

Einführung in die Moleküldynamiksimulation von Proteinen

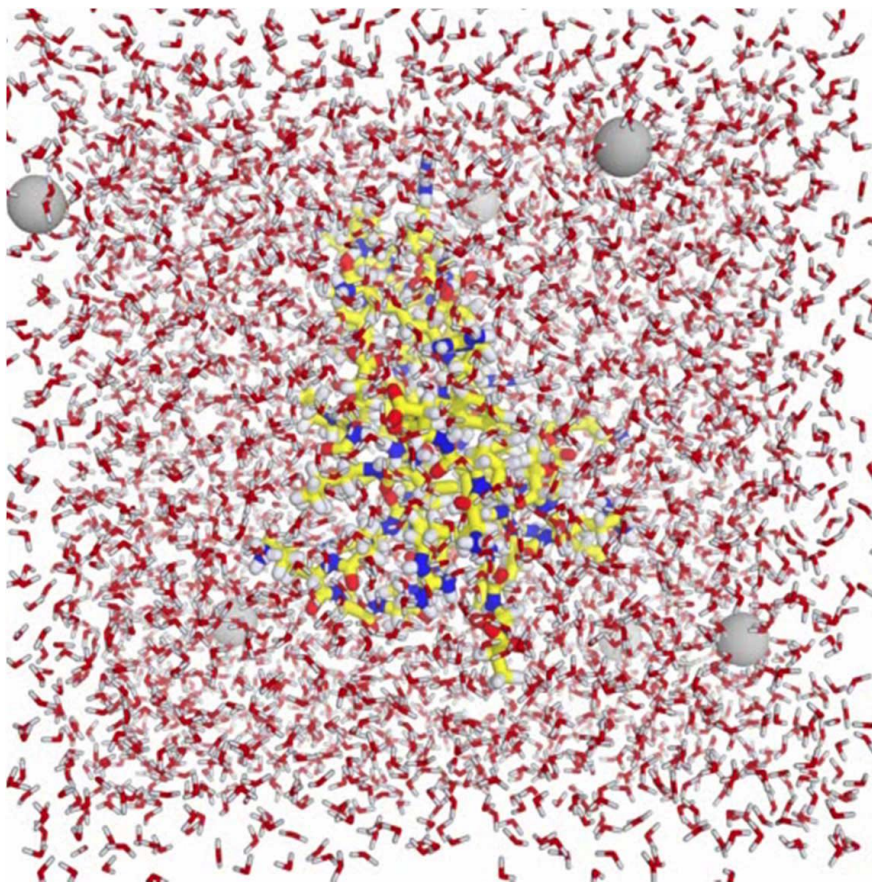
Peter Güntert · Sina Kazemi

| | |
|---|-----------|
| 1. Einführung | 2 |
| 1.1. Was ist MD Simulation? | 2 |
| 1.2. Meilensteine der MD Simulation | 2 |
| 2. Klassische Mechanik | 3 |
| 2.1. Bewegungsgleichung: Newton Gleichung | 3 |
| 2.2. Kräfte = –Gradient der potenziellen Energie | 4 |
| 2.3. Energien | 4 |
| 3. MD Algorithmen | 5 |
| 3.1. Integration der Bewegungsgleichungen: Leap-frog Algorithmus | 5 |
| 3.2. Erhaltungsgrößen: Energieerhaltung | 6 |
| 3.3. Genauigkeit der Energieerhaltung: maximaler Zeitschritt | 7 |
| 3.4. Feste Bindungslängen: SHAKE | 7 |
| 3.5. Temperatur: Berechnung, Stabilisierung | 7 |
| 3.6. Periodische Randbedingungen | 8 |
| 3.7. Druck: Berechnung über Virialsatz, Stabilisierung | 9 |
| 4. Kraftfelder | 10 |
| 4.1. Kovalente Struktur: Bindungslängen, -winkel, Torsionswinkel, Impropers | 10 |
| 4.2. Nichtbindende Wechselwirkung: van der Waals, elektrostatisch | 12 |
| 4.3. Rechenzeit | 12 |
| 4.4. Cutoff | 13 |
| 4.5. Lösungsmittel (Wasser): explizite, implizite Behandlung | 14 |
| 5. MD Techniken | 14 |
| 5.1. Energieminimierung | 14 |
| 5.2. Anfangsgeschwindigkeiten | 15 |
| 5.3. Äquilibration | 15 |
| 5.4. Simulated Annealing | 15 |
| 6. Literatur | 16 |
| 6.1. Lehrbücher | 16 |
| 6.2. Originalliteratur | 16 |

1. Einführung

1.1. Was ist MD Simulation?

Eine MD Simulation ist die numerische Berechnung der Zeitentwicklung eines molekularen Systems nach den Gesetzen der klassischen Mechanik. Das molekulare System besteht im Allgemeinen aus Atomen, die als Massenpunkte behandelt werden und sich unter dem Einfluss von Kräften bewegen, die aus einer potenziellen Energiefunktion abgeleitet werden. In der Biologie besteht das molekulare System typischerweise aus einem oder mehreren Makromolekülen (Protein, DNA, RNA; in der Abbildung gelb/blau/rot), die von kleinen Lösungsmittelmolekülen (Wasser; in der Abbildung grau/rot) umgeben sind. Möglicherweise sind auch Liganden, Ionen (graue Kugeln in der Abbildung) oder Bausteine von Membranen vorhanden.



Die gesamte Rechnung erfolgt klassisch ohne (direkte) Berücksichtigung von quantenmechanischen Effekten. Dies ist eine grobe Näherung, weil Vorgänge im atomaren und subatomaren Bereich eigentlich durch die Quantenmechanik beschrieben werden müssten. So lassen sich zum Beispiel mit Hilfe der klassischen MD Simulation chemische Reaktionen nicht adäquat behandeln oder gar vorhersagen.

1.2. Meilensteine der MD Simulation

Erste MD Simulationen wurden in den 1950er-Jahren durchgeführt, als erstmals geeignete Computer verfügbar wurden. Die MD Simulation hat sich seither parallel mit der Leistungsfähigkeit von Rechnern weiterentwickelt. Größere Speicher erlaubten die Behandlung von zunehmend größeren und komplexeren Systemen und schnellere Prozessoren ermöglichten Simulationen über längere Zeiten und mit realistischeren Energiefunktionen.

Einige Meilensteine in der Entwicklung der MD Simulation

| Jahr | System | Zeitbereich |
|------|---|-----------------------|
| 1957 | Harte Scheiben (2D) (Alder and Wainwright 1957) | |
| 1964 | Monoatomare Flüssigkeit (Rahman 1964) | 10^{-11} s |
| 1971 | Flüssiges Salz (Ionen) | 10^{-11} s |
| 1971 | Molekulare Flüssigkeit (Wasser) (Rahman and Stillinger 1971) | 10^{-11} s |
| 1975 | Einfaches kleines Polymer | 10^{-11} s |
| 1977 | Protein im Vakuum (McCammon et al. 1977) | 2×10^{-11} s |
| 1982 | Einfache Membran | 2×10^{-10} s |
| 1983 | Protein in Lösung | 2×10^{-11} s |
| 1986 | DNA in Lösung | 10^{-10} s |
| 1989 | Protein-DNA Komplex in Lösung | 10^{-10} s |
| 1991 | Protein-Protein Komplex in Lösung | 10^{-9} s |
| 1998 | 1 μ s Simulation eines Proteins in Lösung (Duan and Kollman 1998) | 10^{-6} s |
| 2013 | Nobelpreis für Martin Karplus, Michael Levitt, Arieh Warshel | |

Heute ist die Speichergröße für MD Simulationen keine wesentliche Einschränkung mehr. Hingegen ist die Simulationsdauer immer noch sehr viel kürzer als viele biologisch relevante Vorgänge, insbesondere die Proteinfaltung, und ist, neben der klassischen Näherung, nach wie vor die bedeutendste Einschränkung für MD Simulationen. In den letzten Jahren wurden Fortschritte erzielt, indem Teile der Kraftberechnungen auf hochgradig parallele Grafikprozessoren (GPUs) ausgelagert wurden. Allgemein ist jedoch zu sagen, dass MD Simulationen nicht beliebig parallelisiert werden können, weil sie immer in iterativen Zeitschritten ablaufen.

2. Klassische Mechanik

2.1. Bewegungsgleichung: Newton Gleichung

In der klassischen Mechanik wird der Zustand eines Systems, das aus N Teilchen (Massenpunkten) besteht, durch die Positionen $r_i(t)$ und Geschwindigkeiten $v_i(t)$ aller Massenpunkte $i = 1, \dots, N$ im Zeitpunkt t *vollständig* beschrieben, d.h. es lässt sich theoretisch im Allgemeinen aus der Kenntnis dieser Größen das gesamte Verhalten des Systems zu allen zukünftigen und früheren Zeiten berechnen.

Dies bedeutet nicht, dass man dafür eine einfache Formel angeben kann! Es bedeutet auch nicht, dass man in der Praxis die Zeitentwicklung eines beliebigen mechanischen Systems über längere Zeiten

stabil vorhersagen kann, denn mechanische Systeme neigen zu chaotischem Verhalten, indem kleinste Änderungen der Anfangspositionen oder -geschwindigkeiten im Laufe der Zeit zu beliebig großen Unterschieden führen können.

Die Geschwindigkeit ist die Ableitung des Ortsvektors nach der Zeit:

$$\mathbf{v}_i = \frac{d\mathbf{r}_i}{dt}$$

Die Beschleunigung ist die Ableitung der Geschwindigkeit nach der Zeit und damit die zweite Ableitung des Ortsvektors nach der Zeit:

$$\mathbf{a}_i = \frac{d\mathbf{v}_i}{dt} = \frac{d^2\mathbf{r}_i}{dt^2}$$

Isaac Newton hat erkannt, dass die Beschleunigung \mathbf{a}_i eines Teilchens durch die Masse m_i und die auf das Teilchen wirkende Kraft \mathbf{F}_i bestimmt wird:

$$m_i \mathbf{a}_i = \mathbf{F}_i$$

Dieses sogenannte Zweite Newtonsche Gesetz aus dem Jahr 1687 ist die Grundgleichung der klassischen Mechanik, wenn das System durch kartesische Koordinaten der Teilchen beschrieben wird. Es ist die theoretische Grundlage der MD Simulation, die sich damit befasst, die Newton-Gleichung auf molekulare Systeme anzuwenden und numerisch näherungsweise zu lösen, denn die Bewegungsgleichungen sind im Allgemeinen für Systeme mit mehr als zwei Teilchen nicht explizit lösbar. Obwohl die der MD Simulation zu Grunde liegende Theorie somit schon sehr lange bekannt ist, wurden MD Simulationen erst mit der Entwicklung von Computern, die die notwendigen numerischen Rechnungen in kurzer Zeit durchführen können, möglich.

2.2. Kräfte = –Gradient der potenziellen Energie

Die in der Newtonschen Bewegungsgleichung vorkommenden Kraftvektoren $\mathbf{F}_i(t)$ hängen von den physikalischen Wechselwirkungen zwischen den Teilchen ab, welche grundsätzlich experimentell bestimmt werden müssen. In den meisten Fällen, insbesondere in der MD Simulation, hängen die Kräfte nur von den Positionen $\mathbf{r}_i(t)$ aber nicht von den Geschwindigkeiten $\mathbf{v}_i(t)$ ab. Dann ist es möglich, die Kraftvektoren als negative Gradienten (Ableitungen nach den drei Ortskoordinaten x, y, z) einer skalaren Funktion $V(\mathbf{r}_1, \dots, \mathbf{r}_N)$, des Potentials oder der potenziellen Energie, zu erhalten:

$$\mathbf{F}_i = - \frac{\partial V(\mathbf{r}_1, \dots, \mathbf{r}_N)}{\partial \mathbf{r}_i}$$

Ist das Teilchen mehreren Wechselwirkungen unterworfen, z.B. indem es mit mehreren anderen Teilchen wechselwirkt, so berechnet sich die gesamte potenzielle Energie V des Teilchens als Summe der einzelnen Beiträge. Entsprechend ergibt sich der für die Newton-Gleichung (1) relevante Kraftvektor $\mathbf{F}_i(t)$ als Summe der einzelnen Kraftvektoren.

Für die MD Simulation ist die Wahl der potenziellen Energiefunktion V , die eine möglichst gute klassische Beschreibung des eigentlich quantenmechanischen Systems leisten muss, von größter Bedeutung.

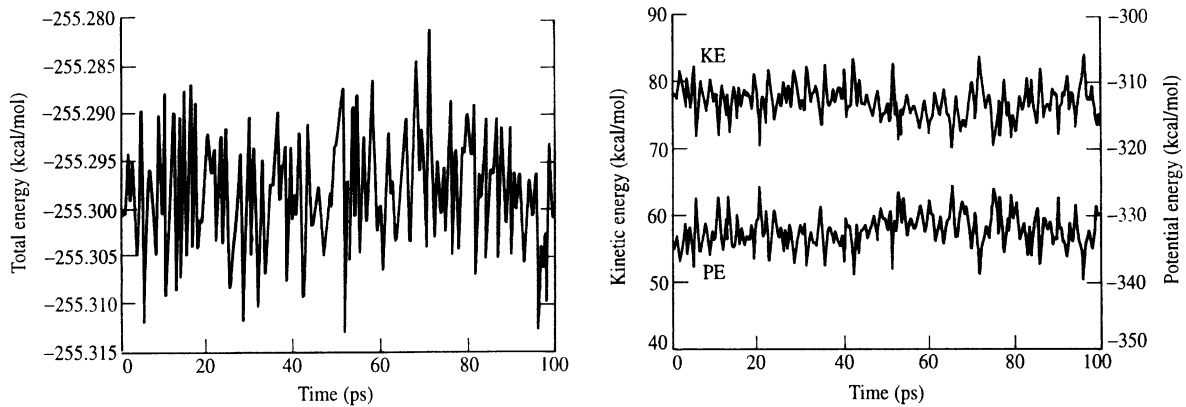
2.3. Energien

Zusätzlich zur oben genannten potenziellen Energie, $E_{\text{pot}} = V(\mathbf{r}_1, \dots, \mathbf{r}_N)$, die im Allgemeinen nur von der Position der Teilchen abhängt, gibt es in einem klassischen mechanischen System auch kinetische Energie, die von der Geschwindigkeit der Teilchen abhängt:

$$E_{\text{kin}} = \sum_{i=1}^N \frac{1}{2} m_i v_i^2$$

Die Summe der beiden Energien, die Gesamtenergie $E = E_{\text{pot}} + E_{\text{kin}}$, ist für ein abgeschlossenes (nicht mit der Umgebung wechselwirkendes) System zeitlich konstant, d.h. eine Erhaltungsgröße der Bewegung (siehe unten).

In der Abbildung kann man deutlich erkennen, dass in einer MD Simulation die Gesamtenergie (links) sehr viel kleinere Fluktuationen zeigt als die kinetische Energie oder die potenzielle Energie, welche keine Erhaltungsgrößen der Bewegung sind (man beachte die unterschiedlichen Energieskalen).



3. MD Algorithmen

In diesem Kapitel werden die grundlegenden Algorithmen für die MD Simulation eingeführt.

3.1. Integration der Bewegungsgleichungen: Leap-frog Algorithmus

Die Newtonsche Bewegungsgleichung ist eine Differentialgleichung 2. Ordnung. Es wurden verschiedene Methoden entwickelt, um eine näherungsweise Lösung zu bestimmen. Für die MD Simulation hat sich ein einfaches von L. Verlet 1967 in die MD Simulation eingeführtes Verfahren bewährt (Verlet 1967). Wir berechnen eine Näherung der Positionen \mathbf{r}_i zur Zeit $t + \Delta t$, wobei Δt ein kleiner, aber endlicher Zeitschritt ist, durch Taylor-Reihenentwicklung der Positionen zur Zeit t :

$$\begin{aligned} \mathbf{r}_i(t + \Delta t) &= \mathbf{r}_i(t) + \Delta t \frac{d\mathbf{r}_i(t)}{dt} + \frac{\Delta t^2}{2} \frac{d^2\mathbf{r}_i(t)}{dt^2} + \frac{\Delta t^3}{3!} \frac{d^3\mathbf{r}_i(t)}{dt^3} + O(\Delta t^4) \\ &= \mathbf{r}_i(t) + \Delta t \mathbf{v}_i(t) + \frac{\Delta t^2}{2} \mathbf{a}_i(t) + \frac{\Delta t^3}{3!} \frac{d^3\mathbf{r}_i(t)}{dt^3} + O(\Delta t^4) \end{aligned}$$

wobei $O(\Delta t^4)$ für die weiteren Terme der Ordnung Δt^4 und größer steht. Analog gilt für die Positionen zur früheren Zeit $t - \Delta t$,

$$\mathbf{r}_i(t - \Delta t) = \mathbf{r}_i(t) - \Delta t \mathbf{v}_i(t) + \frac{\Delta t^2}{2} \mathbf{a}_i(t) - \frac{\Delta t^3}{3!} \frac{d^3\mathbf{r}_i(t)}{dt^3} + O(\Delta t^4)$$

Durch Addieren dieser zwei Gleichungen und Verwenden der Newton-Gleichung erhalten wir

$$\mathbf{r}_i(t + \Delta t) + \mathbf{r}_i(t - \Delta t) = 2\mathbf{r}_i(t) + \Delta t^2 \mathbf{a}_i(t) + O(\Delta t^4) = 2\mathbf{r}_i(t) + \Delta t^2 \frac{\mathbf{F}_i(t)}{m_i} + O(\Delta t^4)$$

Somit können die Positionen zum neuen Zeitpunkt $t + \Delta t$ aus den Positionen zu den früheren Zeitpunkten t und $t - \Delta t$ berechnet werden:

$$\mathbf{r}_i(t + \Delta t) = 2\mathbf{r}_i(t) - \mathbf{r}_i(t - \Delta t) + \Delta t^2 \frac{\mathbf{F}_i(t)}{m_i} + O(\Delta t^4)$$

Interessant an diesem sogenannten Verlet-Algorithmus ist, dass die Terme der Ordnung Δt^3 wegfallen und deshalb nur kleinere Fehler der Ordnung Δt^4 auftreten.

Ein Nachteil des ursprünglichen Verlet-Algorithmus' ist, dass die Geschwindigkeiten gar nicht mehr vorkommen und deshalb durch wesentlich ungenauere numerische Ableitung berechnet werden müssen, um z.B. die kinetische Energie zu bestimmen. Aus diesem Grund wird in der Praxis häufig eine Variante des Verlet-Algorithmus', der leap-frog („Bocksprung“) Algorithmus verwendet, in dem die Positionen zu „ganzen“ Zeiten $t - \Delta t$, t , $t + \Delta t$ usw. und die Geschwindigkeiten zu „halben“ Zeiten $t - \Delta t/2$, $t + \Delta t/2$ usw. verwendet werden. Die Geschwindigkeiten und Positionen werden abwechselungsweise um einen Zeitschritt Δt weitergeführt:

$$\begin{aligned} \mathbf{r}_i(t) &= \mathbf{r}_i(t - \Delta t) + \Delta t \mathbf{v}_i\left(t - \frac{\Delta t}{2}\right) \\ \mathbf{v}_i\left(t + \frac{\Delta t}{2}\right) &= \mathbf{v}_i\left(t - \frac{\Delta t}{2}\right) + \Delta t \mathbf{a}_i(t) = \mathbf{v}_i\left(t - \frac{\Delta t}{2}\right) + \Delta t \frac{\mathbf{F}_i(t)}{m_i} \end{aligned}$$

3.2. Erhaltungsgrößen: Energieerhaltung

Gewisse Eigenschaften eines klassischen mechanischen Systems verändern sich im Laufe der Zeit nicht, selbst wenn das System unter dem Einfluss komplizierter Wechselwirkungen komplexe Bewegungen ausführt. Insbesondere kann man zeigen (Satz von Noether), dass zu jeder kontinuierlichen Symmetrie des Systems eine solche Erhaltungsgröße gehört.

| Invarianz bezüglich | Erhaltungsgröße |
|---------------------|---|
| Zeitverschiebung | Gesamtenergie $E = E_{\text{pot}} + E_{\text{kin}}$ |
| Translation | Gesamtimpuls $\mathbf{P} = \sum_{i=1}^N \mathbf{p}_i$, mit $\mathbf{p}_i = m_i \mathbf{v}_i$ |
| Rotation | Gesamtdrehimpuls $\mathbf{L} = \sum_{i=1}^N \mathbf{r}_i \times \mathbf{p}_i$ |

Um zu sehen, dass die Gesamtenergie für ein abgeschlossenes System, das nicht mit seiner Umgebung wechselwirkt, eine Erhaltungsgröße ist, betrachten wir die Ableitung der Gesamtenergie nach der Zeit:

$$\begin{aligned} \frac{dE}{dt} &= \frac{d}{dt} \sum_{i=1}^N \frac{1}{2} m_i \mathbf{v}_i^2 + \frac{dV(\mathbf{r}_1, \dots, \mathbf{r}_N)}{dt} = \sum_{i=1}^N \frac{1}{2} m_i 2\mathbf{v}_i \cdot \frac{d\mathbf{v}_i}{dt} + \sum_{i=1}^N \frac{\partial V(\mathbf{r}_1, \dots, \mathbf{r}_N)}{\partial \mathbf{r}_i} \cdot \frac{d\mathbf{r}_i}{dt} \\ &= \sum_{i=1}^N m_i \mathbf{v}_i \cdot \mathbf{a}_i - \mathbf{F}_i \cdot \mathbf{v}_i = \sum_{i=1}^N \mathbf{v}_i \cdot (m_i \mathbf{a}_i - \mathbf{F}_i) = 0 \end{aligned}$$

3.3. Genauigkeit der Energieerhaltung: maximaler Zeitschritt

In einer MD Simulation gilt die Energieerhaltung nur näherungsweise, weil die Bewegungsgleichungen durch den numerischen Algorithmus nicht exakt gelöst werden. Die Genauigkeit und Stabilität einer MD Simulation kann danach beurteilt werden, wie genau die Gesamtenergie erhalten bleibt.

Es hat sich gezeigt, dass MD Simulationen von Proteinen bei Laborbedingungen für Zeitschritte bis zu einer Länge von ca. 1 fs (bzw. 2 fs, wenn die Bindungslängen konstant gehalten werden; siehe unten), stabil durchgeführt werden können. 1 fs ist sehr kurz im Vergleich mit den Zeitskalen der meisten biologisch interessanten Vorgänge. D.h. es müssen sehr viele Zeitschritte durchgeführt werden, um biologisch relevante Vorgänge simulieren zu können.

3.4. Feste Bindungslängen: SHAKE

Ein Hauptproblem der MD Simulation ist, dass sehr kurze Zeitschritte im fs-Bereich (10^{-15} s) verwendet werden müssen, um die schnellsten Bewegungen erfassen zu können. Die schnellsten Bewegungen sind normalerweise Fluktuationen der Bindungslängen, insbesondere der kovalenten Bindungen der leichten Wasserstoffatome. Diese Fluktuationen der Bindungslängen können eliminiert werden, indem die entsprechenden Bindungslängen festgehalten werden. Bei MD Rechnungen mit kartesischen Atomkoordinaten geschieht dies normalerweise dadurch, dass zunächst ein „freier“ Zeitschritt gemacht und anschließend die neuen Positionen z.B. mit dem SHAKE Verfahren (Ryckaert et al. 1977) so korrigiert werden, dass die Bindungslängen exakt den Idealwerten entsprechen. Dadurch können etwa doppelt so lange Zeitschritte (2 fs statt 1 fs) verwendet werden. Entsprechend halbiert sich der Rechenaufwand.

3.5. Temperatur: Berechnung, Stabilisierung

In einer MD Simulation kann die momentane Temperatur T des Systems auf Grund des Äquipartitionssatzes aus der statistischen Mechanik, der besagt, dass im Gleichgewicht im Mittel jeder Freiheitsgrad eine kinetische Energie von $\frac{1}{2}k_B T$ hat, berechnet werden. Da jedes der N Atome drei Koordinaten und damit drei Freiheitsgrade hat, ergibt sich $E_{\text{kin}} = 3N \frac{1}{2}k_B T$ und somit

$$T = \frac{2E_{\text{kin}}}{3Nk_B}$$

Experimente im Labor werden oft bei konstanter Temperatur durchgeführt. Um die Temperatur in einer MD Simulation stabil zu halten oder an einen neuen Wert anzupassen, kann ein einfaches Verfahren, das der schwachen Kopplung des Systems an ein Wärmebad mit einer vorgegebenen Referenztemperatur T_{ref} entspricht, angewandt werden (Berendsen et al. 1984). Das Ziel dabei ist, die Temperatur langsam an den Referenzwert zu führen. Dadurch werden Störungen des Systems, wie sie bei einem „harten“ Eingriff, z.B. bei einer schlagartigen größeren Temperaturänderung, auftreten würden, vermieden. Wir möchten deshalb erreichen, dass sich die Temperatur zur Zeit t , $T(t)$, ausgehend von ihrem Wert $T(0)$ zur Zeit $t = 0$ über einer vorgegebene Zeitdauer τ an den Referenzwert T_{ref} annähert

$$T(t) = T_{\text{ref}} + (T(0) - T_{\text{ref}}) e^{-\frac{t}{\tau}}$$

Für die Ableitung der Temperatur nach der Zeit ergibt sich

$$\frac{dT}{dt} = (T(0) - T_{\text{ref}}) e^{-\frac{t}{\tau}} = \frac{T_{\text{ref}} - T(t)}{\tau}$$

Da die kinetische Energie von den Geschwindigkeiten abhängt, kann die Temperatur durch Skalieren der Geschwindigkeiten, $\mathbf{v}_i^{\text{neu}} = \lambda \mathbf{v}_i$, in jedem Zeitschritt angepasst werden. Um abrupte

Änderungen zu vermeiden, sollte der Skalierungsfaktor λ nahe bei 1 sein. Die entsprechende Änderung der kinetischen Energie ist

$$\Delta E_{\text{kin}} = \sum_{i=1}^N \frac{1}{2} m_i (\mathbf{v}_i^{\text{neu}})^2 - \sum_{i=1}^N \frac{1}{2} m_i \mathbf{v}_i^2 = (\lambda^2 - 1) \sum_{i=1}^N \frac{1}{2} m_i \mathbf{v}_i^2 = (\lambda^2 - 1) E_{\text{kin}}$$

Die zugehörige Temperaturänderung ist

$$\Delta T = (\lambda^2 - 1) T$$

und wir erhalten

$$\frac{T_{\text{ref}} - T(t)}{\tau} = \frac{dT}{dt} \approx \frac{\Delta T}{\Delta t} = (\lambda^2 - 1) \frac{T(t)}{\Delta t}$$

Auflösen nach λ liefert den gesuchten Skalierungsfaktor

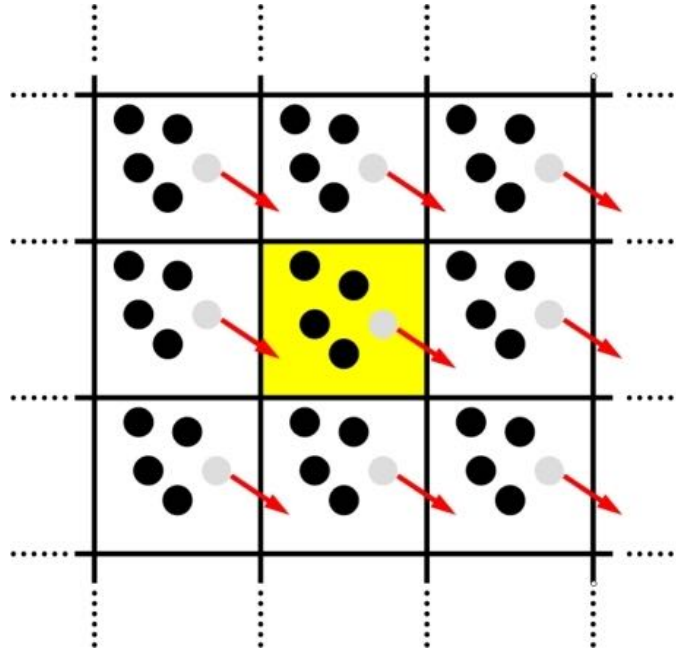
$$\lambda = \sqrt{1 + \frac{\Delta t}{\tau} \frac{T_{\text{ref}} - T(t)}{T(t)}}$$

In jedem Zeitschritt werden alle Geschwindigkeiten mit λ multipliziert. Ist die momentane Temperatur zu hoch, wird $\lambda < 1$. Ist sie zu tief, wird $\lambda > 1$. Je länger die charakteristische Zeit τ für die Temperaturanpassung gewählt wird, umso näher bleibt λ bei 1, umso weniger wird ins System eingegriffen und umso länger dauert die Temperaturanpassung.

3.6. Periodische Randbedingungen

Im Gegensatz zu einem echten makroskopischen System, das sehr viele Moleküle enthält, sollte das System in einer MD Simulation möglichst klein sein, um den Rechenaufwand zu minimieren. Man rechnet deshalb fast immer mit einer einzigen Kopie des Makromoleküls, die von gerade genügend Lösungsmolekülen (z.B. Wasser) umgeben ist, um grobe Artefakte zu vermeiden. Trotzdem gibt es in einer MD Simulation eines Proteins in explizitem Wasser im Allgemeinen wesentlich mehr Wasser- als Proteinatome. Um den Druck zu kontrollieren und zu vermeiden, dass die Wassermoleküle davonfliegen, müsste das System eigentlich in Wände eingeschlossen werden. Während die Wände in einem echten System (z.B. einer Proteinlösung in einem Glasröhrchen) nur marginalen Einfluss haben, wäre deren Effekt auf die kleine „Simulationsbox“ in einer MD Simulation stark und führte zu großen Artefakten.

Aus diesem Grund begrenzt man die Systemgröße auf eine andere Art, mit periodischen Randbedingungen. Dabei wird das System von z.B. quaderförmigen Grenzen umgeben und Atome, die die Grenzen verlassen, treten sofort mit der gleichen Geschwindigkeit auf der gegenüberliegenden Seite wieder ins System ein. Man kann sich vorstellen, dass das System (in der Abbildung gelb) von unendlich vielen identischen Kopien umgeben ist:



Um dies zu implementieren, müssen einerseits die Ortskoordinaten $\mathbf{r}_i = (x_i, y_i, z_i)$ von Atomen, die das System verlassen, zurückgesetzt werden. Angenommen, dass die Simulationsbox ein am Koordinatenursprung zentrierter Quader mit den Kantenlängen L_x, L_y, L_z ist, kann das erreicht werden, indem man die Koordinaten x_i durch $x_i - L_x \text{nint}(x_i/L_x)$ und analog für y_i und z_i ersetzt. Die Funktion $\text{nint}(x)$ liefert für ein reelles Argument x die nächstgelegene ganze Zahl. Andererseits müssen alle Abstandsvektoren $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$, die für die Berechnung der potenziellen Energie benötigt werden, zur nächstgelegenen Kopie des betreffenden Atoms auf dem periodischen Gitter berechnet werden (sogenannte *minimal image condition*). Dies kann wiederum dadurch erreicht werden, dass man die Koordinaten x_{ij} durch $x_{ij} - L_x \text{nint}(x_{ij}/L_x)$ und analog für y_{ij} und z_{ij} ersetzt.

3.7. Druck: Berechnung über Virialsatz, Stabilisierung

Der Druck wird in der Thermodynamik normalerweise durch die (gemittelte) Kraft auf die Wände des Systems definiert. Diese Kräfte entstehen dadurch, dass Moleküle gegen die Wände stoßen und daran reflektiert werden. Im Fall von periodischen Randbedingungen kann der Druck nicht auf diese Art berechnet werden, da die Moleküle die Grenzen des Systems ohne Wechselwirkung durchdringen können. Der Druck wird deshalb im Allgemeinen in MD Simulationen über den Virialsatz berechnet.

$$P = \frac{2}{3V} \left(E_{\text{kin}} + \frac{1}{2} \sum_{i=1}^{n-1} \sum_{j=i+1}^n (\mathbf{r}_i - \mathbf{r}_j) \cdot \mathbf{F}_{ij} \right)$$

P bezeichnet den (momentanen) Druck, V das Volumen des Systems, E_{kin} die kinetische Energie, und \mathbf{F}_{ij} die vom Atom j auf das Atom i ausgeübte Kraft. In einer MD Simulation sind die Fluktuationen des momentanen Drucks im Vergleich mit Fluktuationen der momentanen Temperatur im Allgemeinen sehr groß. Sie können mehrere Größenordnungen größer als der Mittelwert sein.

Um den Druck zu stabilisieren, kann ein analoges Verfahren wie für die Temperaturstabilisierung (siehe oben) angewandt werden. Im Fall des Drucks werden an Stelle der Geschwindigkeiten die Positionen (und damit die Systemgröße) skaliert, $\mathbf{r}_i^{\text{neu}} = \mu \mathbf{r}_i$ und $V^{\text{neu}} = \mu^3 V$. Betrachten wir ein ideales Gas (das Ergebnis gilt aber auch im allgemeinen Fall), so ist $PV = Nk_B T$ und die Druckänderung auf Grund der Skalierung ist

$$\Delta P = \frac{Nk_B T}{V^{\text{neu}}} - \frac{Nk_B T}{V} = \left(\frac{1}{\mu^3} - 1\right) \frac{Nk_B T}{V} = \left(\frac{1}{\mu^3} - 1\right) P$$

Analog wie bei der Temperaturanpassung ergibt sich

$$\frac{P_{\text{ref}} - P(t)}{\tau_P} = \frac{dP}{dt} \approx \frac{\Delta P}{\Delta t} = \left(\frac{1}{\mu^3} - 1\right) \frac{P(t)}{\Delta t}$$

Auflösen nach μ liefert den gesuchten Skalierungsfaktor

$$\mu = \sqrt[3]{1 - \frac{\Delta t}{\tau_P} \frac{P_{\text{ref}} - P(t)}{P(t)}}$$

In jedem Zeitschritt werden alle Ortsvektoren mit μ multipliziert. Ist der momentane Druck zu hoch, wird $\mu > 1$ und das System dehnt sich aus. Ist er zu tief, wird $\mu < 1$ und das System verkleinert sich.

4. Kraftfelder

Die potenzielle Energiefunktion wird in der MD Simulation traditionell als Kraftfeld (*force field*) bezeichnet, weil daraus die im System wirkenden Kräfte abgeleitet werden. Die Terme der Energiefunktion sollten so gewählt werden, dass sie einerseits das System möglichst realistisch in klassischer Näherung beschreiben und andererseits mit möglichst kleinem Rechenaufwand berechenbar sind.

4.1. Kovalente Struktur: Bindungslängen, -winkel, Torsionswinkel, Improper

In einer normalen MD Simulation wird jedes Atom als Massenpunkt mit drei Freiheitsgraden behandelt. Um sicherzustellen, dass die Geometrie der kovalenten Struktur des Proteins erhalten bleibt, müssen Potentiale angelegt werden, die die Bindungslängen und Bindungswinkel nahe bei ihren Idealwerten halten.

Außerdem werden im Allgemeinen Torsionswinkelpotentiale angelegt, die bei den bevorzugten Stellungen der Torsionswinkel Minima haben.

Das Potenzial für Bindungslängen ist üblicherweise quadratisch in der Abweichung vom Ideal- oder Gleichgewichtswert:

$$E_{\text{bond}} = \sum_{\text{bonds}} k_{\text{bond}} (r - r_{\text{eq}})^2$$

Die Summe läuft über alle kovalenten Bindungen, k_{bond} ist die Kraftkonstante, r die aktuelle und r_{eq} die ideale Länge der betreffenden Bindung. Das Potenzial ist harmonisch, d.h. eine Bindung wird wie eine Feder behandelt und kann nicht gebrochen werden. Es können keine chemischen Reaktionen behandelt werden. Der Rechenaufwand ist proportional zur Molekülgröße.

Die Größen k_{bond} und r_{eq} hängen vom Typ der kovalenten Bindung ab und werden in einer Energiefunktion normalerweise über die Atomtypen der beiden kovalent gebundenen Atome definiert. Z.B. gilt für das AMBER Kraftfeld (erste Zahl: k_{bond} in kcal/mol/Å²; zweite Zahl r_{eq} in Å):

| | | | | | | | | |
|-------|--------|--------|-------|--------|--------|-------|--------|--------|
| BONDS | | | CB-N* | 436.00 | 1.3740 | OH-P | 230.00 | 1.6100 |
| OW-HW | 553.00 | 0.9572 | CB-NB | 414.00 | 1.3910 | OS-P | 230.00 | 1.6100 |
| HW-HW | 553.00 | 1.5139 | CB-NC | 461.00 | 1.3540 | C*-HC | 367.00 | 1.0800 |
| OT-HW | 553.00 | 0.9572 | CK-H5 | 367.00 | 1.0800 | C -N | 490.00 | 1.3350 |
| OU-HU | 553.00 | 0.9572 | CK-N* | 440.00 | 1.3710 | C*-CB | 388.00 | 1.4590 |
| HU-HU | 553.00 | 1.5139 | CK-NB | 529.00 | 1.3040 | C*-CT | 317.00 | 1.4950 |
| OU-MU | 553.00 | 0.1500 | CM-CM | 549.00 | 1.3500 | C*-CW | 546.00 | 1.3520 |
| HU-MU | 553.00 | 0.8735 | CM-CT | 317.00 | 1.5100 | CA-CN | 469.00 | 1.4000 |
| OV-HV | 553.00 | 1.0000 | CM-HA | 367.00 | 1.0800 | CB-CN | 447.00 | 1.4190 |
| HV-HV | 553.00 | 1.6330 | CM-H4 | 367.00 | 1.0800 | CC-CT | 317.00 | 1.5040 |
| C -CA | 469.00 | 1.4090 | CM-H5 | 367.00 | 1.0800 | CC-CV | 512.00 | 1.3750 |
| C -CB | 447.00 | 1.4190 | CM-N* | 448.00 | 1.3650 | CC-CW | 518.00 | 1.3710 |
| C -CM | 410.00 | 1.4440 | CQ-H5 | 367.00 | 1.0800 | CC-NA | 422.00 | 1.3850 |
| C -CT | 317.00 | 1.5220 | CQ-NC | 502.00 | 1.3240 | CC-NB | 410.00 | 1.3940 |
| C -N* | 424.00 | 1.3830 | CT-CT | 310.00 | 1.5260 | CN-NA | 428.00 | 1.3800 |
| C -NA | 418.00 | 1.3880 | CT-HC | 340.00 | 1.0900 | CR-H5 | 367.00 | 1.0800 |
| C -NC | 457.00 | 1.3580 | CT-H1 | 340.00 | 1.0900 | CR-NA | 477.00 | 1.3430 |
| C -O | 570.00 | 1.2290 | CT-H2 | 340.00 | 1.0900 | CR-NB | 488.00 | 1.3350 |
| C -O2 | 656.00 | 1.2500 | CT-H3 | 340.00 | 1.0900 | CT-N | 337.00 | 1.4490 |
| C -OH | 450.00 | 1.3640 | CT-HP | 340.00 | 1.0900 | CT-N3 | 367.00 | 1.4710 |
| CA-CA | 469.00 | 1.4000 | CT-N* | 337.00 | 1.4750 | CT-S | 227.00 | 1.8100 |
| CA-CB | 469.00 | 1.4040 | CT-N2 | 337.00 | 1.4630 | CT-SH | 237.00 | 1.8100 |
| CA-CM | 427.00 | 1.4330 | CT-OH | 320.00 | 1.4100 | CV-H4 | 367.00 | 1.0800 |
| CA-CT | 317.00 | 1.5100 | CT-OS | 320.00 | 1.4100 | CV-NB | 410.00 | 1.3940 |
| CA-HA | 367.00 | 1.0800 | H -N2 | 434.00 | 1.0100 | CW-H4 | 367.00 | 1.0800 |
| CA-H4 | 367.00 | 1.0800 | H -N* | 434.00 | 1.0100 | CW-NA | 427.00 | 1.3810 |
| CA-N2 | 481.00 | 1.3400 | H -NA | 434.00 | 1.0100 | H -N | 434.00 | 1.0100 |
| CA-NA | 427.00 | 1.3810 | HO-OH | 553.00 | 0.9600 | H -N3 | 434.00 | 1.0100 |
| CA-NC | 483.00 | 1.3390 | HO-OS | 553.00 | 0.9600 | HS-SH | 274.00 | 1.3360 |
| CB-CB | 520.00 | 1.3700 | O2-P | 525.00 | 1.4800 | S -S | 166.00 | 2.0380 |

Analog wird für Bindungswinkel üblicherweise ein Potenzial verwendet, das quadratisch mit der Abweichung vom Ideal- oder Gleichgewichtswert ansteigt:

$$E_{\text{angle}} = \sum_{\text{angles}} k_{\text{angle}} (\tau - \tau_{\text{eq}})^2$$

Die Summe läuft über alle Bindungswinkel, d.h. über alle Tripel von Atomen, die durch zwei kovalente Bindungen verbunden sind, k_{angle} ist die Kraftkonstante, τ der aktuelle und τ_{eq} der ideale Wert des betreffenden Bindungswinkels. Der Rechenaufwand ist proportional zur Molekülgröße.

Für die Dieder- oder Torsionswinkel wird ein Potenzial verwendet, das die energetisch günstigen Stellungen der Torsionswinkel bevorzugt. Z.B. im AMBER Kraftfeld:

$$E_{\text{tor}} = \sum_{\text{torsions}} k_{\text{tor}} \frac{1 + \cos(n\theta - \gamma)}{2}$$

Die Summe läuft über alle Torsionswinkel, d.h. über alle Quadrupel von Atomen, die durch drei aufeinanderfolgende kovalente Bindungen verbunden sind, k_{tor} ist die Kraftkonstante, θ der aktuelle Wert des Torsionswinkels, n die Anzahl der Minima und γ die Position des ersten Energiemaximums des betreffenden Torsionswinkels.

Zusätzlich zu den eigentlichen Torsionswinkeln werden oft noch Potenziale für sogenannt uneigentliche Torsionswinkel („improper“) verwendet, die dazu dienen aromatische Ringe und Peptidgruppen planar zu halten. Diese Potenziale werden wie die Torsionswinkelpotenzial für 4 Atome definiert, die aber nicht aufeinanderfolgend kovalent gebunden sind. Im AMBER Kraftfeld hat das Potenzial die gleiche Form wie das Diederwinkelpotenzial:

$$E_{\text{improper}} = \sum_{\text{torsions}} k_{\text{improper}} \frac{1 + \cos(n\theta - \gamma)}{2}$$

In anderen Kraftfeldern, z.B. CHARMM, hat das Potenzial die Form

$$E_{\text{improper}} = \sum_{\text{improvers}} k_{\text{improper}} (\theta - \theta_{\text{eq}})^2$$

4.2. Nichtbindende Wechselwirkung: van der Waals, elektrostatisch

Zwischen allen Atompaaaren, die nicht direkt oder über zwei kovalente Bindungen verbunden sind, wird die nichtbindende Wechselwirkung berechnet, die im Allgemeinen aus einem Lennard-Jones Potenzial, das die sterische Abstoßung und eine schwache van der Waals Anziehung zwischen ungeladenen Atomen beschreibt, und dem elektrostatischen Potenzial, das die Anziehung oder Abstoßung geladener Atome beschreibt, besteht. Das Lennard-Jones Potenzial für zwei Atome im Abstand r hat die Form

$$E_{\text{vdW}} = \frac{A}{r^{12}} - \frac{B}{r^6}$$

Die beiden Konstanten A und B hängen von den involvierten Atomtypen ab.

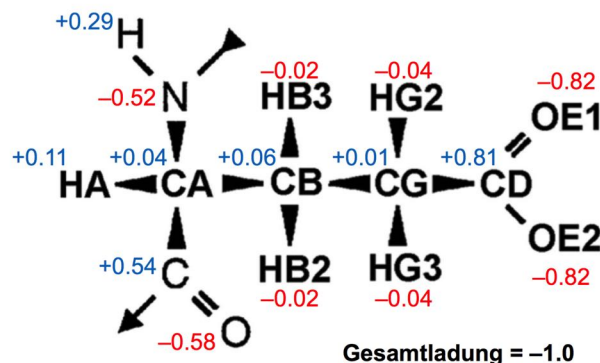
Das elektrostatische Potenzial für zwei Atome mit Partialladungen q_1 und q_2 im Abstand r hat die Form

$$E_{\text{el}} = \frac{1}{4\pi\epsilon_0} \frac{q_1 q_2}{\epsilon r} = 332 \frac{q_1 q_2}{\epsilon r}$$

Die Dielektrizitätskonstante im Vakuum ϵ_0 hat den Wert 8.854×10^{-12} As/Vm. Die Konstante 332 gilt, wenn die für MD Simulationen üblichen Einheiten verwendet werden: Energie in kcal/mol, Partialladungen in Elementarladungen, Distanzen in Å. ϵ bezeichnet die Dielektrizitätskonstante des Mediums. Wenn das Lösungsmittel explizit behandelt wird, ist $\epsilon = 1$.

Das elektrostatische Potenzial wird nicht nur für explizit geladene Atome berechnet, sondern es wird jedem Atom eine Partialladung zugeschrieben, die z.B. die Polarisierung einer kovalenten Bindung beschreibt. Auf diese Art können auch Wasserstoffbrückenwechselwirkungen über das elektrostatische Potenzial modelliert werden.

Als Beispiel sind in der Abbildung die Partialladungen für einen Glu Aminosäurerest angegeben.



4.3. Rechenzeit

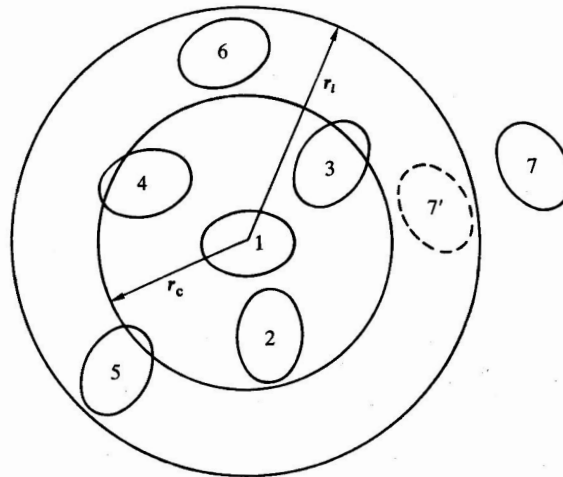
Der Rechenaufwand für die Berechnung der mit der kovalenten Struktur verbundenen, „bindenden“ Wechselwirkungen wächst linear mit der Größe des Moleküls, wogegen die nichtbindenden Wechselwirkungen grundsätzlich über fast alle $N(N - 1)/2$ Atompaaare laufen. Der Rechenaufwand für die nichtbindenden Wechselwirkungen steigt daher quadratisch mit der Anzahl Atome an. Für ein Makromolekül (das zusätzlich noch von vielen Lösungsmittelmolekülen umgeben ist) ist die Berechnung der nichtbindenden Wechselwirkungen der bei weitem aufwändigste Teil einer MD Simulation.

4.4. Cutoff

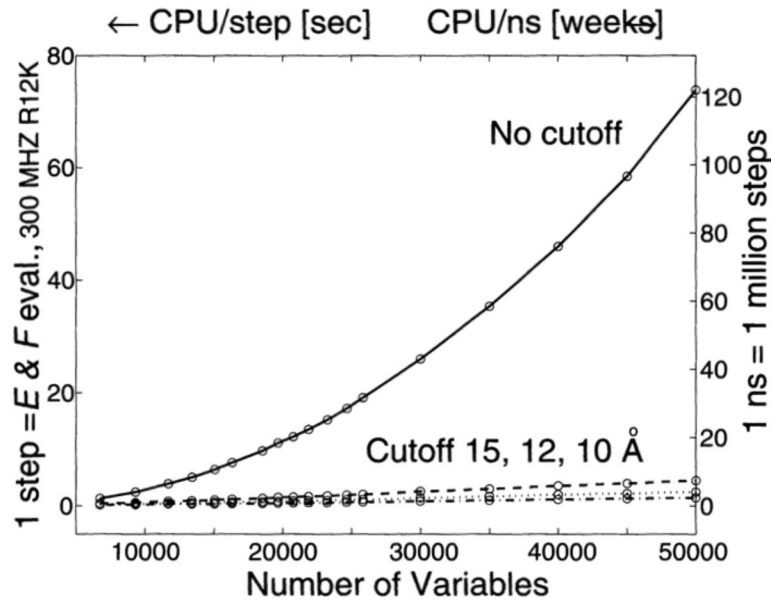
Der Aufwand für die Berechnung der nichtbindenden Wechselwirkungen kann reduziert werden, wenn man die Wechselwirkung zwischen Atomen, die weit voneinander entfernt sind, vernachlässigt. Um festzustellen, welche Atome weit voneinander entfernt sind, muss allerdings der Abstand aller Atompaare berechnet werden, was wiederum einen quadratisch zunehmenden Rechenaufwand benötigt. Darum führt man diese Berechnung einmal durch und speichert die Atompaare, deren Abstand unter einem gewissen Grenzwert liegt, in einer Paarliste ab. In den folgenden Zeitschritten werden die nichtbindenden Wechselwirkungen nur noch für die Atompaare in der Paarliste berechnet. Nach einer festgelegten Anzahl von Zeitschritten oder wenn sich ein Molekül eine gewisse Distanz von seiner Ursprungsposition weg bewegt hat, wird die Paarliste erneuert. Dadurch fällt dieser aufwändige Schritt nur z.B. alle 50–100 Zeitschritte an.

Wie die Abbildung zeigt, wird die Distanzgrenze r_1 für den Eintrag in die Paarliste etwas größer gewählt als der eigentliche Cutoff r_c , die maximale Distanz, bis zu der die nichtbindenden Wechselwirkungen berechnet werden. Dadurch bleibt die Paarliste länger aktuell, wenn die Moleküle sich bewegen.

In der Abbildung werden für das Molekül 1 die Wechselwirkungen zu den Molekülen 2–6 in die Paarliste eingetragen, aber zunächst nur für die Moleküle 2–4, die näher als die Cutoffdistanz r_c liegen, berechnet. Falls die Moleküle 5–6 im Lauf der MD Simulation näher als die Cutoffdistanz kommen, werden ihre Wechselwirkungen mit dem Molekül 1 berechnet. Anders für das Molekül 7, das am Anfang nicht in die Paarliste kommt, weil es zu weit entfernt ist. Seine Wechselwirkungen mit Molekül 1 werden auch dann nicht berücksichtigt, wenn es näher als die Cutoffdistanz kommt, sondern erst, wenn die Paarliste neu erstellt wird.



Die nichtbindenden Wechselwirkungen nehmen mit dem Abstand r der involvierten Atome ab. Die van der Waals Wechselwirkung geht mit r^{-6} und damit sehr rasch zurück. Im Gegensatz dazu sinkt die elektrostatische Wechselwirkung nur mit r^{-1} und bleibt auch für längere Distanzen noch bedeutsam. D.h. dass man für realistische MD Simulationen elektrostatische Wechselwirkungen bis zu Abständen von ca. 12 Å berechnen muss. Da Proteine zusammen mit der umgebenden Wasserbox wesentlich größer als 12 Å sind, ergibt sich dennoch durch den Cutoff eine große Rechenzeiterparnis.



4.5. Lösungsmittel (Wasser): explizite, implizite Behandlung

Die Wechselwirkungen des Lösungsmittels (normalerweise Wasser) mit biologischen Makromolekülen sind wichtig für eine realistische Beschreibung ihres Verhaltens. MD Simulationen, in denen das Lösungsmittel nicht behandelt wird („Simulationen im Vakuum“) führen zu starken Artefakten und sind nicht empfehlenswert. Wasser bildet sehr viele Wasserstoffbrücken aus und hat eine hohe Dielektrizitätskonstante (ca. 80; im Gegensatz zu ca. 2 im Proteininnern), welche zu einer Abschwächung der elektrostatischen Wechselwirkungen führt.

Grundsätzlich kann das Lösungsmittel auf zwei unterschiedliche Arten behandelt werden: Entweder, indem das Protein explizit von z.B. Wassermolekülen umgeben und die MD Simulation für das gesamte Wasser-Protein System durchgeführt wird, oder durch implizite Wassermodelle, in denen das Wasser durch ein Kontinuum approximiert wird. Letztere können Rechenzeit einsparen, sind aber im Allgemeinen weniger genau als die explizite Behandlung des Wassers.

5. MD Techniken

5.1. Energieminimierung

MD Simulationen starten von einer Ausgangsstruktur. Wenn diese energetisch sehr ungünstig ist, z.B. weil Atome überlappen, bedeutet das, dass die MD Simulation bei einer sehr hohen potenziellen Energie startet, die dann zu einem großen Teil in kinetische Energie und damit schnelle Bewegungen überführt wird, die die Startstruktur (zu) stark verändern oder die MD Rechnung zum Absturz bringen können. Deshalb ist es sinnvoll, zunächst die Energie der Startstruktur mit anderen Algorithmen zu minimieren, bevor die eigentliche MD Simulation gestartet wird.

Dazu werden lokale Minimierungsalgorithmen eingesetzt, die im Wesentlichen durch den Gradienten der Energiefunktion gesteuert in jedem Schritt versuchen, eine Struktur mit tieferer Energie zu finden. Diese Verfahren können besser mit hohen Energien umgehen als eine MD Simulation. Aber sie haben einen kleinen Konvergenzradius, d.h. sie bleiben leicht in einem lokalen Minimum, das weit vom globalen Minimum entfernt ist, stecken. Das ist in diesem Zusammenhang jedoch kein ernsthaftes

Problem, da während der anschließenden MD Simulation solche lokalen Minima wieder verlassen werden können.

5.2. Anfangsgeschwindigkeiten

Die Anfangsgeschwindigkeiten sollten so gesetzt werden, dass die kinetische Energie der gewünschten Temperatur entspricht. Für die Geschwindigkeitsverteilung kann die Maxwell-Boltzmann-Verteilung gewählt werden, die die Geschwindigkeiten der Teilchen eines idealen Gases im thermodynamischen Gleichgewicht beschreibt. Im Allgemeinen stellt sich die für das System charakteristische Gleichgewichtsverteilung der Geschwindigkeit während der Äquilibrationsphase rasch ein.

5.3. Äquilibration

Auch nach einer lokalen Energieminimierung ist ein Protein-Lösungsmittelsystem im Allgemeinen noch weit von einem stabilen Gleichgewichtszustand entfernt. Deshalb wird die MD Simulation mit einer Äquilibrationsphase gestartet, bevor die Trajektorie aufgenommen wird, die am Ende analysiert wird. Die Dauer der Äquilibration sollte so gewählt werden, dass Temperatur und Druck den gewünschten Wert erreicht haben und sich die potenzielle Energie im Mittel nicht mehr signifikant ändert. Besonders am Anfang der Äquilibration ist es wichtig, zu große Änderungen der Proteinstruktur zu vermeiden. Dies kann durch Positionseinschränkungen erfolgen, die jedes Proteinatom mit einem quadratischen Potenzial („einer Feder“) an seine Ausgangsposition ziehen, während sich die Wassermoleküle frei bewegen können.

5.4. Simulated Annealing

MD Simulationen können für unterschiedliche Zwecke eingesetzt werden. Einerseits kann das Ziel sein, die Dynamik eines Systems so realistisch wie möglich zu simulieren. In diesem Fall sollte die MD Simulation unter möglichst gleichen Bedingungen (Temperatur, Druck usw.) durchgeführt werden, wie das Experiment.

Eine andere wichtige Anwendung von MD Simulationen ist es, Proteinkonformationen zu finden, die energetisch möglichst günstig sind. In diesem Fall ist der genaue Ablauf der Bewegung (die Trajektorie) nicht wichtig, da nur die Endstruktur oder die Struktur mit tiefster Energie interessiert. Die MD Simulation dient als Minimierungsalgorithmus, der gegenüber einfachen Minimierungsalgorithmen den Vorteil hat, dass lokale Maxima der potenziellen Energie überwunden werden können und der Algorithmus dadurch eine bessere Chance hat, das globale Minimum (oder Strukturen in dessen Nähe) zu finden. Über die Temperatur kann gesteuert werden, wie viel kinetische Energie im System ist, und damit, welche Höhe von Energiebarrieren noch überwindbar ist, weil zwischen kinetischer und potenzieller Energie ausgetauscht werden kann, aber die (konstante) Gesamtenergie eine absolute obere Schranke für die potenzielle Energie darstellt.

Simulated annealing ist eine effiziente Methode, dies zu erreichen, in dem zunächst eine MD Simulation bei hoher Temperatur gestartet wird, so dass auch höhere Energiebarrieren überwunden und der Konformationsraum rasch durchlaufen werden kann. Anschließend wird die Temperatur langsam abgesenkt, um den lokalen Konformationsraum genauer abzusuchen.

6. Literatur

6.1. Lehrbücher

- Allen MP, Tildesley DJ (1987) Computer simulation of liquids. Clarendon Press, Oxford
- Leach AR (2001) Molecular Modelling: Principles and Applications. Pearson, Harlow, UK
- Rapaport DC (2004) The Art of Molecular Dynamics Simulation. Cambridge University Press
- Schlick T (2010) Molecular Modeling and Simulation: An Interdisciplinary Guide. Springer, New York.
- Tuckerman ME (2010) Statistical Mechanics: Theory and Molecular Simulation. Oxford University Press

6.2. Originalliteratur

- Alder BJ, Wainwright TE (1957) Phase transition for a hard sphere system. J Chem Phys 27:1208-1209
- Berendsen HJC, Postma JPM, van Gunsteren WF, DiNola A, Haak JR (1984) Molecular dynamics with coupling to an external bath. J Chem Phys 81:3684-3690
- Duan Y, Kollman PA (1998) Pathways to a protein folding intermediate observed in a 1-microsecond simulation in aqueous solution. Science 282:740-744
- McCammion JA, Gelin BR, Karplus M (1977) Dynamics of folded proteins. Nature 267:585-590
- Rahman A (1964) Correlations in motion of atoms in liquid argon. Phys Rev 136:A405-A411
- Rahman A, Stillinger FH (1971) Molecular dynamics study of liquid water. J Chem Phys 55:3336-3359
- Ryckaert JP, Ciccotti G, Berendsen HJC (1977) Numerical integration of the Cartesian equations of motion of a system with constraints: molecular dynamics of *n*-alkanes. J Comput Phys 23:327-341
- Verlet L (1967) Computer "experiments" on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules. Phys Rev 159:98-103