. Krishnan, M. Erratum: "Electrostatic free energy for a confined nanoscale object in a fluid" [J. Chem. Phys.138, 114906 (2013)]. *The Journal of Chemical Physics* **139** (2013).

296. Nakamura, I. & Wang, Z.-G. Effects of dielectric inhomogeneity in polyelectrolyte solutions. *Soft Matter* **9,** 5686–5690 (24 2013).

297. Pippig, M. PFFT - An Extension of FFTW to Massively Parallel Architectures. *SIAM J. Sci. Comput.* **35,** C213 –C236 (2013).

298. Pippig, M. & Potts, D. Parallel Three-Dimensional Nonequispaced Fast Fourier Transforms and Their Application to Particle Simulation. *SIAM J. Sci. Comput.* **35,** C411 –C437 (2013).

299. Pronk, S., Páll, S., Schulz, R., Larsson, P., Bjelkmar, P., Apostolov, R., Shirts, M. R., Smith, J. C., Kasson, P. M., Spoel, D. van der, Hess, B. & Lindahl, E. GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit. *Bioinformatics* **29,** 845–854 (2013).

300. Semenov, I., Raafatnia, S., Sega, M., Lobaskin, V., Holm, C. & Kremer, F. Electrophoretic mobility and charge inversion of a colloidal particle studied by single-colloid electrophoresis and molecular dynamics simulations. *Phys. Rev. E* **87,** 022302 (2 2013).

301. Xue, C. & Deng, S. Coulomb Green's function and image potential near a planar diffuse interface, revisited. *Comput. Phys. Commun.* **184,** 51 –59 (2013).

302. Zwanikken, J. W. & Cruz, M. O. de la. Tunable soft structure in charged fluids confined by dielectric interfaces. *Proceedings of the National Academy of Sciences U.S. A.* **110,** 5301–5308 (2013).

303. Cametti, C. Does Electrical Conductivity of Linear Polyelectrolytes in Aqueous Solutions Follow the Dynamic Scaling Laws? A Critical Review and a Summary of the Key Relations. *Polymers* **6,** 1207–1231 (2014).

304. Hickey, O. A., Holm, C. & Smiatek, J. Lattice-Boltzmann simulations of the electrophoretic stretching of polyelectrolytes: The importance of hydrodynamic interactions. *The Journal of Chemical Physics* **140,** – (2014).

305. Ma, M. & Xu, Z. Self-consistent field model for strong electrostatic correlations and inhomogeneous dielectric media. *The Journal of Chemical Physics* **141,** – (2014).

306. Raafatnia, S., Hickey, O. A. & Holm, C. Mobility Reversal of Polyelectrolyte-Grafted Colloids in Monovalent Salt Solutions. *Phys. Rev. Lett.* **113,** 238301 (23 2014).

307. Smiatek, J, Wohlfarth, A & Holm, C. The solvation and ion condensation properties for sulfonated polyelectrolytes in different solvents-a computational study. *New Journal of Physics* **16,** 025001 (2014).

308. Curtis, R. A. & Lue, L. Depletion forces due to image charges near dielectric discontinuities. *Current Opinion in Colloid & Interface Science* **20,** 19 –23 (2015).

309. Raafatnia, S., Hickey, O. A. & Holm, C. Electrophoresis of a Spherical Polyelectrolyte-Grafted Colloid in Monovalent Salt Solutions: Comparison of Molecular Dynamics Simulations with Theory and Numerical Calculations. *Macromolecules* **48,** 775–787 (2015).

310. Skyner, R., McDonagh, J., Groom, C. R., Mourik, T. van & Mitchell, J. A Review of Methods for the Calculation of Solution Free Energies and the Modelling of Systems in Solution. *Physical Chemistry Chemical Physics* (2015).

311. Wang, R. & Wang, Z.-G. On the theoretical description of weakly charged surfaces. *The Journal of Chemical Physics* **142,** 104705 (2015).

312. Arnold, A. & Pippig, M. Equivalence of P3M and NFFT-Based Fast Summation Methods. **in preparation.**

313. Distributed, D. & Environment, U. N. *http://dune-project.org*

314. *ESPResSo – Extensible Simulation Package for Research on Soft Matter*

315. *ESPResSo homepage* http://espressomd.org.

316. *IPBS homepage on GitHub* `http://ipbs.github.com/`.

317. *JUGENE – Jülich Blue Gene/P*

318. *JUQUEEN website*

319. *JUROPA – Jülich Research on Petaflop Architectures*

320. *ScaFaCoS - Scalable Fast Coloumb Solvers*

# A. A sample MEMD script

In this appendix, we want to give an example of a simple and typical script how the ESPResSo software package works and how the MEMD algorithm would be used within it. The details are explained as comments (dark green) in the script itself.

This simulation script will create an electrically neutral system with 10 000 particles at a volumetric density of 0.3 where half of the particles are negatively charged and the other half is positively charged. The particles interact pairwise with a short-ranged Lennard-Jones potential [14] and the electrostatic interaction are calculated using the MEMD algorithm. The system assumes a normal dielectric background of $\varepsilon_W = 78.5$ for water, and it contains a dielectric wall with $\varepsilon = 1$ at the $x = 0$ box boundary. The script will output the total energy of the system on the screen several times and is not meant to be a physical simulation, merely a conceptual example.

The relevant lines for the MEMD algorithm are line 88 and lines 101−102. In the former, we turn on the inclusion of electrostatic interactions using the MEMD algorithm and its tuning routine. In the latter lines, we manually set a local permittivity on a lattice link. The command intentionally contains the bjerrum length $l_B$ again to remind the user that the permittivity $\varepsilon(\mathbf{r})$ is a relative value and is set in reference to the given bjerrum length which corresponds to $\varepsilon = 1$.

```
1  # System parameters
2  ############################################################
3
4  # System size
5  set density  0.3
6  set temp     1.0
7  set num_part 10000
8
9  # Tuning parameters
```

```
10 set time_step    0.01
11 set skin         0.3
12
13
14 # Interaction parameters
15 ############################################################
16
17 # Lennard-Jones parameters
18 set lj_eps      0.25
19 set lj_sig      1.0
20 set lj_cut      1.12246
21 set lj_shift    0.25
22 set lj_off      0.0
23
24 # Coulomb parameters
25 set bjerrum       0.714
26
27
28 # Integration parameters
29 ############################################################
30
31 # number of simulation steps
32 set int_steps     100000
33 set int_n_times   100
34
35
36 #  Create System
37 ############################################################
38
39 # calculate box size from density
40 set box_l [ expr pow( ($num_part/$density), (1.0/3.0) ) ]
41 # setmd is an ESPResSo command to set simulation parameters
42 setmd box_l $box_l $box_l $box_l
43
44 # loop over number of particles
45 for {set i 0} { $i < $num_part } {incr i} {
46   # create random position in box
47   set posx [expr $box_l*[t_random]]
48   set posy [expr $box_l*[t_random]]
49   set posz [expr $box_l*[t_random]]
50
51   # create particle with ESPResSo command part
52   part $i pos $posx $posy $posz
53   # give positive charge to even particle numbers
```

```
  if {[expr $i % 2 == 0]} {
    part $i q +1.0 type 1
  } else {
    part $i q -1.0 type 0
  }
}

#  Short-range Interactions
#############################################################

# inter is another ESPResSo command to set interactions
inter 0 0 lennard-jones $lj_eps $lj_sig $lj_cut $lj_shift $lj_off
inter 1 0 lennard-jones $lj_eps $lj_sig $lj_cut $lj_shift $lj_off
inter 1 1 lennard-jones $lj_eps $lj_sig $lj_cut $lj_shift $lj_off


#  Warmup Integration
#############################################################

puts "Warmup."
setmd time_step $time_step
setmd skin $skin
# set a temperature for NVT ensemble
thermostat langevin $temp 1.0

# use an external script extension to warm up particles if
# they are randomly placed very close
source md_tools.tcl
grow_lj { {0 0} {0 1} {1 1} }

#  Electrostatics
#############################################################

# setup MEMD coulomb interaction with given bjerrum length
inter coulomb $bjerrum memd tune
# get lattice size of MEMD algorithm, at sixth position
set memd_lattice [ lindex [ inter coulomb ] 5 ]
# calculate epsilon as 1/78.5th of what corresponds to $bjerrum
set small_eps [ expr 1.0/78.5 ]

# for all sites on plane x=0
set x 0
for {set y 0} { $y <= $memd_lattice } {incr y} {
  for {set z 0} { $z <= $memd_lattice } {incr z} {
```

```
98      # for y- and z-direction
99      for {set dir 1} { $dir <= 2 } {incr dir} {
100       # actually set relative epsilon
101       inter coulomb $bjerrum memd localeps
102             node $x $y $z dir $dir eps $small_eps
103     }
104   }
105 }
106
107
108 #  Integration
109 ##############################################################
110
111 puts "Starting simulation."
112
113 # We integrate ($int_steps * $int_n_times) steps
114 for {set i 0} { $i < $int_n_times } {incr i} {
115   integrate $int_steps
116   # analyze the total energy and output it on screen
117   puts "Total energy: [analyze energy total]"
118 }
119
120 puts "Finished simulation."
```

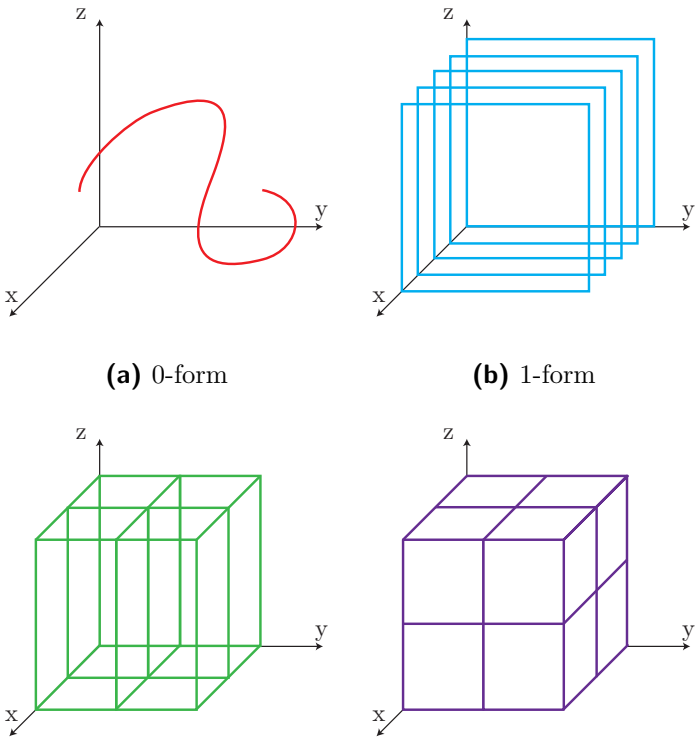
# B. Permittivity as a differential form

We have worked on the possibility of treating the dielectric permittivity in tensor form, since the assumption that it merely has a value and a direction is oversimplified. Diving into tensor analysis and differential forms would, however, go beyond the scope of this thesis, and there are individual theses written on just this subject [98]. We will, however, sketch some ideas we took away from this detour and explain why it will not work with this algorithm.

Simply put, the calculus of quantities that can be integrated is called the calculus of differential forms. The degree of these forms is the number of spatial dimensions of the area over which it is integrated. Since in simulations, we are working in three-dimensional space, only 0-forms, 1-forms, 2-forms, and 3-forms can exist. Hereby, 0-forms are nothing but functions whose integration consists of evaluation the function at a certain point. A 1-form is integrated over a path and can be depicted using a surface, as shown in figure B.1.

A 2-form is integrated over a surface using an anti-symmetrized tensor product as exterior product. Their graphical representation is a tube, constructed from two surfaces. As the coefficients of a 2-form increase, the tube becomes tighter, and the tubes are oriented in the direction of the associated dual vector. A 3-form is a volume element, and just like for 2-forms, the boxes get denser and more closely spaced for higher coefficients. Forms of degree greater than three vanish in three-dimensional space by the anti-commutativity of the exterior product. The exterior product of two differential forms results in a differential form that has the dimension of the sum of the two forms.

The connection to our algorithmic approach is shown in the graphical representations. In tensorial electrodynamics, the **E**- and **H**-fields, and




**(a)** 0-form **(b)** 1-form

**(c)** 2-form **(d)** 3-form

**Figure B.1.:** Differential forms and their graphical representations in space: (a) 0-forms are functions that can be evaluated in every point. (b) 1-forms, here $dx$, can be seen as surfaces with a value and an associated orientation. (c) 2-forms, like $dy\,dz$ are tubes formed by the superposition of surfaces $dy$ and $dz$. (d) 3-forms are represented by volumes, consisting of three sets of surfaces, here $dx\,dy\,dz$.

since we disregard the magnetic permeability $\mu$ also the **B**-field, are 1-forms represented by surfaces on a lattice. The electric flux density **D** and the current density **j** are 2-forms and can be represented by a tube around a lattice link. The electric charge density $\rho$ is a 3-form living in the lattice cubes, with coefficients equal to the charge density scalars

on the lattice vertices.

We can already see that the dielectric permittivity $\varepsilon$ is necessarily a 1-form, from the dimensional argument of the definition $\mathbf{D} := \varepsilon \wedge \mathbf{E}$.

This explains our choices where we store each variable in the simulation: The magnetic fields will be placed on lattice surfaces, the displacement fields on links (representing tubes around those links), just like the currents $\mathbf{j}$ and the permittivity $\varepsilon$. The electric charge density is stored as separate coefficients on the lattice vertices.

The problem with this formalism is that our whole algorithm is based on the assumption that the Helmholtz decomposition

$$\mathbf{D} = -\varepsilon \nabla \Phi + \nabla \times \mathbf{A} = \mathbf{D}^{||} + \mathbf{D}^{\perp} \tag{B.1}$$

holds and that we can treat the dynamics of the transversal fields separate from the electric field values. But this assumption is no longer valid if $\varepsilon$ has a tensor form. This is not a catastrophe, since the tensorial nature of $\varepsilon$ only shows up in very uncommon media and has never been needed for coarse-grained simulations. But if it would have been possible, it would be nice to have the option available. This way, we at least learned about the graphical representation of the relevant values in our algorithmic construction.