

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
імені В. Н. КАРАЗІНА

**МЕТОД ДИНАМІКИ ЧАСТИНОК
В МАТЕМАТИЧНОМУ МОДЕЛЮВАННІ
ДИНАМІЧНИХ СИСТЕМ**

Навчально-методичний посібник
для студентів другого курсу другого (магістерського) рівня
вищої освіти зі спеціальності «Прикладна математика»

Електронне видання

Харків – 2023

УДК: 519.6+ 51-7
М 54

Рецензенти:

Ю. В. Ромашов – д.тех.н., професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки Харківського національного технічного університету радіоелектроніки;

О. М. Дацюк – канд.тех.н., доцент кафедри біомедичної інженерії Харківського національного технічного університету радіоелектроніки.

*Затверджено до розміщення в мережі Інтернет рішенням Науково-методичної ради
Харківського національного університету імені В. Н. Каразіна
(протокол № 8 від 16 травня 2023 року)*

М 54 **Метод динаміки частинок в математичному моделюванні динамічних систем :** навчально-методичний посібник для студентів другого курсу другого (магістерського) рівня вищої освіти зі спеціальності «Прикладна марематика» [Електронне видання] / укладач Н. М. Кізілова. – Харків : ХНУ імені В. Н. Каразіна, 2023. – (PDF 100 с.)

У виданні наведені основні дані про метод молекулярної динаміки (або метод динаміки частинок), який зараз є найчастіше використовуваним методом математичного моделювання систем атомів, молекул, молекулярних структур, наночастинок, нано- і мікрорідин, а також макроскопічних систем. Наведені основні алгоритми інтегрування диференціальних систем рівнянь динаміки частинок як точок, типи силових полів, алгоритми, недоліки методу та його подальші ускладнення і модифікації, а також приклади застосувань методу у хімічних, фізичних і біологічних науках.

Для студентів другого (магістерського) рівня вищої освіти зі спеціальності «Прикладна математика».

УДК 519.6+ 51-7

© Харківський національний університет
імені В. Н. Каразіна, 2023

© Кізілова Н. М., уклад., 2023

© Дончик І. М., макет обкладинки, 2023

Електронне навчальне видання комбінованого використання
Можна використовувати в локальному та мережному режимі

Кізілова Наталія Миколаївна

**МЕТОД ДИНАМІКИ ЧАСТИНОК
В МАТЕМАТИЧНОМУ МОДЕЛЮВАННІ ДИНАМІЧНИХ СИСТЕМ**

Навчально-методичний посібник для студентів другого курсу другого (магістерського) рівня вищої освіти зі спеціальності «Прикладна математика»

Коректор *О. В. Пікалова*
Комп'ютерне верстання *Н. М. Кізілова*
Макет обкладинки *І. М. Дончик*

Підписано до розміщення 20.04.2023. Гарнітура Times New Roman.
Ум. друк. арк. 3,75. Обсяг 4,462 Мб. Зам. № 93/23.

Харківський національний університет імені В. Н. Каразіна,
61022, м. Харків, майдан Свободи, 4.
Свідоцтво суб'єкта видавничої справи ДК № 3367 від 13.01.2009
Видавництво ХНУ імені В. Н. Каразіна

ЗМІСТ

Вступ.....	5
1. Метод молекулярної динаміки	8
1.1. Основні визначення	8
1.2. Історія методу.....	10
1.3. Теорія і алгоритми методу молекулярної динаміки.....	16
1.4. Алгоритм Верле	18
1.5. Дрейф енергії.....	21
1.6. Алгоритм Бітмана	22
1.7. Алгоритм leapfrog	23
1.8. Системи з обмеженнями.....	25
1.9. Інші інтегратори.....	27
1.10. Потенціали в системах частинок	28
1.11. Статистичні характеристики системи	36
1.12. Загальний алгоритм ММД.....	39
1.13. Недоліки і обмеження ММД.....	40
2. Модифікації і узагальнення ММД	45
2.1. Керована молекулярна динаміка (КМД).....	46
2.2. Методи квантової механіки та змішані КМ/ММ методи	47
2.3. Моделі для специфічних зовнішніх полів	47
2.4. Стратегії вдосконалення формування ансамблю	48
3. Моделі силових полів.....	50
3.1. REBO.....	51
3.2. AIREBO.....	51
3.3. CHARMM	52
3.4. GROMACS.....	52
3.5. GROMOS	52
3.6. AMBER	53
3.7. LAMMPS.....	53
3.8. NAMD	53
3.9. Інші моделі	53
3.10. Методи машинного навчання і нейронних мереж у ММД	53
4. Програмне забезпечення ММД	61
4.1. Структура і складові пакетів програм.....	61
4.2. Приклади програмних пакетів для різних прикладних задач	62
4.3. Пакет PyMol with Python	68
4.3.1. Інтерфейс програми	69
4.3.2. Візуалізація молекул.....	70

4.3.3. Опції збереження в PyMOL	71
4.3.4. Скрипти PyMOL with Python	72
4.3.5. Графік потенціальної енергії	73
4.3.6. Ініціалізація частинок і розрахунок поля сил.....	74
4.3.7. Інтегрування рівнянь руху	75
4.3.8. Приклади візуалізації і розрахунків ММД	76
Приклад 1.....	77
Приклад 2.....	80
Приклад 3.....	82
5. Приклади застосування ММД в сучасних технологіях.....	85
6. Приклади задач для самостійного розв’язання за допомогою ММД.....	95
7. Перелік посилань	99

ВСТУП

Математичне моделювання широко використовується в різних галузях науки, виробництві і господарстві. Математичні моделі, які зводяться до систем диференціальних рівнянь, важливі у дослідженнях фізики конденсованого стану, хімічної кінетики, фізичної хімії, фізики фазових переходів, механіки рідини, механіки наноструктур і нанорідин, біології, генетики, сучасних технологій виробництва нових матеріалів, дизайну ліків та багатьох інших.

Точні аналітичні розв'язки таких систем можливі лише у невеликій кількості спрощених постановок задач, тому вкрай важливі наближені методи розв'язання, особливо чисельні методи, які дозволяють облічувати розв'язки достатньо складних систем рівнянь з багатьма змінними. Методи обчислювальної математики базуються на формулюванні наближеної системи рівнянь, в яких обчислення похідних за часом та просторовими координатами замінюється дискретними аналогами. Деякі підходи використовують *опис Лагранжа*, коли система координат і умовний спостерігач, який зв'язаний з нею, рухається разом із досліджуваною геометрією. Наприклад, популярний *метод скінченних елементів*, в якому замість пошуку розв'язків континуальних рівнянь обчислюються значення невідомих величин в скінченному наборі вузлів, які зв'язані з системою (частинками рідини, твердого тіла, елементів конструкцій, біологічних клітин і тканин). Рівняння для цих обчислень отримуються шляхом мінімізації загальних функціоналів систем рівнянь [1,2]. Протилежний, *Ейлерів підхід*, базується на уявленні системи координат як фіксованої сітки у просторі, а частинки досліджуваної системи рухаються і потрапляють у вузли цієї сітки, густина якої підбирається відповідно до особливостей задачі, а також умов збіжності і стійкості наближеного розв'язку. При цьому похідні за часом і координатами замінюються дискретними аналогами (*метод скінчених різниць*, finite difference methods), які засновані на використанні розкладання функцій, які обчислюються, у ряди Тейлора. В отримані дискретні аналоги похідних входять сусідні точки з найближчого оточення даної точки. Методом ітерацій проводяться розрахунки значень функцій в точках сітки в послідовні моменти часу з деяким кроком Δt .

Нові розробки і підходи останніх десятиріч тісно зв'язані з мікро- і нанотехнологіями і біотехнологіями, які дозволяють створювати пристрої, робота яких визначається фізико-хімічними процесами на мікро- і нанорівні, тобто за участю від десятків тисяч до навіть десятків молекул. Існує багато малих пристроїв, наприклад, мікрорідинні системи, в яких мікрооб'єми рідини нагріваються або охолоджуються, змішуються або розділяються, проходять хімічні реакції і виділяється енергія та багато інших. В багатьох мініатюрних пристроях використовуються малі комп'ютерні чіпи, мікродвигуни і маніпулятори, медичні лабораторії на чіпах (lab on a chip devices),

які потребують охолодження, змащування, постачання і виведення речовин. В Мікро- і наносистемах значно підвищуються вклад процесів на поверхнях розділу рідина - газ або рідина - тверде тіло, в яких переважну участь беруть взаємодії між молекулами. Традиційні методи і підходи до моделювання тепломасопереносу, особливо на поверхнях розділу фаз, мають суттєві обмеження при описі процесів в мікро- і наносистемах. Тому в останні десятиріччя суттєво зріс інтерес до моделювання процесів на нано-, мікро- і мезорівнях шляхом прямого інтегрування систем рівнянь динаміки Ньютона для великої кількості частинок – атомів, молекул, наночастинок, мікрочастинок, біологічних клітин та мезоструктури у нових перспективних матеріалах (композити, наноструктуровані поверхні, тонкі плівки та ін.).

В останні роки популярним став *метод динаміки частинок* (або метод молекулярної динаміки, ММД), який заснований на прямому інтегруванні рівнянь Ньютона руху частинок, які складають будь-яку систему. Для газів, рідин і твердих матеріалів це можуть бути атоми, молекул або крупніші частинки, такі як гранули пористих середовищ. Для молекулярної біології частинки – це молекули білків, вуглеводів, або навіть клітин. ММД дозволяє з достатньо високою точністю відтворювати тривимірну структуру молекули у водному або іншому розчині, будувати нові ліки, які будуть впливати на конкретні частини білків або на рецептори бактеріальних клітин і відповідні структури вірусів. ММД відіграє важливу роль у вивченні макромолекулярних систем частково тому, що закони класичної статистичної механіки можуть головним чином керувати процесами, які відбуваються в експериментальних умовах. ММД – це підхід комп'ютерного моделювання, який використовується для чисельного інтегрування рівнянь руху атомів і молекул за наближеннями відомих законів фізики і хімії.

Структура біомолекул на атомному рівні є надзвичайно корисною і зазвичай дає істотне уявлення про те, як функціонує молекула. Проте атоми в молекулах перебувають у постійному русі, а молекулярна функція і міжмолекулярні взаємодії залежать від динаміки залучених молекул. Тому для молекулярних біологів і біофізиків, які намагаються зрозуміти, як працює білок чи інша біомолекула, важливо мати не просто статичний «знімок», а можливість спостерігати ці біомолекули у дії, збурювати їх на атомарному рівні та дивитися, як вони реагують. На жаль, спостерігати за рухом окремих атомів та збурювати їх потрібним чином складно. Привабливою альтернативою є робота з комп'ютерним моделюванням відповідних біомолекул на атомному рівні за допомогою ММД. Ця тенденція особливо помітна в неврології: ММД використовувався для вивчення білків, які важливі для передачі сигналів нейронами, для допомоги в розробці препаратів, які впливають на нервову систему, для виявлення механізмів агрегації білків, які пов'язані з нейродегенеративними захворюваннями, а також для закладення основи для розробки вдосконалених оптогенних інструментів.

Значний прогрес в обчислювальній техніці і алгоритмів паралельних розрахунків дозволяє використовувати цей метод для систем, які налічують навіть мільйони частинок, що робить цей простий метод дуже популярним при математичному моделюванні різних систем. Останні кілька років симуляції ММД стали набагато потужнішими і доступними. Донедавна більшість високоефективних робіт, виконуваних з використанням методу, потребувала використання суперкомп'ютерів (кластерів). Але комп'ютерне обладнання, яке нещодавно з'явилося, зокрема графічні процесори (GPU), дозволяє виконувати потужні симуляції локально і при скромних витратах. Графічні процесори дозволили симуляціям, які виконуються на одному або двох недорогих комп'ютерних чіпах, перевершити за продуктивністю ті, що раніше виконувались на більшості суперкомп'ютерів. Ці графічні процесори зробили симуляції в біологічно значущих часових масштабах доступними для набагато більшого числа дослідників, аспірантів і студентів ніж будь-коли раніше. Пакети програмного забезпечення з використанням ММД також стали простішими у використанні, з кращою підтримкою для нефаківців. Нарешті, хоча фізичні моделі, які лежать в основі ММД, за своєю сутністю є наближеними, за останні роки вони стали значно точнішими завдяки співпраці фізиків і математиків.

Значні поліпшення швидкості моделювання, точності та доступності, а також поширення відкритих експериментальних даних про склад і структуру матеріалів підвищили привабливість біомолекулярного моделювання для експериментаторів. Зараз експерименти з молекулярними системами, розчинами, суспензіями клітин, матеріалами та різними «розумними» покриттями можна проводити на комп'ютері (*in silico*) з використанням ММД, що значно зменшує витрати на матеріали і експериментальну техніку. Приклади таких «комп'ютерних експериментів» теж наведені у даному виданні, а відповідні задачі будуть запропоновані студентам як практичні або індивідуальні завдання.

Таким чином, ММД являє собою інтерфейс між теорією і експериментом і може бути використаний як **віртуальний експеримент**. ММД досліджує зв'язок між молекулярною структурою, рухом і функцією молекули, структури, тканини або матеріалу. Молекулярна динаміка є **міждисциплінарним методом**, і її закони та теорії походять з різних галузей, таких як математика, фізика та хімія. Крім того, він реалізує алгоритми з інформатики та теорії інформації. Спочатку метод був розроблений в рамках теоретичної фізики, але сьогодні застосовується також і в інших галузях, таких як комп'ютерне моделювання матеріалознавства та біомолекулярних систем. **Молекулярна динаміка як дисципліна** вклучає молекулярне моделювання та комп'ютерне моделювання на основі законів статистичної механіки. Основне обґрунтування методу МД як спеціальної дисципліни молекулярного моделювання та комп'ютерного моделювання на основі статистичної механіки.

1. МЕТОД МОЛЕКУЛЯРНОЇ ДИНАМІКИ

1.1. Основні визначення

Метод молекулярної динаміки (ММД) – це метод комп’ютерного моделювання фізичного руху атомів, молекул та будь-яких інших частинок, які вважаються малими у порівнянні з розміром області, в якій знаходяться (рис. 1). Наприклад, такими «частинками» можуть бути малі неорганічні (NaCl , Ca_2CO_3 , ...) або великі органічні (цукор, крохмаль, білки, ...) молекули, малі частинки (оксиди, кристали, нанотрубки, волокна, ...), і навіть окремі тварини або люди, коли моделюється рух зграї птахів, табуна коней або натовпу людей на деякій достатньо великій території. ММД широко застосовується в хімічній фізиці, біофізиці, матеріалознавстві, нанонауках і нанотехнологіях [3] та ін.

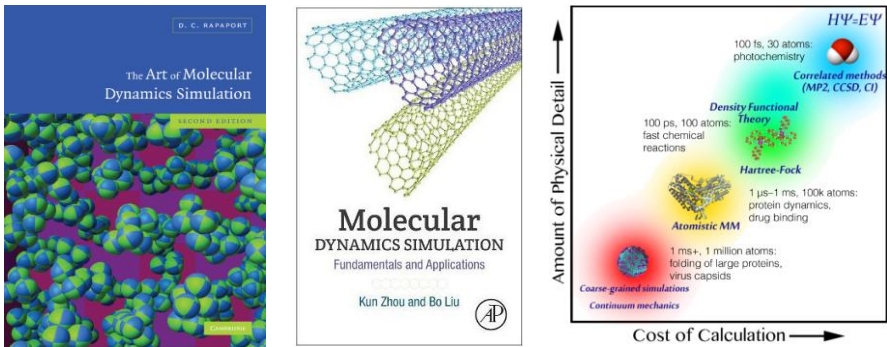


Рис. 1. Приклади методів і підходів до симуляцій ММД

Траєкторії руху частинок розраховуються шляхом чисельного розв’язання рівнянь Ньютона динаміки руху системи взаємодіючих частинок, де сили взаємодії обчислюються за допомогою міжатомних потенціалів або силових полів молекулярної механіки (ММ). При цьому вважається, що частинки можуть взаємодіяти під час зіткнень (зустрічей) протягом деякого фіксованого періоду часу. В результаті підхід дає змогу спостерігати еволюцію динамічної системи, тобто її поведінку з часом.

Методи ММ, які також називаються методами силового поля, вважають потенціальну енергію взаємодії атомів і молекул функцією координат їх ядер (наближення Борна-Опенгеймера), без урахування складної структури атомів і руху електронів в них. В дійсності між компонентами атомів (електрони, нейтрони, протони та їх складові) існують квантово-механічні взаємодії. Методи квантової механіки (МКМ) описують ці взаємодії і розглядають електронну структуру в молекулярній системі на основі підходів

і рівнянь квантової механіки. На жаль, навіть коли електрони та їх рух не беруть до уваги, все одно потрібно враховувати велику кількість частинок в кожному атомі, що робить обчислення дуже трудоемними. Тому зазвичай для виконання обчислень у системах, які містять значну кількість атомів, використовуються методи ММ. У деяких випадках ММ може надати такі ж точні результати, як високорівневі обчислення МКМ, але набагато швидше. Однак ММ не може забезпечити властивості, які залежать від електронного розподілу в молекулі. ММ використовує прості моделі взаємодій у системі, які включають внесок від розтягування міжатомних зв'язків, їх згинання під кутом і обертання навколо одинарних зв'язків. Суттєвою особливістю ММ є використання набору параметрів, розроблених для відносно невеликої кількості випадків, застосовуваних до ширшого кола проблем. Крім того, параметри, отримані з даних про малі молекули, також можуть бути використані для дослідження набагато більших молекулярних систем, таких як білки, нуклеїнові кислоти, полімери тощо.

Інтегрування за часом рівнянь Ньютона проводиться на малих кроках часу Δt , так що для розрахунків тривалої еволюції системи потрібні тисячі і більше ітерацій, що призводить до накопичення похибки чисельних розрахунків. Крім того, коли пара частинок зближується, необхідно зменшувати крок інтегрування, а при цьому швидко зростають чисельні помилки. Крім того, при прямому інтегруванні рівнянь другого порядку число обчислень для кожного кроку зростає зі зростанням числа частинок приблизно як $\sim N^2$. Для зменшення числа операцій і похибки інтегрування потрібне використання спеціальних алгоритмів і параметрів.

Результати інтегрувань у вигляді швидкостей, координат і траєкторій руху кожної з частинок можуть бути осереднені і використані для визначення макроскопічних термодинамічних властивостей системи – температури, тиску, внутрішньої енергії та інших. При цьому припускається виконання *ергодичної гіпотези* (від грецьк. *érgon* – робота і *hodós* – шлях), а саме: середні за часом значення фізичних величин, які характеризують систему, дорівнюють їх середнім статистичним значенням.

Ергодична гіпотеза слугує для обґрунтування статистичної фізики. Фізичні системи, для яких справедлива ця гіпотеза, називають ергодичними. Ергодична гіпотеза у класичній статистичній механіці рівноважних систем – це припущення про те, що середні за часом від функцій, які залежать від координат та імпульсів усіх частинок системи (фазових змінних), взяті по траєкторії руху системи як точки у фазовому просторі, дорівнюють середнім статистичним за рівномірним розподілом фазових точок у нескінченно тонкому шарі поблизу поверхні постійної енергії. Такий розподіл частинок в системі називається *мікроканонічним розподілом Гіббса*.

Для розуміння малого масштабу рухів у просторі і за часом, які вивчаються за допомогою ММД, характерні розміри систем (L), амплітуди коливань (H) і характерні часи (T) наведені в таблиці 1.

Таблиця 1. Значення характерних розмірів молекулярних систем (L), амплітуд коливань частинок (H) і характерні часи (T) фізичних процесів [4]

№ з/п	Система	L (нм)	H (нм)	T (с)	Метод
1	Колівання хімічного зв'язку	0.2-0.5	0.001-0.01	10^{-14} – 10^{-13}	КМ/ММ
2	Еластичні коливання білкової глобули	1-2	0.005-0.05	10^{-12} – 10^{-11}	Стандартні ММД
3	Обертання бокових ланцюгів молекули	0.5-1	0.5-1	10^{-11} – 10^{-10}	Стандартні ММД
4	Крутильні коливання груп	0.5-1	0.05	10^{-11} – 10^{-9}	Стандартні ММД
5	Згинальні коливання	1-2	0.1-0.5	10^{-11} – 10^{-7}	Динаміка Ланжевена
6	Обертання прихованих бічних ланцюгів	0.5	0.5	10^{-4} –1	Розширені ММД
7	Алостеричні переходи	0.5-4	0.1-0.5	10^{-5} –1	Розширені ММД
8	Локальна денатурація	0.5-1	0.5-1	10^{-5} –10	Розширені ММД
9	Циклічні рухи	1-5	1-5	10^{-9} – 10^{-5}	Броунівська динаміка
10	Спіральні переходи	1-5	1-5	10^{-9} – 10^{-6}	Розширені ММД
11	Угруповання білків	$\gg 1$	> 5	10^{-7} – 10^{-4}	Броунівська динаміка

1.2. Історія методу

ММД відноситься до моделювання систем із заданою кількістю (N) тіл, які взаємодіють між собою і з зовнішнім полем сил. Інтерес до дослідження еволюції *системи N тіл* виник ще у XVII столітті у зв'язку з задачами небесної механіки про взаємодію планет і їх супутників, стабільність Сонячної системи з багатьма планетами, які обертаються навколо неї, та інших зоряних систем. *Задача трьох тіл* небесної механіки полягає у визначенні відносного руху трьох тіл (матеріальних точок), які взаємодіють між собою відповідно до закону тяжіння Ньютона (наприклад, Сонце, Земля та Місяць). Ця задача, на відміну від задачі двох тіл, у загальному випадку не має рішення у вигляді аналітичних виразів. Відомі лише окремі точні рішення, і тільки для деяких початкових умов (початкових швидкостей і координат точок).

В 1892 – 1899 рр. видатний математик Анрі Пуанкаре довів, що існує безліч частинних розв'язків задачі трьох тіл. В 1912 р. фінський математик

Карл Зундман знайшов загальний аналітичний розв'язок задачі трьох тіл у вигляді рядів, які сходяться на будь-який момент часу і з будь-якими початковими умовами, але сходяться вкрай повільно. Зараз відомо, що системи із $n \geq 3$ диференціальних рівнянь порядку $m > 1$, які описують поведінку $N \geq 3$ тіл, що взаємодіють, відповідають системам з динамічних хаосом, які припускають існування біфуркацій розв'язків, тобто неоднозначності розв'язку системи диференціальних рівнянь.

Таким чином, обчислення розв'язків задачі N тіл з довільними початковими умовами можливе тільки шляхом чисельних розрахунків, і з другої половини ХХ ст. почали розроблятися відповідні алгоритми, засновані на використанні випадкових величин. Стимулом цих досліджень були деякі цікаві рішення математичних задач з використанням ймовірностей. Ще в 1777 р. французький дослідник де Бюффон сформулював таку задачу (голка Бюффона):

Площина розграфлена паралельними прямими, які розташовані на відстані $2a$ одна від одної (рис. 2а). На площину навмання кидають голку завдовжки b . Знайти ймовірність p того, що голка перетне одну з прямих. Задача має аналітичний розв'язок $p = b/\pi a$. Таким чином, експериментальним шляхом кидання голки і обчислення ймовірності p як відношення успішних випадків (голка перетинає пряму) до загального числа кидків, можна наприкінці облічити число π . Прямі експерименти (1864 р.) показали, що в циклі з 530 – 590 кидань голки можна обчислити значення π з точністю до $10^{-4} - 10^{-6}$.

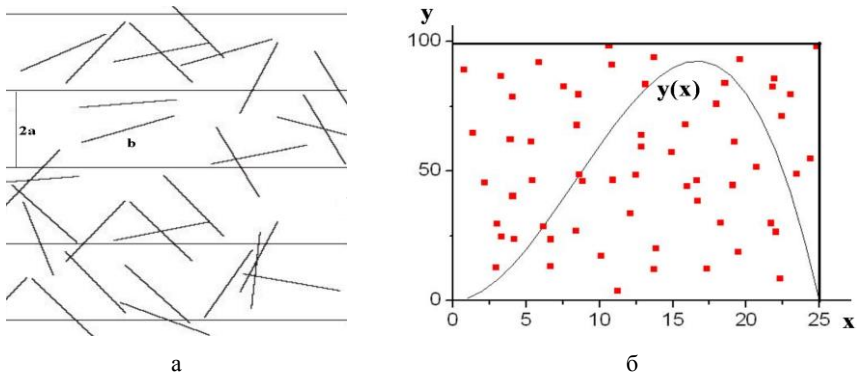


Рис. 2. Схема експерименту задачі «голка Бюффона» (а) і обчислення визначеного інтегралу методом «кидання точок» (б)

Можна показати, що підхід з випадковим киданням точок можна використати, наприклад, до **обчислення визначених інтегралів** як площини під графіком функції, яку інтегрують. Якщо розташувати область інтегрування в прямокутнику, такому що будь-яка сторона прямокутника містить хоча б

одну точку графіка функції, але не перетинає його (рис. 2б), і задавати випадковим шляхом координати точок, які кидають на прямокутник, то легко показати, що при достатньо великій кількості кидань відношення кількості точок, які потрапили під криву до загальної кількості точок, які кидали, дорівнює відношенню площини під кривою (тобто інтегралу функції) до площини прямокутника. Це твердження вірно також до будь-якої n -вимірної функції.

Цей підхід був використаний в *методах Монте-Карло* – широкому класі обчислювальних алгоритмів, які покладаються на повторну випадкову вибірку для отримання числових результатів. Основна концепція методу полягає у використанні випадковості для вирішення проблем, які в принципі можуть бути детермінованими. Ці методи часто використовуються у фізичних та математичних задачах, особливо коли важко або неможливо використовувати інші підходи.

Перші варіанти методу Монте-Карло були розроблені для вирішення проблеми голки Бюффона, що дозволяло високоточні експериментальні вимірювання числа π , e та інших ірраціональних чисел. У 1930-х рр. відомий фізик Енріко Фермі вперше експериментував з методом Монте-Карло для опису дифузії нейтронів. В 1946 р. фізики-ядерники з Лос-Аламосу досліджували дифузію нейтронів в активній зоні ядерного реактору. Незважаючи на наявність більшості даних вимірювань, таких як середня відстань, яку нейтрон пройде в речовині до того, як він зіткнеться з атомним ядром, або кількість енергії, яку нейтрон може виділити після зіткнення, фізики не змогли вирішити проблему дифузії з використанням традиційних детермінованих математичних методів. Наприкінці 1940-х рр. польсько-американський математик і фізик Станіслав Улам, який працював в Лос-Аламоській лабораторії, розробив сучасну версію методу Монте-Карло з ланцюгами Маркова. При цьому використовувався стохастичний підхід для апроксимації багатьох вимірних інтегралів у рівняннях перенесення, які описували рух нейтронів в ізотропному середовищі. Оскільки проект був секретний, метод не міг бути названий іменами його винахідників, і тому метод отримав назву «метод Монте-Карло» від назви однойменної комуни у князівстві Монако з численними казино.

Спочатку використання методу було дуже обмежене у зв'язку з відсутністю комп'ютерів. Потім перший американський комп'ютер ENIAC (1945 р.), який міг з великою швидкістю генерувати випадкові числа та використовувати їх у математичних моделях, був запрограмований для виконання повністю автоматизованих обчислень методом Монте-Карло, і з його допомогою за часом були розроблені проекти ядерної (1948 р.) і водневої (1950-ті рр.) бомб.

В 1950-ті рр. Алан Тьюрінг використовував методи Монте-Карло в своїх працях з повчальних машин для вибору мутацій генетичного типу. Розроблені генетичні евристичні методології динаміки частинок були потім успішно використані у молекулярній хімії. В 1960-ті роки інтенсивно роз-

роблялися теорії більш складних методів Монте-Карло для руху частинок, які відповідають молекулам рідини або твердого тіла, або марківських інтерпретацій класу нелінійних параболічних диференціальних рівнянь у часткових похідних, які виникають у механіці рідини. Ще в 1941 р. інтегрування рівнянь руху N тіл було проведене за допомогою аналогових комп'ютерів. Відомі спроби моделювання руху атомів шляхом побудови фізичних моделей, що склалися з макроскопічних тіл (сфер), які відповідали б атомам. Мета такого моделювання полягала в тому, щоб відтворити структуру рідини за допомогою великої кількості сфер, і використати таку модель для вивчення руху рідини з урахуванням випадковості зіткнень сфер під час їх руху.

З появою комп'ютерів стався великий прорив, і стохастичний підхід методів Монте-Карло почав широко застосовуватися у багатьох галузях науки для обробки сигналів, розв'язання задач оптимізації і керування тощо. Сукупність різних алгоритмів, які використовують стохастичний підхід, називають *стохастичним моделюванням*.

В 1953 – 1954 рр. Енріко Фермі та його колеги чисельно розраховували траєкторії руху в системах з 32 і 64 частинок з урахуванням нелінійних сил взаємодії (квадратична і кубічна залежності від відстані) за допомогою комп'ютера MANIAC-I (**M**athematical **A**nalyzer, **N**umerator, **I**ntegrator, and **C**omputer).

В 1957 р. було опубліковане дослідження¹ молекулярної динаміки фазових переходів (рідина - тверде тіло) в системі твердих сфер, які знаходилися в прямокутній області і взаємодіяли тільки під час зіткнень, а між зіткненнями рухалися вільно. Обчислення проводилися на системах з 32, 108, 256 і 500 частинок з використанням комп'ютерів UNIVAC (**U**niversal **A**utomatic **C**omputer) і IBM 704. Обчислення процесу 7000 зіткнень в системі з 32 частинок зайняло 1 годину обчислень. В 1959 – 1960 рр. вийшли статті тих самих авторів з викладенням алгоритму чисельних розрахунків для кількох степенів частинок, а також результати моделювання для системи пружних сфер. В 1960 р. була опублікована стаття², в якій розглядалась молекулярна динаміка процесу радіоактивного пошкодження міді за рахунок значного переміщення атомів Cu за рахунок високих енергій радіоактивного бомбардування матеріалу. Тема роботи була зумовлена задачами захисту від ядерного нападу. В моделі були введені парні взаємодії частинок з використанням потенціалу Борна-Майєра $\Phi_{j,k} = V \exp(-\beta r_{j,k})$, $\beta, V = \text{const}$, а також пружних сил відштовхування протягом зближень. Початкова конфігурація системи і результати обчислення переміщень атомів в результаті опромінення наведені

¹ Alder B.J., Wainwright T.E. Phase Transition for a Hard Sphere System. The Journal of Chemical Physics. 1957. Vol. 27 (5). P. 1208–1209.

² Gibson J.B., Goland A.N., Milgram M., Vineyard G.H. Dynamics of Radiation Damage. Physical Review. 1960. Vol. 120. P. 1229–1253.

на рис. 3а,б відповідно. Для інтегрування використовувався метод скінчених різниць. Обчислення проводилися з використанням IBM 704, і один крок розрахунків займав близько хвилини.

У 1964 р. було проведене перше моделювання динаміки системи з 864 атомів рідкого аргону³, в якому використовувався потенціал Леннарда-Джонса (LJ-potential). Отримані розрахунки коефіцієнту дифузії в системі за допомогою комп'ютера CDC 3600 добре погоджувались з експериментальними даними. На сьогодні потенціал LJ є одним із найбільш часто використовуваних міжмолекулярних потенціалів в ММД. Програмний код, який був розроблений для розрахунків, пізніше часто використовувався в різних дослідженнях.

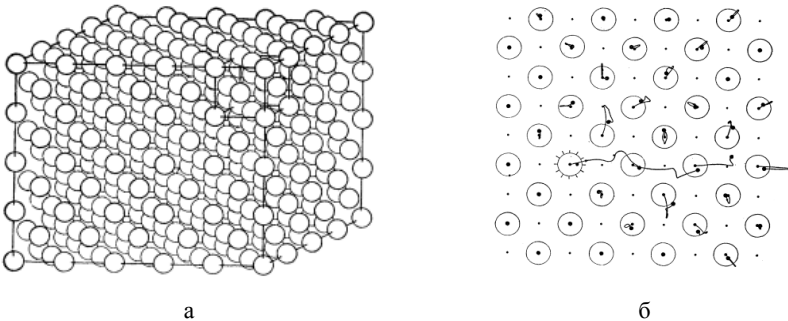


Рис. 3. Початкова конфігурація системи атомів міді в решітці (а) і їх скінчені переміщення за рахунок передачі значної кінетичної енергії після радіоактивного опромінування (б) (адаптовано з роботи³)

Таким чином, історія розв'язання детермінованих задач з використанням стохастичних методів привела від задачі N тіл до методів Монте-Карло, а потім до ММД, який був вперше використаний саме у теоретичній фізиці. Потім у цілій низці публікацій 1960 – 1970-х він був використаний у матеріалознавстві для моделювання структури і властивостей матеріалів, які складаються із багатьох частинок з різними фізичними властивостями. З 1970-х рр. ММД також широко застосовується в біохімії та біофізиці для уточнення 3D-структур білків, первинна структура яких – це ланцюг з амінокислот, а також інших крупних біологічних молекул на основі експериментальних обмежень, одержаних за допомогою рентгенівської кристалографії або спектроскопії. Моделювання «скручування» (фолдингу) ланцюга в клубок теж проводиться за допомогою ММД. Фінальна структура обчислюється на основі мінімізації потенціальної енергії системи частинок та інших припущень. Перше

³ Rahman A. Correlations in the Motion of Atoms in Liquid Argon. Phys. Rev. 1964. Vol. 136, A405.

моделювання⁴ фолдингу за допомогою ММД привело до розробки ефективних методів сучасної обчислювальної біохімії і біофізики.

У нанонауках і нанотехнологіях ММД використовується для вивчення явищ на молекулярному і атомному рівнях, динаміку яких не можна спостерігати безпосередньо, наприклад, зростання кристалів, утворення тонких плівок, процеси в наноматеріалах і нанопристроях, які ще не створені або не можуть бути створені. У структурній біології ММД застосовується для вивчення рухів великих молекул, таких як білки і нуклеїнові кислоти, що може бути корисним для інтерпретації результатів певних біофізичних експериментів, моделювання взаємодії з іншими молекулами, для прогнозування 3D-структури білка шляхом імітації згортання ланцюга з амінокислот, утвореного в процесі синтезу з матричної РНК у рибосомах клітин. Підходи ММД використовуються для пошуку активних центрів в молекулах і молекулярного дизайну ліків, молекулярних рецепторів/сенсорів, в дослідженнях упакування білків і спіралізації ДНК, фазових переходів в рідких кристалах, при віртуальному моделюванні клітин і клітинних структур для вивчення взаємодії їх між собою та з різними речовинами. Деякі приклади результатів моделювання систем частинок за допомогою ММД наведені на рис. 4.

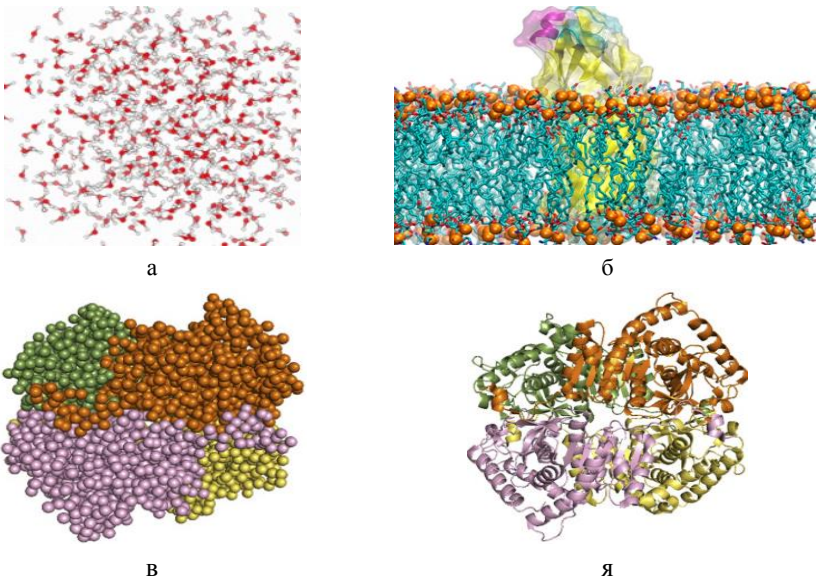


Рис. 4. ММД- її моделювання молекул води (а), мембрани клітини (б), молекули білку як сукупності атомів (в) та з визначенням топології (г)

⁴ Levitt M., Warshel A. Computer simulation of protein folding. Nature. 1975. Vol. 253. P. 694 – 698.

Для перевірки точності таких розрахунків використовуються експериментальні зображення молекул за допомогою рентгенівської кристалографії або ЯМР-спектроскопії. З 1994 р. із періодичністю кожні два роки проводиться масштабний всесвітній експеримент із передбачення білкових структур CASP (Critical Assessment of Protein Structure Prediction, критична оцінка передбачення білкових структур), в якому на постійній основі бере участь понад 100 дослідницьких груп [3]. CASP перевіряє результати моделювання білкових структур різними методами і алгоритмами шляхом зіставлення з даними вимірювань та надає незалежну оцінку структурного моделювання. Основна мета CASP – допомога у поліпшенні методів визначення тривимірної структури білків з їх послідовностей амінокислот. У 2013 р. за розробку багатомасштабних моделей складних хімічних систем з використанням ММД була присуджена Нобелівська премія з хімії.

1.3. Теорія і алгоритми методу молекулярної динамки

ММД полягає у ітеративних розрахунках траєкторій руху всіх молекул (частинок), що входять до системи шляхом інтегрування їх рівнянь руху

$$m_j \frac{d^2}{dt^2} \vec{r}_j = \vec{F}_j, \quad (1)$$

де $j=1,2,\dots,n$ – номер частинки, m_j і \vec{r}_j – маса і радіус-вектора (тобто, вектор координат частинки у вибраній системі координат) частинок, \vec{F}_j – загальна (рівнодійна) сила, яка діє на j -ту частинку як з боку зовнішніх сил, так і з боку інших частинок, тобто \vec{F}_j залежить не тільки від координат частинки \vec{r}_j , але й від координат всіх частинок, які знаходяться поблизу j -ї і можуть безпосередньо впливати на її рух. Приклад системи з двох типів частинок і сил їх попарної взаємодії наведений на рис. 5а.

Моделювання динаміки частинок системи проводиться у деякій обмеженій області простору (3D-задачі) або площини (2D-задачі) зазвичай кубічної просторової комірки з періодичними умовами на границях області. Останнє означає, що протилежні грані області, в якій рухаються частинки системи, ототожнюються, внаслідок чого формально моделюється не скінченно велика система частинок. Частинка, яка на якомусь кроці за часом вискочила за стінку області з будь-якої сторони, потрапляє в ту ж область з протилежної сторони (рис. 5б), що необхідно для збереження числа частинок у системі.

Для інтегрування системи звичайних диференціальних рівнянь (ЗДР) другого порядку треба задати початкові умови

$$\vec{r}_j \Big|_{t=t_0} = \vec{r}_j^0, \quad \frac{d}{dt} \vec{r}_j \Big|_{t=t_0} = \vec{v}_j^0, \quad (2)$$

де \vec{r}_j^0 і \vec{v}_j^0 – координати і швидкості всіх частинок в деякий початковий момент часу t_0 .

Початкові координати задають виходячи з початкової структури матеріалу, наприклад, кристаліта, частинки якого знаходяться у вузлах кристалічної решітки, або газу, частинки якого знаходяться на деяких відстанях, які задають шляхом генерації координат як випадкових чисел з заданих діапазонів. Початкові швидкості повинні задовольняти 1) умові відсутності макроскопічного руху (течії) $\sum_{j=1}^N \vec{v}_j = 0$ і 2) умові термодинамічної рівноваги

$$\sum_{j=1}^N m_j \vec{v}_j^2 = 3k_B T, \text{ де } T \text{ – відома температура системи.}$$

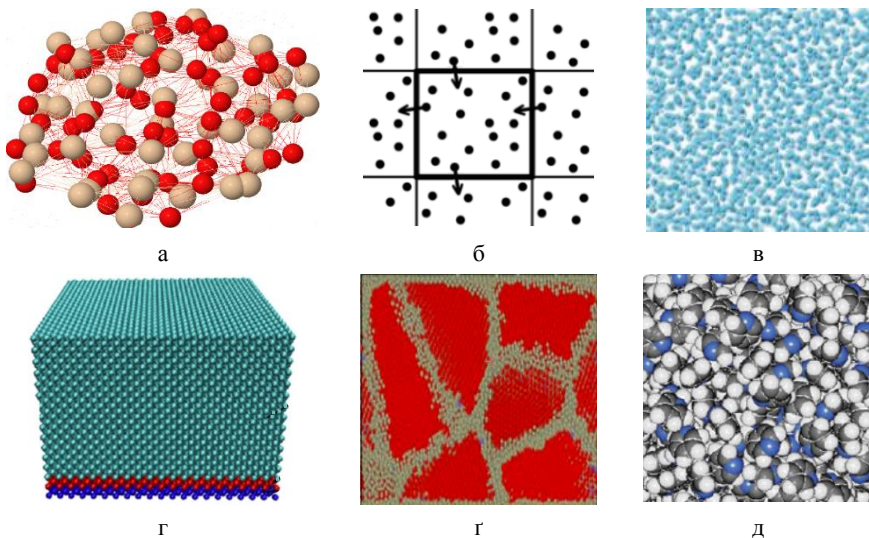


Рис. 5. Ілюстрації сил попарних взаємодій (а), періодичних граничних умов (б) модельованої періодичної системи і початкових конфігурацій ММД у вигляді рідини (в), кристалу (г), полікристалу (г) і аморфного тіла (д)

Сили, які входять до правих частин рівнянь (1), можуть бути потенційними, тобто мати потенціали Φ_k , так що $\vec{F}_k = -\nabla\Phi_k$ (сили тяжіння, пружності, електростатичні та ін.) або не потенційними (сили сухого тертя, в'язкості та ін.). Таким чином, розв'язання задачі динаміки системи частинок зводиться до завдання значень координат і швидкостей всіх частинок в початковий момент часу t_0 відповідно (2), обчислити всі сили їх взаємодії в момент часу t_0 в правих частинах (1), які, відповідно до положень класичної механіки,

задають стан системи у наступний момент часу. Далі проводиться інтегрування рівнянь (1) за допомогою вибраного інтегратора з обчисленням координат і швидкостей всіх частинок в системі в наступний момент часу $t_0 + \Delta t$.

Якщо розв'язується задача «статики» молекули або системи атомів, молекул, клітин, нано- або мікрочастинок, на кожному кроці розраховується загальна потенціальна енергія системи $U(\vec{r}_1, \vec{r}_2, \dots, \vec{r}_n)$ як функція координат всіх частинок. Вважається, що глобальний цієї функції відповідає стабільній конфігурації молекули або структури, тоді як локальні мінімуми дають метастабільні стани, можливі переходи між якими за допомогою сил або випадкових збурень подалі вивчаються, у тому числі експериментально. Таким чином, ММД дає можливість проведення *комп'ютерного експерименту in silico*, що є проміжною ланкою між аналітичними теоріями фізики, хімії, біології та фізико-хімічним експериментом, який часто потребує коштовного обладнання та інших матеріальних ресурсів. Фізичний експеримент, який потребує вивчення багатьох комбінацій речовин, може тривати довго, тоді як комп'ютерний експеримент означає поступовий перегляд різних комбінацій з автоматичним повторним запуском софту, що значно прискорює отримання розв'язку задачі.

1.4. Алгоритм Верле

Найбільш поширеним в ММД є алгоритм інтегрування Верле-Штрюмера. Розглянемо розкладання в ряд Тейлора радіус-векторів $\vec{r}(t + \Delta t)$ і $\vec{r}(t - \Delta t)$:

$$\begin{aligned}\vec{r}(t + \Delta t) &= \vec{r}(t) + \vec{v}(t)\Delta t + \frac{\vec{a}(t)}{2}\Delta t^2 + \frac{\vec{b}(t)}{3!}\Delta t^3 + O(\Delta t^4), \\ \vec{r}(t - \Delta t) &= \vec{r}(t) - \vec{v}(t)\Delta t + \frac{\vec{a}(t)}{2}\Delta t^2 - \frac{\vec{b}(t)}{3!}\Delta t^3 + O(\Delta t^4),\end{aligned}\quad (3)$$

де $\vec{v}(t)$ – швидкість, $\vec{a}(t) = \vec{F}(t)/m$ – прискорення, $\vec{b}(t)$ – похідна від прискорення або ривок (jerk).

Після складання рівнянь (3) отримаємо

$$\vec{r}(t + \Delta t) = 2\vec{r}(t) - \vec{r}(t - \Delta t) + \vec{a}(t)\Delta t^2 + O(\Delta t^4), \quad (4)$$

де прискорення $\vec{a}(t)$ обчислюється з параметрів фізичного поля на даний момент часу.

Таким чином, алгоритм (4) дозволяє обчислити значення координат точки в наступний момент часу, використовуючи її координати на двох попередніх кроках без обчислення швидкості, тобто без чисельного диференціювання.

Цей підхід був використаний Ньютоном у його першій книзі «Philosophiæ Naturalis Principia Mathematica» (1687 р.) для виводу другого закону Кеплера,

а також в 1791 р. Даламбером. В 1907 р. норвезький астрофізик Карл Штрюмер (Størmer) використав алгоритм для розрахунків траєкторій заряджених частинок в магнітному полі. В 1909 р. британські астрономи Філіп Коуел (P.H. Cowell) і Ендрю Кроммелін (A.C.C. Crommelin) використали його для розрахунків траєкторії комети Галлея. В 1960-х французький фізик Лу Верле (Verlet) використав алгоритм у розрахунках системи молекул ММД. Зараз цей алгоритм називають інтегратором Верле, або Штермера-Верле.

Інтегратор Верле забезпечує добру чисельну стійкість, без значних додаткових обчислювальних витрат порівняно з методом Ейлера. Також він має інші властивості, важливі для фізичних систем, такі як збереження симплектичної форми на фазовому просторі, збереження лінійних перших інтегралів системи і фазового об'єму для низки систем. За рахунок видалення в (3) всіх членів непарного ступеня, алгоритм знижує рівень локальних помилок, які з'являються під час інтегрування внаслідок дискретизації.

В 2000 р. датська студія ІО Interactive розробила комп'ютерну гру Hitman: Codename47, в якій для розрахунків траєкторій точок використовувався інтегратор Верле. Починаючи з того часу, алгоритм широко використовується в комп'ютерній графіці.

Для порівняння, метод Ейлера розв'язання задачі (1), (2), який базується на використанні апроксимації другої похідної в (1) у вигляді

$$\frac{d^2\vec{r}}{dt^2} \approx \frac{\vec{r}(t + \Delta t) - 2\vec{r}(t) + \vec{r}(t - \Delta t)}{\Delta t^2} = \frac{\vec{F}}{m}, \quad (5)$$

має перший порядок точності при еквівалентній кількості операцій.

Інтегратор Верле (4) дозволяє обчислити координати точок в наступний момент часу $t + \Delta t$, виходячи з координат на двох попередніх кроках t і $t - \Delta t$, а також фізичного поля сил в момент часу t . Таким чином, для обчислень значень координат для першого кроку за часом треба використати інше наближення, наприклад, розкладення в ряд Тейлора

$$\vec{r}(\Delta t) = \vec{r}(t_0) + \vec{v}(t_0)\Delta t + \frac{\vec{a}(t_0)\Delta t^2}{2} + O(\Delta t^3) \quad (6)$$

з точністю наближення $\sim O(\Delta t^3)$. Значення $\vec{a}(t_0)$ не входять до умов (2) і часто постулюються на першому кроці як $\vec{a}(t_0) = 0$ з подальшою корекцією.

Якщо крок за часом змінюється, метод Верле не апроксимує розв'язок диференціального рівняння типу (1), що було детально показано в попередніх публікаціях⁵. При цьому більш точні значення дає використання формули, яка теж виводиться з розкладань Тейлора у вигляді

$$\vec{r}(t_{k+1}) = \vec{r}(t_k) + \frac{(\vec{r}(t_k) - \vec{r}(t_{k-1}))}{\Delta t_{k-1}} \Delta t_k + \vec{a}(t_k) \frac{\Delta t_k + \Delta t_{k-1}}{2}. \quad (7)$$

⁵ Skeel R. D. Variable Step Size Destabilizes the Störmer/Leapfrog/Verlet Method. BIT Numerical Mathematics. 1993. Vol. 33. P. 172–175.

Після закінчення розрахунків координат частинок необхідно розрахувати їх кінетичну енергію

$$K(t_k) = \sum_{j=1}^n \frac{m_j v_j^2(t_k)}{2} \quad (8)$$

і температуру

$$T(t_k) = \frac{2K(t_k)}{nk_B}, \quad (9)$$

де k_B – постійна Больцмана, n – число ступенів свободи молекули (у загальному випадку можливості для кожної частинки поступових рухів в просторі вздовж 3-х осей координат, $n=3$).

Для цього потрібно обчислити швидкості частинок, наприклад, за допомогою центральної різниці

$$\vec{v}(t) = \frac{(\vec{r}(t + \Delta t) - \vec{r}(t - \Delta t))}{2\Delta t}, \quad (10)$$

але при цьому обчислена швидкість на крок часу відстає від координати.

Кращі результати дає застосування того ж методу Верле для розкладань вектору швидкості в ряд Тейлора, а саме

$$\vec{r}(t + \Delta t) = \vec{r}(t) + \vec{v}(t)\Delta t + \frac{\vec{a}(t)\Delta t^2}{2}, \quad \vec{v}(t + \Delta t) = \vec{v}(t) + \frac{(\vec{a}(t) + \vec{a}(t + \Delta t))}{2}\Delta t. \quad (11)$$

Формула (11) може використовуватися на будь-яких кроках за часом, починаючи з першого. Алгоритм обчислень має такий вигляд:

- 1) розрахувати значення $\vec{v}(t + \Delta t / 2) = \vec{v}(t) + \vec{a}(t)\Delta t / 2$;
- 2) розрахувати значення $\vec{r}(t + \Delta t) = \vec{r}(t) + \vec{v}(t + \Delta t / 2)\Delta t$;
- 3) обчислити поле сил $\vec{F}(t + \Delta t)$ і прискорення $\vec{a}(t + \Delta t) = \vec{F}(t + \Delta t) / m$ для кожної з частинок;
- 4) розрахувати значення $\vec{v}(t + \Delta t) = \vec{v}(t + \Delta t / 2) + \vec{a}(t + \Delta t)\Delta t / 2$.

Зверніть увагу, що алгоритм Верле обчислення швидкості 1)-4) передбачає, що прискорення $\vec{a}(t + \Delta t)$ залежить тільки від координат частинок $\vec{r}(t + \Delta t)$ і не залежить від швидкості в той самий момент часу $\vec{v}(t + \Delta t)$, що вносить локальну похибку. Можна показати, що похибка у алгоритмі Верле для швидкості має той самий порядок, що і базовий алгоритм Верле для координат. Зауважимо, що алгоритм швидкості не обов'язково споживає більше пам'яті, тому що в базовому алгоритмі Верле ми відстежуємо два вектори положення, а у алгоритмі Верлі для швидкості – відстежуємо один вектор положення та один вектор швидкості.

Як видно з (4), (11), локальні похибки алгоритму Верле для координат і для швидкостей мають порядок $\sim O(\Delta t^4)$ і $\sim O(\Delta t^2)$, відповідно. Глобальні (кумулятивні) похибки, які є результатом накопичення локальних

помилки, мають і для координат, і для швидкостей порядок $\sim O(\Delta t^2)$, тому алгоритм Верле вважається інтегратором другого порядку точності. Цей алгоритм добре зберігає повну енергію системи. Оскільки будь-яка система молекул демонструє випадкові флуктуації, амплітуда яких зростає з температурою (молекулярно-кінетичний, або тепловий рух) і, таким чином, є системою хаотичною, то точний розрахунок фазових траєкторій на часі більшому так званого горизонту передбачення, неможливий, точність методу інтегрування не відіграє ключової ролі. Завданням та критерієм коректності розрахунків ММД щодо даних експериментів, є моделювання статистичних властивостей реальної фізичної системи. Для цього вирішальне значення має відсутність дрейфу повної енергії системи (energy drift).

1.5. Дрейф енергії

Дрейф енергії у комп'ютерному моделюванні – це поступова зміна загальної енергії E системи з часом. За законами механіки повна енергія системи не змінюється ($E = \text{const}$), вона може тільки переходити з одного типу до іншого (потенціальна в кінетичну або внутрішню і навпаки), однак при моделюванні і комп'ютерних розрахунках енергія може коливатися на малих часах і збільшуватися / зменшуватися – на довгих часах внаслідок використання скінченного кроку за часом. При цьому важливою є симплектичність інтегратора ММД, який використовується.

Симплектичний інтегратор в математиці – це така схема чисельного інтегрування гамільтонових систем, яка входить до підкласу геометричних інтеграторів, які за визначенням є канонічними перетвореннями. Пошук симплектичних інтеграторів для різних класів задач математичної фізики є важливою областю досліджень математики⁶.

Так, якщо замість точного обчислення координат точок $\vec{r}_j(t)$ і їх швидкостей $\vec{v}_j(t) \equiv \dot{\vec{r}}_j(t)$ ми обчислюємо швидкість $\vec{v}_j^{\text{num}}(t) \equiv \vec{v}_j(t) + \delta\vec{v}_j(t)$ з деякою малою похибкою $\delta\vec{v}_j(t)$, загальна енергія руху системи буде дорівнювати

$$E = \sum_{j=1}^N m_j (\vec{v}_j(t) + \delta\vec{v}_j(t))^2 = \sum_{j=1}^N m_j (\vec{v}_j(t))^2 + 2 \sum_{j=1}^N m_j \vec{v}_j(t) \delta\vec{v}_j(t) + \sum_{j=1}^N m_j (\delta\vec{v}_j(t))^2, \quad (12)$$

де перший доданок в правій частині (12) відповідає реальній енергії системи, яка повинна зберігатися; другий доданок за відсутності кореляцій між швидкостями дорівнює нулю; а третій визначає загальний дрейф (шриффт) енергії за рахунок накопичення малих похибок на великому числі точок системи.

⁶ Yoshida H. Construction of higher order symplectic integrators. Physics Letters, Ser.A. 1990. Vol. 150. p. 262–268.

У випадку *несимплектичного інтегратора* (проста схема Ейлера, сімейство методів Рунге-Кутти) дрейф енергії є загасаючим. Для симплектичних інтеграторів, які зазвичай використовуються в ММД (сімейство інтеграторів Верле), має місце поступове збільшення енергії на дуже тривалих масштабах часу, хоча похибка залишається приблизно постійною. Такі інтегратори насправді не відтворюють реальну гамільтонову механіку системи частинок, але замість того вони відтворюють близький і зв'язаний «тінювий» гамільтоніан, значення якого вони зберігають на багато порядків точніше. Точність збереження енергії для справжнього гамільтоніана залежить від кроку інтегрування за часом. Це обмеження на максимальний розмір кроку, який відповідає умові стійкості чисельної схеми, є пропорційним до періоду найбільш швидких коливань в системі. Якщо період власних коливань системи є $T = 2\pi / \omega$, де ω – кругова частота), то умова на крок має вигляд $\Delta t \leq \frac{2\pi m}{\omega n}$, де m, n – натуральні числа, які визначають порядок параметричного резонансу в схемі. Так, для алгоритму Верле резонанси до четвертого порядку ($n/m=4$) часто призводять до чисельної нестабільності, що викликає необхідність обмежень кроку $\Delta t \leq \sqrt{2} / \omega \approx 0.225T$.

Оскільки найшвидші рухи у більшості молекулярних систем пов'язані з рухом найлегших атомів водню, зазвичай використовуються алгоритми для обмеження руху водню і, таким чином, збільшення максимального кроку, який можна використовувати в моделюванні зі збереженням стабільності схеми. Оскільки масштаби рухів з часом і більш важких атомів не сильно відрізняються від масштабів рухів водню, на практиці це дозволяє збільшити крок за часом для всієї системи приблизно в 2 рази. Для збереження стійкості алгоритму Верле в ММД використовується крок $\Delta t = 1 \text{ фс} = 10^{-15} \text{ с}$ (фемтосекунда) для моделювання без обмежень на відстані між молекулами і $\Delta t = 2 \text{ фс}$ – для моделювання з обмеженнями, хоча для певних систем або вибору параметрів можуть бути вибрані і більші кроки без порушення стійкості схеми. Таким чином, ММД потребує дуже великої кількості кроків за часом для обчислення траєкторій руху системи атомів / молекул навіть протягом 1 с.

1.6. Алгоритм Бімана

У 1973-1976 рр. був розроблений і опублікований метод чисельного інтегрування ЗДР 2-го порядку, зокрема рівнянь руху Ньютона, спеціально для цілей ММД. Він належить до класу неявних (предиктор-коректор) багатокрокових методів, називається алгоритмом Бімана і є варіантом алгоритму Верле. Спочатку обчислюється новий вектор координат (предиктор)

$$\vec{r}(t + \Delta t) = \vec{r}(t) + \vec{v}(t)\Delta t + \frac{1}{6}(4\vec{a}(t) - \vec{a}(t - \Delta t))\Delta t^2 + O(\Delta t^4), \quad (13)$$

а потім обчислюється вектор швидкості (коректор)

$$\vec{v}(t + \Delta t) = \vec{v}(t) + \frac{1}{6}(2\vec{a}(t + \Delta t) + 5\vec{a}(t) - \vec{a}(t - \Delta t))\Delta t + O(\Delta t^3), \quad (14)$$

причому крок коректора необхідно повторити трохи більше двох разів, а потім Біман запропонував оновити значення для швидкості за формулою

$$\vec{v}(t + \Delta t) = \vec{v}(t) + \frac{1}{12}(5\vec{a}(t + \Delta t) + 8\vec{a}(t) - \vec{a}(t - \Delta t))\Delta t + O(\Delta t^3), \quad (15)$$

яка відповідає методу Адамса-Моултона другого порядку.

У задачах, де сили є функцією не тільки координат, але й швидкостей руху (наприклад, сили опору руху тіла в рідині і газах), швидкості спочатку прогноуються за формулою (14), а потім розраховуються сили, після чого обчислюються скориговані значення швидкостей за (15) з новими силами (тобто, прискореннями).

1.7. Алгоритм leapfrog

Популярний аналог алгоритму Верле є *інтегратор чехарди (leapfrog integrator)*. При цьому підході обчислення швидкостей проводиться на півкрокових відстанях часу, а обчислення координат – на повних кроках:

$$\begin{aligned} \vec{v}(t + \Delta t / 2) &= \vec{v}(t - \Delta t / 2) + \vec{F}(t)\Delta t / m, \\ \vec{r}(t + \Delta t) &= \vec{r}(t) + \vec{v}(t + \Delta t / 2)\Delta t. \end{aligned} \quad (16)$$

Алгоритм (16) можна використовувати зі змінним кроком за часом. Замість (16) можна записати алгоритм для постійного цілого кроку у вигляді:

$$\begin{aligned} \vec{v}(t + \Delta t) &= \vec{v}(t) + \frac{\vec{F}(t) + \vec{F}(t + \Delta t)}{2m}\Delta t, \\ \vec{r}(t + \Delta t) &= \vec{r}(t) + \vec{v}(t)\Delta t + \frac{\vec{F}(t)}{2m}\Delta t^2. \end{aligned} \quad (17)$$

Інша популярна форма (17) – схема удар-дрейф-удар (hit-drift-hit):

$$\begin{aligned} \vec{v}(t + \Delta t / 2) &= \vec{v}(t) + \vec{F}(t)\Delta t / 2m, \\ \vec{r}(t + \Delta t) &= \vec{r}(t) + \vec{v}(t + \Delta t / 2)\Delta t, \\ \vec{v}(t + \Delta t) &= \vec{v}(t + \Delta t / 2) + \vec{F}(t + \Delta t)\Delta t / 2m, \end{aligned} \quad (18)$$

яку можна використати і зі змінним кроком.

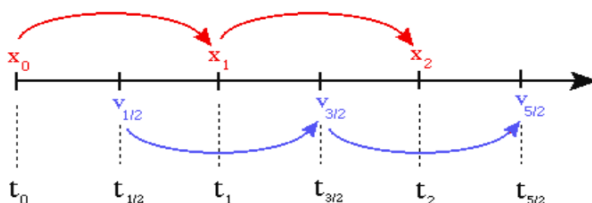


Рис. 6. Схема стрибків для обчислень за алгоритмом leapfrog (чехарда)

Таким чином, схеми інтегрування leapfrog еквівалентні оновленню значень швидкості і координат точок в моменти часу, що чергуються, і розташовані в шаховому порядку таким чином, що вони «перескакують» один через одного, як під час дитячої гри в «чехарду» (рис.6). Цей метод потребує тієї ж кількості операцій, що і метод Ейлера, однак, на відміну від останнього, він має другий порядок точності. Схема leapfrog також стійка для коливального типу рухів частинок, якщо $\Delta t = \text{const}$ і $\Delta t \leq 2/\omega$. Крім того, вона дозволяє проводити розрахунки з від'ємним кроком, тобто розраховувати траєкторії системи у минулому.

За необхідністю, глобальна точність інтегратора (18) може бути підвищена до 4-го порядку, якщо використати підхід⁷ з використанням 4-ч проміжних кроків:

$$\begin{aligned} \bar{\mathbf{r}}^{(1)}(t) &= \bar{\mathbf{r}}(t) + \frac{\bar{\mathbf{v}}(t)\Delta t}{2(2-\sqrt[3]{2})}, & \bar{\mathbf{v}}^{(1)}(t) &= \bar{\mathbf{v}}(t) + \frac{\bar{\mathbf{F}}^{(1)}(t)\Delta t}{(2-\sqrt[3]{2})m}, \\ \bar{\mathbf{r}}^{(2)}(t) &= \bar{\mathbf{r}}^{(1)}(t) + \frac{(1-\sqrt[3]{2})\bar{\mathbf{v}}^{(1)}(t)\Delta t}{2(2-\sqrt[3]{2})}, & \bar{\mathbf{v}}^{(2)}(t) &= \bar{\mathbf{v}}^{(1)}(t) - \frac{\sqrt[3]{2}\bar{\mathbf{F}}^{(2)}(t)\Delta t}{(2-\sqrt[3]{2})m}, \\ \bar{\mathbf{r}}^{(3)}(t) &= \bar{\mathbf{r}}^{(2)}(t) + \frac{(1-\sqrt[3]{2})\bar{\mathbf{v}}^{(2)}(t)\Delta t}{2(2-\sqrt[3]{2})}, & \bar{\mathbf{v}}^{(3)}(t) &= \bar{\mathbf{v}}^{(2)}(t) + \frac{\bar{\mathbf{F}}^{(3)}(t)\Delta t}{(2-\sqrt[3]{2})m}, \\ \bar{\mathbf{r}}(t+\Delta t) &= \bar{\mathbf{r}}^{(3)}(t) + \frac{\bar{\mathbf{v}}^{(3)}(t)\Delta t}{2(2-\sqrt[3]{2})}, & \bar{\mathbf{v}}(t+\Delta t) &= \bar{\mathbf{v}}^{(3)}(t), \end{aligned} \quad (19)$$

де верхні індекси $(.)^{(1),(2),(3)}$ відносяться до послідовних проміжних значень часу в діапазоні між t і $t+\Delta t$; таким чином, ці чотири кроки не обов'язково дорівнюють $\Delta t/4$, але в сумі повинні давати повний крок Δt .

Для порівняння загальна похибка обчислень за часом для методів leapfrog, leapfrog з додатковими проміжними кроками (mid-points) і популярного 4-точкового методу Рунге-Кутта RK4 наведені на рис. 7. Коливання похибки з кожним кроком відповідають різній точності обчислень координат і швидкості, як вже зазначалося вище. За наявності проміжних точок глобальні похибки обчислень як координати, так і швидкості, монотонно зменшуються з кожним кроком, на відміну від стандартного алгоритму обчислень leapfrog без проміжних кроків. Алгоритм RK4 має достатньо високу точність з меншим коливаннями похибки завдяки відсутності окремих розрахунків швидкості, але він часто демонструє значний дрефт енергії.

Дрейф енергії також може виникати внаслідок недосконалості оцінки функції енергії, коли в чисельній схемі жертвують точністю заради швидкості обчислень. Наприклад, коли для оцінки сил парних взаємодій між

⁷ Kinoshita H., Yoshida H., Nakai H. Symplectic integrators and their application to dynamical astronomy. Celestial Mechanics and Dynamical Astronomy. 1990. Vol. 50. P. 59–71.

часинками икористовують схеми відсікання (вплив частинок, які знаходяться достатньо далеко від розглядуваної частинки, не враховується, тобто частинки, які не входять до деякої «блинної сфери» спливу, відсікаються; див. нижче), які вносять систематичні помилки в енергію з кожним кроком інтегрування за часом, оскільки частинки переміщуються вперед і назад вздовж радіусу сфери спливу, якщо не використовується достатнє згладжування.

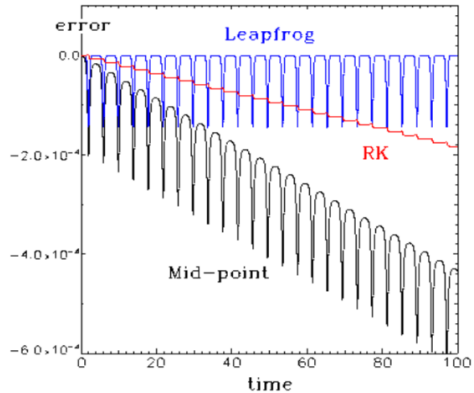


Рис. 7. Загальна похибка (error) обчислень з кроками а часом (time) для методів Leapfrog, Mid-point і Рунге-Кутти RK4

1.8. Системи з обмеженнями

Алгоритм Верле є також дуже зручним для моделювання динаміки зв'язаних частинок або твердих тіл. В таких системах відстані між деякими парами частинок вважаються незмінними, так що замість розв'язання системи ЗДР (1) з початковими умовами (2) для вільної системи частинок (N тіл), розв'язується система диференціально-алгебраїчних рівнянь (ДАР) (1), (2), (20), де

$$g_p(\vec{r}_j) = 0, \quad p = 1, 2, \dots, m \quad (20)$$

– система геометричних і кінематичних обмежень для частинок системи, наприклад, збереження відстаней або двограних кутів, знаходження на поверхні тіла або інші.

Розрахунки за (1), (2) за різними алгоритмами Верле можуть давати значення координат $\{\vec{r}_j\}_{j=1}^N$, які задовольняють (20) з похибками, які вищі, ніж похибка алгоритму. Система вигляду (1), (20) вперше розглядає Лагранж у своїй праці «Аналітична механіка» (1788 р.), де він запропонував загальні методи розв'язання ДАС. На цих методах та їх подальших модифікаціях базуються сучасні алгоритми обмежень ММД, які дуже важливі, наприклад, в обчислювальній хімії.

Ці алгоритми включають:

1) вибір нових (*внутрішніх*) координат, які автоматично задовольняють геометричні обмеження, такі як довжина зв'язку, кут зв'язку та кут кручення між частинками. Внутрішні координати вводяться таким чином, щоб рівняння (20) виконувалися автоматично. Наприклад, якщо в молекулі води H_2O атоми водню утворюють з атомом кисню сталу структуру у вигляді трикутника НОН з кутом 104.45° при вершині О, і при відхиленні від цього значення кута вини-

кають сили, що намагаються повернути атоми у стабільне положення, то замість системи із 3 атомів (9 координат в просторі) можна ввести «центр мас» молекули або одну з вершин (3 координати) і вектор «осі» трикутника або однієї зі сторін (довжина і 2 кути в просторі) і, таким чином, відповідне обмеження в (20) приведе до зниження числа ЗДР в (1) на 3 (оскільки задані 3 умови зв'язку атомів НОН в молекулі H_2O – три сторони трикутника, або дві сторони і кут). Проблема полягає в тому, що ЗДР рівняння для нових внутрішніх координат будуть відрізнятися за формою від рівнянь руху (1), можуть стати достатньо громіздкими і прийняти неконсервативну форму, яка викличе значний шрифт енергії при чисельних розрахунках;

2) визначення *додаткових сил обмежень* у явному вигляді, наприклад, додати штучні сили пружності між парами частинок, якби частинки були зв'язані пружинами з достатньо великою жорсткістю. Цей підхід вводить додаткові явні сили і, таким чином, потребує більше обчислювальної потужності для обчислення траєкторій частинок системи на заданому інтервалі часу. Проблема цього методу полягає і в тому, що цей рух частинок задовольняє цим силам з похибками, а не в точний спосіб як в 1), що не забезпечує точного виконання відповідного обмеження в (20). Крім того, якщо введені достатньо сильні додаткові сили відштовхування / притягання, то для інтегрування (1) будуть потрібні дуже короткі кроки за часом, що робить алгоритм неефективним у відношенні обсягу обчислень;

3) неявно мінімізувати сили обмежень *за допомогою методу множників Лагранжа* або методів проєкцій. Цей підхід виявився дуже ефективним і використовується у більшості моделей ММД з обмеженнями. Наприклад, якщо в (20) задані постійні відстані між деякими точками, можна ввести в (1) додаткові нульові «сили»

$$\phi_k(t) = |\vec{r}_{kr}(t) - \vec{r}_{ks}(t)|^2 - d_k^2 = 0, \quad (21)$$

де $k=1,2,\dots,q$ – типи обмежень, r, s – номери частинок, які підпадають до k -го обмеження, наприклад, всі атоми водню у молекулах води, для яких відстані О-Н і Н-Н в кожній з молекул мають бути постійними, але різними величинами d_{O-H} і d_{H-H} .

Підстановка (21) до рівнянь (1) з потенційними силами дає

$$m_j \frac{d^2 \vec{r}_j(t)}{dt^2} = -\nabla \left(\Phi(\vec{r}_j(t)) - \sum_{k=1}^q \lambda_k \phi_k(t) \right), \quad i=1,2,\dots,N, \quad (22)$$

де в останній доданок в правих частинах (22) може входити значення $\vec{r}_j(t)$, а λ_k – множники Лагранжа.

Додавання до (21) обмежувальних сил (20) не змінює загальну енергію системи частинок, оскільки повна робота, що виконується цими силами над повним набором частинок, дорівнює нулю, тому що сили в сумах будуть мати протилежні знаки при однакових переміщеннях (сили дії і протидії, відповідно до 3-го закону Ньютона).

Двічі інтегруємо (22) за часом і отримуємо для координат

$$\vec{r}_j(t + \Delta t) = \vec{r}_j^{\text{num}}(t + \Delta t) + \sum_{k=1}^q \frac{\lambda_k}{m_j} \frac{\partial \phi_k}{\partial \vec{r}_j} \Delta t^2, \quad i=1,2,\dots,N. \quad (23)$$

де $\vec{r}_j^{\text{num}}(t + \Delta t)$ – невідкореговані координати, які отримані шляхом інтегрування (1) без «нульових» сил.

Обмеження (16) повинні виконуватися і для координат (23) на новому кроці за часом, тобто

$$\phi_k(t + \Delta t) = |\vec{r}_{kr}(t + \Delta t) - \vec{r}_{ks}(t + \Delta t)|^2 - d_k^2 = 0. \quad (24)$$

Підстановка (23) до (24) дає систему з N лінійних алгебраїчних рівнянь для визначення N множників Лагранжа λ_k

$$\left| \vec{r}_r^{\text{num}}(t + \Delta t) - \vec{r}_s^{\text{num}}(t + \Delta t) + \sum_{k=1}^q \left(\frac{\partial \phi_k}{\partial \vec{r}_r} - \frac{\partial \phi_k}{\partial \vec{r}_s} \right) \frac{\lambda_k}{m_j} \Delta t^2 - d_k^2 \right| = 0, \quad (25)$$

розв'язок якої легко знайти методом Ньютона-Рафсона.

В силу того, що обмеження відстаней накладені не на всі пари частинок в системі, якобіан $J_{ik} \equiv \frac{\partial \phi_i}{\partial \lambda_k}$ задачі (25) є блочною матрицею з багатьма

нулями. Для ітерацій як початкове значення вектору невідомих $\vec{\lambda}$ можна взяти нульове значення $\vec{\lambda} = (0, 0, \dots, 0)$. Отримані значення $\lambda_j = J_{jk}^{-1} \phi_k$ задають мінімальну похибку на обмеження (20), накладену штучними силами (21).

Були розроблені високоефективні алгоритми розв'язання задачі мінімізації сили обмежень за допомогою методу множників Лагранжа, такі як SETTLE, SHAKE, M_SHAKE, SHAPE, LINKS та інші.

Порівняно з 2), підходи 1) і 3) є більш ефективними у розумінні витрат часу і комп'ютерної потужності. Ефективні алгоритми обмежень, наприклад, нехтують рухом частинок за деякими ступенями свободи (трансляційними або обертальними). Наприклад, в ММД атомних систем (гази, кристали, метали, сплави) довжин ковалентних зв'язків (засновані на поєднанні зовнішніх електронних хмар атомів) зазвичай обмежують. Але цей алгоритм не слід використовувати, якщо коливання за цими ступенями свободи важливі для фізичного явища, яке вивчається.

1.9. Інші інтегратори

Крім зазначених вище (пп. 1.4 – 1.8), в ММД використовуються й інші спеціальні інтегратори і підходи, залежно від типу прикладної задачі, наприклад як *неявна модель розчинника з екранованими кулонівськими потенціалами* (screened Coulomb potentials implicit solvent model, SCP-ISM) – це континуальне наближення динаміки молекул розчинника для використання в комп'ютерному моделюванні біологічних макромолекул, таких як білки і нуклеїнові кислоти. Метод заснований на класичній теорії полярних рідин,

яка була розроблена П. Дебаєм і пізніше скоригована Л. Онзагером. Модель дозволяє додавання квантово-хімічних взаємодій для формального отримання континуальної моделі розчинника, придатної для ефективного комп'ютерного моделювання як малих (за чисельністю молекул), так і великих систем частинок.

1.10. Потенціали в системах частинок

Моделювання за допомогою ММД потребує визначення потенційної функції $\Phi(t) = \sum_{j=1}^N \Phi_j(t)$ в (1) або опису умов, за допомогою яких частинки

в системі будуть взаємодіяти. У біології і хімії цю взаємодію називають силовим полем (емпіричні потенціали), а у фізиці і матеріалознавстві – між атомним потенціалом або потенціалом міжмолекулярної взаємодії. Оскільки рівняння (1) відповідають законам класичної механіки макроскопічних тіл, квантовий опис не розглядається, і для цього використовуються загальноприйняті наближення. **Наближення Борна–Опенгеймера:**

- 1) рух електронів настільки швидкий, що можна вважати, що вони миттєво реагують на рух своїх ядер, внаслідок чого їх можна розглядати окремо;
- 2) ядра атомів набагато важчі за електрони і при однаковому імпульсі електрони будуть рухатися набагато швидше ніж ядра, і тому ядра атомів можна моделювати як точкові частинки, рух яких відповідає класичній ньютонівській динаміці. При цьому вплив електронів наближено описується як поверхня потенційної енергії, яка зазвичай представляє основний стан системи;
- 3) загальна енергія молекули E_{mol} є адитивна функція, яка має вигляд $E_{\text{mol}} = E_{\text{elec}} + E_{\text{vibr}} + E_{\text{rota}} + E_{\text{spin}}$, де E_{elec} – кінетична енергія руху, міжелектронного відштовхування, міжядерного відштовхування та електронно-ядерного тяжіння, E_{vibr} і E_{rota} – енергії вібраційного і обертального (rotations) рухів, E_{spin} – енергія ядерного спіну (мала величина).

Такий підхід не враховує хімічних або біохімічних реакцій, які протікають в системах атомів, молекул, молекулярних структур і клітин. Тому при моделюванні хімічної кінетики ММД використовуються емпіричні силові поля, які складаються із підсумовування зв'язаних сил, пов'язаних із хімічними зв'язками між атомами, кутами зв'язку (як в плоских молекулах H_2 , C_6P_6) та двограними кутами зв'язку (між площинами в 3D), міжмолекулярними силами Ван-дер-Ваальса та електростатичним зарядом. Емпіричні потенціали – це спеціальні функції, які є наближеннями квантово-механічних ефектів. Ці потенціали містять вільні параметри, такі як заряд ядра, радіус атома, коефіцієнти Ван-дер-Ваальса та інші (квантово-хімічне моделювання). Використовують також експериментально виміряні фізичні параметри, такі як параметри решітки, коефіцієнти пружності, спектроскопічні властивості, рентгенограми та ін.

Походження сил, які відповідальні за притягання атомів один до одного, було пояснено в 1930 р. німецьким фізиком-теоретиком Ф. Лондоном (London dispersion forces). Міжатомне притягання виникає внаслідок флуктуацій електричних зарядів у парах атомів, які знаходяться близько один від одного. При цьому нейтральний атом (рис. 8а) отримує миттєвий електричний дипольний момент (рис. 8б), який наводить протилежно спрямований диполь у сусідньому атомі (рис. 8в). Ці диполі притягуються один до одного за рахунок електростатичного притягання. Міжатомна взаємодія проявляється для будь-яких атомів, які утворюють рідини, гази і тверді тіла, наноматеріали, такі як графен і фулерени, атомних і молекулярних кластерів і т.д.

Потенціал такої взаємодії має вигляд $\Phi_L(r) = -C/r^6$, де C – постійна для даної пари атомів, r – відстань між центрами атомів.

Серед існуючих моделей потенціалів в ММД найчастіше використовується **потенціал Ленарда-Джонса** (Lennard-Jones, LJ, 1924 р.) у загальному вигляді

$$\Phi_L(r) = -\frac{C}{r^6} \left(1 - \frac{r_m^6}{2r^6} \right), \quad (26)$$

де r_m – значення відстані r , при якому система знаходиться в стані стійкої рівноваги, $C = \text{const}$, яка знаходиться з квантомеханічної теорії.

Відомо, що у полі потенційних сил робота A_Φ поля по переміщенню частинки з точки А до точки В може бути легко обчислена як

$$A_\Phi|_A^B = \Phi(\vec{r}_B) - \Phi(\vec{r}_A). \quad (27)$$

Найчастіше з урахуванням виразу для константи (26) записується у вигляді

$$\Phi_L(r) = 4\varepsilon \left(\left(\frac{r_0}{r} \right)^{12} - \left(\frac{r_0}{r} \right)^6 \right), \quad (28)$$

де ε – значення мінімуму функції $\Phi_L(r)$ (рис. 9), r_0 – відстань, на якій потенціал дорівнює нулю $\Phi_L(r_0)$, перший доданок відповідає силам відштовхування за рахунок перекриття електронних хмар атомів на малих відстанях (відштовхування Паулі), а другий – силам притягання частинок за рахунок диполь-дипольної взаємодії.

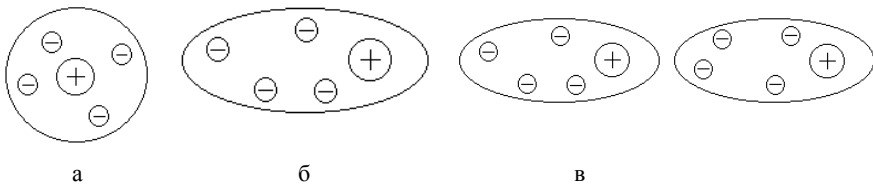
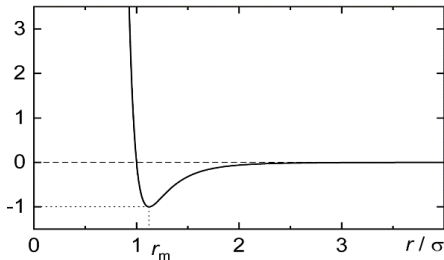


Рис. 8. Нейтральний атом (а), поляризований (б) і парна взаємодія (в)

Рис. 9. Графік функції $\Phi_{LJ}(r)$ (26)

LJ описує стабільний стан системи атомів, які не можуть ані далеко розійтись, ані занадто зблизитися. Цей потенціал є кількісно точним тільки для інертних газів, але він також широко використовується для якісних досліджень і в системах, де важливі дипольні взаємодії, особливо в силових хімічних полях для опису міжмолекулярних взаємодій.

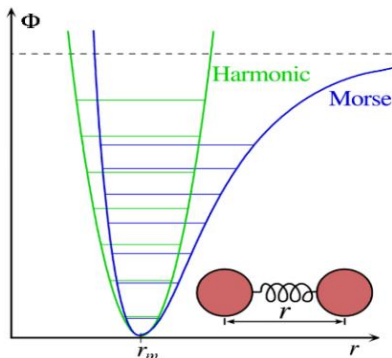
Потенціал Морзе (Морса, Ph. Morse, 1929 р.) описує потенціальну енергію міжатомної взаємодії для визначення коливальної структури двохатомної молекули і записується у вигляді (Рис.10)

$$\Phi_M(r) = \varepsilon(1 - \exp(-a(r - r_m)))^2, \quad (29)$$

де r , ε і r_m мають той самий сенс, параметр a визначає ширину потенційної ями і залежить від коефіцієнту k_e сил взаємодії на «дні» ями як $a = \sqrt{k_e / 2\varepsilon}$.

У випадку взаємодії частинки з твердою поверхнею (29) записують у вигляді

$$\Phi_M(r) = \varepsilon(1 + \exp(-2a(r - r_m)) - 2\exp(-a(r - r_m))), \quad (30)$$

Рис. 10. Графік функції $\Phi_M(r)$ (29) порівняно з гармонійним осцилятором

Потенціал $\Phi_{LJ}(r)$ не включає електростатичні сили і, таким чином, описує системи молекул або електронейтральних атомів. Легко обчислити, що функція $\Phi_{LJ}(r)$ (26) має єдиний мінімум $\Phi_{LJ} = \varepsilon$ (рис. 9) при $r_m = \sqrt[6]{2}r_0$, який відповідає стійкому рівноважному стану (потенціальна яма). Таким чином, потенціал

де перший доданок в правій частині відповідає короткодіючим силам відштовхування, а другий – дальгодіючим силам притягання, аналогічно до потенціала LJ.

Легко обчислити, що $\Phi_M(\infty) = \varepsilon$, мінімальне значення потенціалу $\Phi_M(r_m) = 0$. Потенціал Морзе краще описує енергію і вібраційний спектр двохатомних молекул, ніж класичний гармонічний осцилятор з симетричною потенційною функцією (рис. 10). Слід зауважити, що потенціал обчислюється відносно заданого значення (на

поверхні або нескінченності), тобто з точністю до константи, і тому в формулах (29), (30) постійними можна знехтувати.

Потенціал Букінгема (R. Buckingham) був запропонований ним в 1938 р. для спрощення чисельних розрахунків як модифікація потенціалу LJ, в якому сили відштовхування замінені на експоненту, як в потенціалі Морзе:

$$\Phi_B(r) = A \left(\exp(-Br) - \frac{C}{r^6} \right), \quad (31)$$

де A, B, C = постійні, які залежать від типу частинок, що взаємодіють (частинки газу, рідини, твердого матеріалу); B^{-1} має сенс ефективного радіуса взаємодії.

Наприклад, для газів при деяких типах зіткнень частинок (теорія Чепмена-Коулінга) модифікований потенціал Морзе (M_m) записується як

$$\Phi_{B_m}(r) = \frac{\varepsilon}{1-6/b} \left(\frac{6}{b} \exp \left[b \left(1 - \frac{r}{r_m} \right) \right] - \left(\frac{r_m}{r} \right)^6 \right), \quad (32)$$

де параметр b відповідає швидкості зростання енергії відштовхування при наближенні частинок.

Важливою модифікацією (31), (32) є потенціал Букінгема-Кулона

$$\Phi_{B_C}(r) = A \exp(-Br) - \frac{C}{r^6} + \Phi_C(r), \quad \Phi_C(r) = \frac{q_1 q_2}{4\pi \varepsilon_0 r}, \quad (33)$$

де Φ_C – потенціал Кулона (Ch. Coulomb) парної взаємодії частинок з ефективними зарядами q_1, q_2 , а ε_0 – діелектрична константа.

Оскільки при $r \rightarrow 0$ експонента сходиться, а функція з від'ємним ступенем розходиться, $\Phi_{B,B_m,B_C}(0) = -\infty$ (рис. 11) і потенціали мають фізичний сенс тільки при деяких достатньо великих значеннях $r > r^*$.

Таким чином, для частинок (атомів), які не утворюють хімічних зв'язків, в ММД використовують потенціали Ленард-Джонса, Морзе, Букінгема та їх модифікації. Це потенціали парних центральних взаємодій, тобто сили вважаються направленими між центрами відповідних частинок. При наявності хімічних зв'язків, використовуються більш складні нецентральні потенціали, які стають все більш популярними у зв'язку з моделюванням руху наночастинок в наноматеріалах, наприклад, атомів вуглецю в графені, нанотрубках, фулеренах та ін. (потенціал Бреннера (D.W. Brenner, 1990 р.), Терсофа (J. Tersoff, 1988 р.) та ін.).

Парні потенціали мають нескінченну область впливу ($r \rightarrow \infty$), однак на достатньо великих відстанях вони можуть не перевищувати похибки чисельного алгоритму і, таким чином, взаємодіями достатньо віддалених частинок можна знехтувати. Легко оцінити, що в системі з N частинок число пар

дорівнює $C_N^2 = \frac{N!}{(N-2)!2!} = \frac{N(N-1)}{2} \sim N^2$, а число потенціалів взаємодії значно

зростає з розміром системи. Якщо вести критичну відстань взаємодій r_c (відстань відсічки, cut-off) і враховувати тільки парні взаємодії між частинкою та

її сусідами на відстанях $r < r_c$, можна суттєво прискорити чисельні розрахунки без істотного погіршення точності, тому що такі сили значно зменшуються з квадратом відстані.

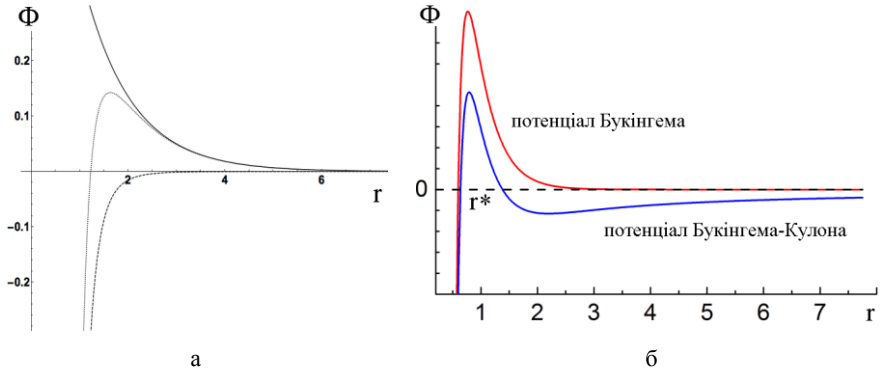


Рис. 11. Графік функції $\Phi_B(r)$ (31): а – зі складовими компонентами (експонента – суцільна, степенева – штрихова, сума – точкова лінії); б – порівняння потенціалів В і ВС

Потенціал Борна-Майєра-Хуггінса (Born-Meyer-Huggins potential) взаємодії пари частинок має вигляд

$$\Phi_{\text{ВМН}} = A \exp(B(r_0 - r)) - \frac{C}{r^6} - \frac{D}{r^8},$$

де r_0 – відстань, на якій $\Phi_{\text{ВМН}}=0$; перша складова відповідає відштовхуванню, друга і третя – диполь-дипольному і диполь-квадрупольному притягінню.

Комбіновані потенціали використовуються як лінійні комбінації експоненціальних і ступеневих у вигляді емпіричних залежностей, які відповідають силам тяжіння на великих відстанях між частинками і силам відштовхування – на малих. Наприклад, в класичній роботі [9] були використані комбінації потенціалів Морса і Борна-Майєра у вигляді

$$\Phi(r) = \begin{cases} 8573 \frac{0.7}{r} \exp(-6.547r), & r < 0.7, \\ 8573 \exp(-6.547r), & 0.7 < r < 1.35, \\ 0.223(\exp(-2a(r-r_0)) - 2 \exp(-a(r-r_0))) & 1.35 < r < 2, \end{cases}$$

де $a=1.3885 \cdot 10^{-10} \text{ м}$, $r_0=2.845 \cdot 10^{-10} \text{ м}$.

Сили Ван-дер-Ваальса (Van der Waals, VdW), або міжмолекулярні сили були відкриті голандським фізиком Й.Д. Ван дер Ваальсом у 1869 р. для опису поведінки молекул газу і рідини. Відоме рівняння стану реального газу має вигляд

$$\left(p + \frac{av}{V^2}\right)(V - bv) = vRT, \quad (34)$$

де p , V , T , v – тиск, об'єм, температура і кількість молів газу, R – газова постійна, поправка a враховує сили міжмолекулярного притягання, b – загальний об'єм молекул газу; при $a=0$, $b=0$ (35) дає рівняння стану ідеального гару (рівняння Клапейрона).

Ван дер Ваальс показав, що при деякій температурі (критична температура) зникають відмінності у фізичних властивостях рідини та її пари, які перебувають у рівновазі. За такої температури густина рідини та її насиченої пари стають однаковими, і видима межа між ними зникає. За роботи з молекулярної фізики він отримав в 1910 р. Нобелівську премію з фізики «За роботу, присвячену рівнянню стану газів і рідин».

Сили Ван-дер-Ваальса складаються з 3 типів слабких електромагнітних взаємодій:

- 1) орієнтаційні сили (диполь-дипольне притягання) – здійснюється між молекулами, які є постійними (не наведеними) диполями (рис. 7б); $\Phi_{d-d}(r) \sim r^{-3}$, де r – відстань між «центрами» диполів;
- 2) дисперсійне притягання (лондонівські сили) – зумовлені взаємодією між миттєвим та наведеним диполем; $\Phi_L(r) \sim r^{-6}$;
- 3) індукційне притягання (поляризаційне) – зумовлені взаємодією між постійним диполем та наведеним (індукованим); $\Phi_{d-i}(r) \sim r^{-6}$.

Теорія Дебая–Гюккеля для повністю іонізованого газу, який вважається в цілому електронейтральним, а заряди протилежно заряджених іонів розглядаються як точкові заряди. Кожний іон має «хмару» із протилежно заряджених частинок. Коли частинки зближаються, виникають сили відштовхання з потенціалом

$$\Phi_{D-H}(r) = 32\pi\kappa_B T d_p \rho_i \kappa^{-2} \text{th}^2 \left(\frac{\rho_i \Phi_0}{4\kappa_B T} \right) e^{-\kappa r}, \quad (35)$$

де d_p – діаметр частинки, κ – постійна Дебая, ρ_i – густина електричного заряду, Φ_0 – потенціал подвійного електричного шару на поверхні частинки (молекули, вірусу, клітини), який визначає її дзета-потенціал $\zeta_p = 4\pi\mu_{bf} v_p / \epsilon_{bf}$, де ϵ_{bf} – діелектрична проникність.

Кулонівські сили електростатичної взаємодії з потенціалом (33)

$$\vec{f}_e = -\nabla\Phi_C = \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}^2} \frac{\vec{r}_{ij}}{|\vec{r}_{ij}|}, \quad (36)$$

де q_i, q_j – заряди частинок, $\vec{r}_{ij} = \vec{r}_j - \vec{r}_i$, \vec{r}_i, \vec{r}_j – радіус-вектори частинок в деякій фіксованій системі координат.

Таким чином, Кулонівські сили спадають з відстанню між молекулами значно повільніше ($f_e \sim r^{-2}$) порівняно з силами ван-дер-Ваальса ($f_{vdW} \sim r^{-7}$).

Броунівські сили пов'язані з хаотичними зіткненнями частинок, які знаходяться в стані молекулярно-теплого руху, інтенсивність якого зростає з температурою

$$\bar{f}_B = \sqrt{\frac{2k_B T}{3\pi^3 \mu_{bf} d_p}} m_p \bar{f}_0, \quad (37)$$

де \bar{f}_0 – випадковий вектор з нульовим середнім.

Хімічні зв'язки між атомами і молекулами виникають у хімічних сполуках під час хімічних реакцій, які, як відомо, можуть протікати в обидві сторони як з утворенням, так і з розпадом нових сполучень. Серед хімічних зв'язків в ММД виділяють **іонні** (рис. 12а), **водневі** (рис. 12б), **металеві** (рис. 12в) і **ковалентні** (рис. 12г) зв'язки. Коли атом металу втрачає частину електронів з найбільш віддаленої орбіти, які притягуються атомом неметалу,

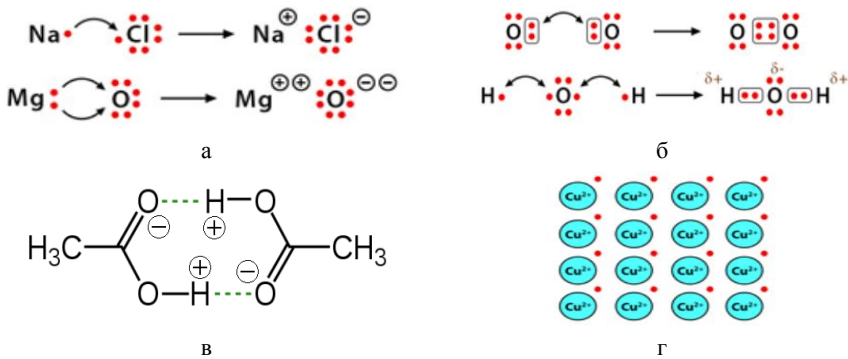


Рис. 12. Типи хімічних зв'язків між атомами і молекулами: іонні (а), ковалентні (б), водневі (в) і металеві (г)

так що два нейтральні до того атоми стають іонами і притягуються, утворюється іонний зв'язок (рис. 12а). Коли два атоми неметалів мають спільні електрони, утворюється ковалентний хімічний зв'язок (рис. 12б). Він може утворювати як неполярні, тобто електронейтральні (O₂), так і полярні молекули, які являють собою електричний диполь за рахунок розведення позитивно зарядженого «центра мас» і негативно зарядженої хмари електронів (H₂O) (рис. 12б). Коли хімічний зв'язок заснований на притяжінні атомаводню (позитивний іон у сполуках, тому що має єдиний слабо зв'язаний електрон на не повністю заповненій орбіталі) до негативно зарядженого іону, утворюється водневий зв'язок. Наприклад, такі зв'язки виникають між молекулами метилендіоксиду CH₄O₂ (рис. 12в). У сукупностях молекул металів позитивно заряджені іони металів занурені у хмару електронів, які можуть рухатися хаотично, а за

наявності зовнішньої різниці електричних потенціалів на металі – рухаються в напрямку катода і у металевому провіднику (дроті) виникає електричний струм. Взаємодія іонів металу у електронній хмарі відповідає металевому зв'язку (рис. 12г). Найчастіше в силових полях ММД використовуються ковалентні зв'язки, які мають кілька ступенів свободи, і у полі зовнішніх сил можуть випробувати різні механічні рухи.

Ковалентні зв'язки – це хімічні зв'язки, які засновані на об'єднанні електронних хмар атомів і утворенні стійкого балансу сил тяжіння та відштовхування між атомами. Якщо в системі з'являються флуктуації положення частинок, виникають відповідні сили, які моделюються як пружні сили зв'язків атома з сусідніми зі спільними електронами (рис. 13а). Три складові ковалентних зв'язків включають сили, які виникають через розтягування «пружин» від рівноважного значення b_0 до поточного b (рис. 13б), зміни кута θ між пружинами (рис. 13в) і кута закручування ϕ пружини (рис. 13г,д)

$$\Phi_{cov} = \sum_{bonds} k_b (b - b_0)^2 + \sum_{angles} k_\theta (\theta - \theta_0)^2 + \sum_{torsions} k_\phi (\cos(n\phi + \delta) + 1), \quad (38)$$

де k_b, k_θ, k_ϕ – модулі жорсткості відповідних змін положень атомів.

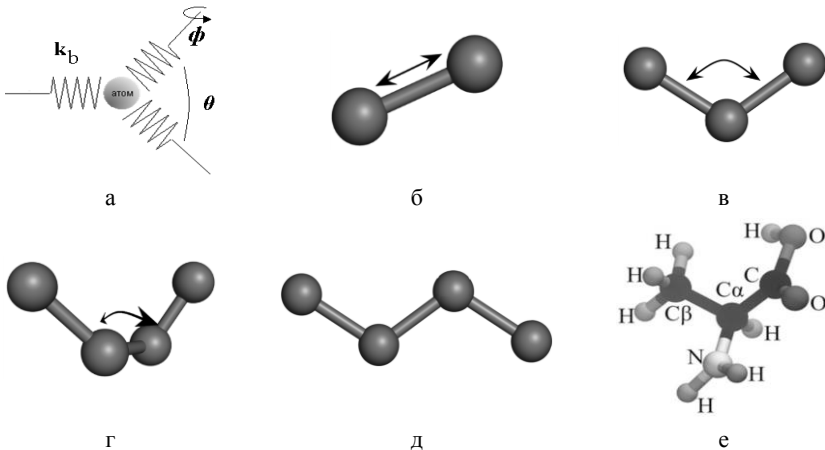


Рис. 13. Спільні електрони в молекулі CH_4 (а), ковалентні зв'язки (б), розтягування зв'язку (в), зміни двогранного кута (г), закручування відносно спільної осі (д) і в площині (е); приклад молекули аланіна (див. текст) (е); (ж)

Зазвичай в молекулах присутні всі типи сил, які діють на зв'язки. Наприклад, в молекулі аланіна $C_3H_7NO_2$, яка складається з 13 атомів (рис. 13е), присутні 12 зв'язків, які розтягуються (чотири C-H, дві C-C, дві N-H, дві C-O, одна C-N, одна O-H), 17 зв'язків зі змінами величини плоского кута (C-O-H, O-C-O, O-C-C, C-C-H, C-C-N, C-C-C, C-C-N, N-C-H, C-N-H, H-N-H), 17 зв'язків із закручуванням (H-O-C-O, H-O-C-C, O-C-C-H, O-C-C-C, C-C-C-H, O-C-C-N, C-C-N-H, O-C-C-N) і 26 зв'язків без обмежень (H-O, H-C, H-N, O-C,

O-N). Взаємодії без зв'язків визначають термодинамічну рівновагу і такі важливі біофізичні процеси, як згортання / розгортання ланцюгів білків, утворення мембрани та міцел, взаємодії білків з ДНК, сольватацію⁸ та ін.

1.11. Статистичні характеристики системи

Оскільки задачею ММД є моделювання статистичних властивостей реальної фізичної системи, результати коректних розрахунків мають задовольняти головним залежностям статистичної механіки. Однією з таких залежностей є розподілення Больцмана, яке дає вірогідність P_j того, що система (сукупність малих частинок) знаходиться у стані з енергією ε_j

$$P_j = P_0 \exp(-\varepsilon_j / k_B T), \quad (39)$$

де k_B – постійна Больцмана, T – температура.

Розподілення Максвела характеризує вірогідність того, що частинка (молекула), яка знаходиться в деякій точці $\vec{r} \in [V, V + dV]$ об'єму простору, має швидкість в інтервалі $\vec{v} \in [\vec{v}, \vec{v} + d\vec{v}]$

$$f(v) = \left(\frac{m}{2\pi k_B T} \right)^{3/2} \exp\left(-\frac{mv^2}{2k_B T} \right). \quad (40)$$

Об'єднане розподілення Максвела–Больцмана можна записати як функцію швидкості і координат

$$dN(v; x, y, z) = \frac{4N_0 v^2}{\sqrt{\pi} v_*^3} \exp\left(-\frac{1}{v_*^2} \left(v^2 + \frac{2\varepsilon(x, y, z)}{m_0} \right) \right) dv dV, \quad (41)$$

де $dN(v; x, y, z)$ – число частинок в об'ємі dV , які мають координати $([x; x + dx], [y; y + dy], [z; z + dz])$ і швидкості $[v; v + dv]$, m_0 – маса частинки, v_* – найбільш вірогідна швидкість, $\varepsilon(x, y, z)$ – потенціальна енергія системи в точці (x, y, z) ; або як функцію імпульсу і координат

$$dP(p_x, p_y, p_z; x, y, z) = \frac{A}{(2\pi m_0 k_B T)^{3/2}} \exp\left(-\frac{p_x^2 + p_y^2 + p_z^2}{2m_0 k_B T} \right) \times \quad (42)$$

$$\times dp_x dp_y dp_z \exp\left(-\frac{\varepsilon(x, y, z)}{k_B T} \right) dx dy dz,$$

де перша експонента відповідає розподіленню Максвела за швидкостями, а друга – розподіленню Больцмана за енергіями (координатами).

На рис. 14 наведені розподілення швидкостей молекул деяких інертних газів. Як завжди, найбільш ймовірні значення в статистичних залежностях

⁸ Сольватація – це електростатична взаємодія між частинками розчиненої речовини та розчинника. Сольватація у водних розчинах називається гідратацією.

представлені максимум відповідної функції щільності вірогідності (v_* у даному випадку).

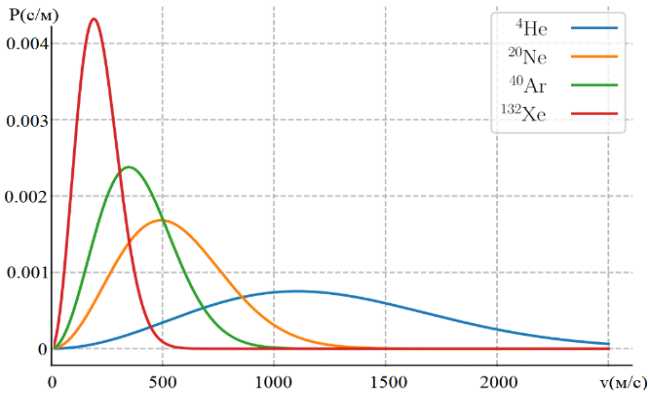


Рис. 14. Розподілення ймовірностей швидкостей частинок для деяких інертних газів

Таким чином, результати розрахунків координат і швидкостей кожної частинки в системі самі собою не становлять інтересу, тому що дають надлишковий з погляду статистичної фізики та термодинаміки опис системи. Метою аналізу чисельних результатів є обчислення фізичних величин, таких як енергія, температура, тиск в системі, тобто ті параметри, які можна виміряти. Це означає, що результати коректних розрахунків ММД повинні давати такі значення спостережуваних (середніх) величин, наприклад енергії, відповідно до деякого розподілу, які характерні для реальної системи. Така система називається статистичним ансамблем. У найбільш природному для постановки задачі ММД випадку в ході моделювання залишаються фіксованими число частинок N , об'єм V і повна енергія E системи. Такий ансамбль (N, V, E) називається **мікроканонічним**. У той же час, практично цікавішим є ансамбль число частинок – об'єм – температура (N, V, T) , який називається **канонічним** (тобто такий, у якому замість повної фіксована лише кінетична енергія молекул). Моделювання **великого канонічного ансамблю** за допомогою ММД спрямоване на використання класичних рівнянь руху для дослідження систем, у яких температура, об'єм і хімічний потенціал (T, V, μ) є постійними, але кількість частинок коливається. У **великому ізобарично-ізоермічному ансамблі** постійними є температура, тиск і хімічний потенціал (T, P, μ) (див. табл. 2). У спеціальних випадках використовують також осмотичний ансамбль (T, P, N, μ) (табл. 2), а на нанорівні використовують відповідні наноканонічні ансамблі, до яких, крім макропараметрів системи (T, P, N, μ) , входять параметри загальної форми і геометричної структури наносистем (шорсткість, фрактальна розмірність, пористість та ін.).

Вибір того чи іншого ансамблю диктується, наприклад, умовами фізичного, хімічного або медико-біологічного експерименту, коли певний набір параметрів системи можна контролювати (наприклад, постійний тиск, постійну температуру, постійну кількість частинок і т.д.). Детальне викладення всіх математичних залежностей і перетворень для кожного зі статистичних ансамблів, а також відповідні алгоритми можна знайти у підручнику [10].

Таблиця 2. Головні типи статистичних ансамблів

№ з/п	Назва	Контрольовані макропараметри
1	Мікроканонічний	N, V, E
2	Канонічний	N, V, T
3	Великий канонічний	T, V, μ
4	Ізотермічно-ізобаричний	T, P, μ
5	Осмотичний	T, P, N, μ

Для моделювання статистичних ансамблів використовуються алгоритми, які називають *термостатами*. Їх завдання – коригування швидкості частинок в такий спосіб, щоб забезпечувати сталість температури. Найбільш грубим методом є термостат Берендсена, який масштабує швидкість частинок на кожному кроці. На даний момент він використовується тільки для стабілізації систем, далеких від стану рівноваги. Найпопулярнішим є термостат Нозе-Хувера (Nose-Hoover), який коригує Лагранжіан системи шляхом додавання до нього фіктивної координати.

Аналогічно вводиться поняття *баростата*, запровадження якого дозволяє моделювати систему число частинок – тиск – температура (NPT). Коригування тиску здійснюється за рахунок варіювання об'єму системи, як алгоритм найпопулярнішим також є баростат Нозе-Хувера (Nose-Hoover), який діє за схожим на термостат алгоритмом.

Таким чином, на кожному кроці розрахунків швидкості методом Верле або іншими проводяться розрахунки температури та тиску за формулами термостату і баростату

$$T = \frac{m_0}{3k_B} \sum_{j=1}^N v_j^2, \quad (43)$$

$$P = \frac{Nk_B T}{V} + \frac{1}{3V} \sum_{j=1}^N \vec{r}_j \vec{f}_j. \quad (44)$$

На сьогодні день існує велика кількість досить досконалих пакетів ММД з готовими бібліотеками силових полів. Окрім описаних вище, існують термостати Нозе-Андерсена (Nosé-Poincaré-Andersen), Нозе-Пуанкаре-Андерсена (Nosé-Andersen) та інші (див. [10]).

1.12. Загальний алгоритм ММД

Таким чином, розв'язання будь-яких прикладних задач з використанням ММД засноване на такому загальному алгоритмі (рис. 15).

1. Визначити фізичні властивості системи частинок і просторову область V їх розташування (2D, 3D, система координат і т.д.).
2. Обґрунтовано задати розміри і число N частинок.
3. За допомогою генератора випадкових величин задати координати $\{x_j(t_0)\}_{j=1}^N$ центрів N частинок з виконанням умов їх попарного неперетинання в 2D- або 3D-просторі.
4. Виходячи з фізичної постановки задачі, задати значення початкових швидкостей $\{\dot{x}_j(t_0)\}_{j=1}^N$ N частинок (можливо, нульових $\{\dot{x}_j(t_0)\}_{j=1}^N = 0$).
5. Обґрунтовано вибрати крок інтегрування за часом Δt .
6. Визначити всі сили взаємодії як функції координат частинок. Якщо сили тільки потенціальні, то визначити потенціали сил. Якщо не потенціальні сили теж враховуються, вибрати наближення для їх обчислення як функцій координат на попередньому кроці і історії (траєкторій руху) динаміки системи.
7. Залежно від поля сил $\vec{F}(\vec{r})|_{t=0}$ визначити вектор прискорення $\vec{a}(\vec{r})|_{t=0}$ для першого кроку інтегрування (можна задати $\vec{a}(\vec{r})|_{t=0} = 0$).



8. Вибрати основний інтегратор (Рунге-Кутта, Штрюмера-Верле, Бімена, leapfrog, ...).
9. Вибрати інтегратор першого кроку за часом (Ейлера, Рунге-Кутта, Верле, ...).
10. Вибрати тип силового поля (алгоритм і критерії відсічі взаємодії частинок).
11. Вибрати тип і алгоритм обмеження на положення частинок.



12. Використання предиктора: прогнозування координат частинок на наступному кроці $\vec{r}(t_{j+1}) = \vec{r}(t_j) + \vec{v}(t_j)\Delta t_j + \vec{a}(t_j)(\Delta t_j)^2 / 2 + \text{more accurate terms}$ для кожної частинки j .
13. Використання коректора: обчислення швидкостей $\vec{v}(t_{j+1}) = \vec{v}(t_j) + \vec{a}(t_j)\Delta t_j + \text{more accurate terms}$.
14. Уточнення координат частинок на наступному кроці з урахуванням коректора залежності від обраного інтегратора.



15. Обчислення потенціалів сил взаємодії для нових значень координат залежно від обраного методу відсічень.



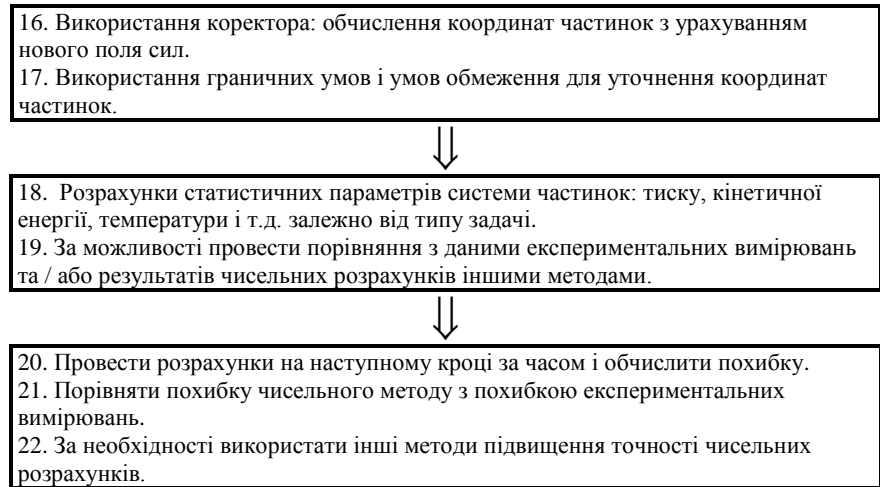


Рис. 15. Загальний алгоритм ММД

1.13. Недоліки і обмеження ММД

На рис. 16а наведений приклад з доповіді CASP8 (2008 р.) порівняння 3D-структури молекули, яка була встановлена за допомогою експериментальних методів та МДД моделюванням за допомогою різних алгоритмів. Значна розбіжність в структурі обумовлена недоліками методу і алгоритмів. Послідовні звіти CASP4 (2000 р.) – CASP10 (2012 р.) показали значне зростання точності ММД як для нескладних за структурою (easy and target difficulty), так і для дуже складних (difficult) молекул (рис. 16б).

Цей прогрес пов'язаний як з вдосконаленням обчислювальної техніки (процесорів, доступної динамічної пам'яті), що дозволяє розраховувати довші та детальніші траєкторії руху частинок, а також з результатами підвищення якості параметрів силового поля, тобто правих частин в рівняннях Ньютона динаміки руху. Саме останній фактор розглядається як ключовий для подальшого розвитку ММД.

Недоліки ММД пов'язані з тим, що розглядаються тільки силові поля молекулярної механіки, при тому що потенційна енергія складних молекул, таких як біологічні макромолекули, складається не тільки з потенціалів всіх сил взаємодії, але й з *ентропійних сил*, які зв'язані з певним розташуванням частинок (наприклад, амінокислот в молекулі білка), тобто геометрією (як кажуть в біофізиці, з конформацією молекули). У фізиці ентропійна сила виникає в результаті статистичної тенденції всієї системи до збільшення її ентропії, що відповідає другому закону термодинаміки. В ізольованих системах частинок реалізуються стани з більшою термодинамічною ймовірністю [7]. Прикладом ентропійної сили є сила, яка виникає в ланцюгових

молекулах полімерів, білків та ін. Випрямлений ланцюг відповідає стану з низькою термодинамічною ймовірністю він намагається скоротитися, що виглядає як сила пружності, яка намагається скоротити відстань між кінцями ланцюга і ця сила пропорційна відстані, тобто велика для випрямленої молекули і менша – для молекули, яка згорнута у клубок.

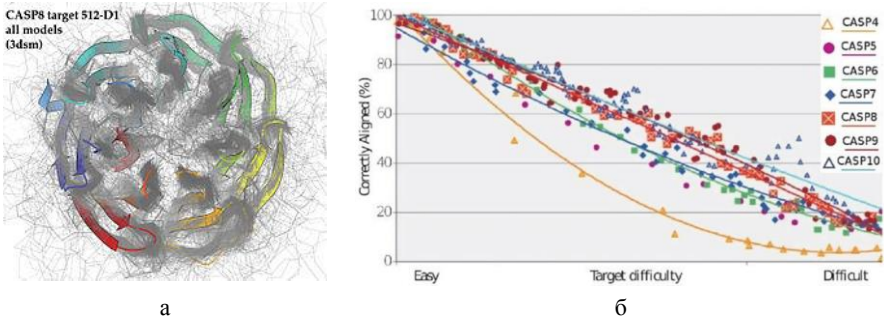


Рис. 16. Точність МД: а – порівняння структури молекули (кольорова) і декількох МДД симуляцій (сірим); б – зростання точності протягом 12 років

Іншим прикладом ентропійних сил є специфічні сили взаємодії, які пов'язані з *гідрофільними* / *гідрофобними* (схильність до контакту з молекулами води) або *ліофільними* / *ліофобними* (схильність до контакту з жировими молекулами) властивостями частинок (амінокислот, молекул, агрегатів з молекул). Вони пов'язані з енергетичною «невигідністю» контактів вода – жир і тому відповідні поверхні розділу намагаються до мінімальної площини поверхні (рис. 17а).

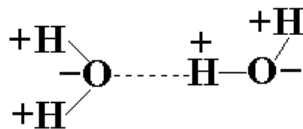


Рис. 17. Краплі води на гідрофобній поверхні рослини (а) і водневі сили (б)

Важливими є також внутрішньомолекулярні *водневі зв'язки*, які не враховуються в моделях силових полів, але можуть додаватися як кулонівські взаємодії (33) точкових зарядів атомів. Це наближення неточне, оскільки водневі зв'язки мають частково *квантово-механічну і хімічну природу*. Водневий зв'язок – це форма диполь – дипольного зв'язку, яка відноситься до тяжіння між атомом водню Н, який пов'язаний з елементом з високою електронегативністю, найчастіше з киснем, азотом, хлором, фтором (рис. 12в,

рис. 176). Таким чином, ці сили вкрай важливі для водних розчинів, які найчастіше за все і вивчаються за допомогою ММД, оскільки в більшості існують як в неживій природі, так і в організмах людей, тварин і рослин.

Крім того, ММД застосовний, якщо довжина хвилі Де Бройля атома (або частинки) набагато менша, ніж міжатомна відстань. Квадрат амплітуди хвилі де Бройля в даній точці є мірою ймовірності того, що частинка знаходиться в цій точці. Де Бройль висунув ідею, що хвильовий характер руху, який був встановлений для фотонів (квантів світла), має універсальний характер і має проявлятися для будь-яких частинок, які мають скінченний імпульс p . **Квантові ефекти** є суттєвими, якщо характерне значення дії (енергія помножена на час, або імпульс p помножений на відстань) стає порівнянним з постійною Планка. Квантові ефекти стають визначальними на низьких температурах, і у цьому випадку треба використовувати квантовохімічні взаємодії. Коли потрібні більш тонкі рівні деталізації молекулярної динаміки, до правої частини (1) додають квантовомеханічні потенціали. Існують гібридні класичні / квантові потенціали, де основна частина системи розглядається на основі класичних потенціалів (VdW, електростатичних сил), але деяка визначена невелика область розглядається як квантова система, в якій проходять більш складні фізичні і хімічні процеси.

Серед недоліків ММД також визначають умови на характерні часи руху системи, а саме ці часи мають бути більшими, ніж час релаксації (час повернення до початкового стану після зняття зовнішніх сил) досліджуваних фізичних величин. Крім того, класичний ММД не застосовується для моделювання систем, які складаються з легких атомів, таких як гелій або водень.

Похибки до розрахунків ММД можуть вноситися і використанням в формулі Кулона діелектричної проникності вакууму ϵ_0 , хоча навколишнє водне або інше середовище має набагато вищу власну діелектричну проникність. Сили VdW в ММД зазвичай описуються потенціалами L-J на основі теорії Ф. Лондона, яка застосовна лише у вакуумі, а не в розчиннику. Однак всі типи сил Ван-дер-Ваальса (див. вище) мають саме електростатичне походження, а значить, залежать від діелектричних властивостей середовища. Однак використання макроскопічної діелектричної проникності на коротких міжатомних відстанях є сумнівним. Пряме вимірювання сил тяжіння між різними матеріалами показує, що для молекул вуглеводів електростатична взаємодія в воді становить $\sim 10\%$ від значення, яке розраховано на формулах з використанням ϵ_0 [8].

Обмеження ММД пов'язані з необхідністю достатньо малих кроків за часом і великого числа розрахунків, що потребує від кількох днів до місяців комп'ютерного часу навіть на багатопроцесорних системах. Якщо число кроків за часом недостатньо, результати розрахунків можуть не відображати реальну поведінку фізичної системи, таку як перехід від регулярної динаміки до хаотичної, біфуркацію стану, фазові перетворення (газ \rightarrow рідина \rightarrow тверде тіло). Для вирішення цієї проблеми розроблені спеціальні алгоритми, напри-

клад алгоритм обмеження SHAKE, який фіксує коливання найшвидших атомів (наприклад, атомів водню) і дозволяє підвищити крок за часом без втрати точності алгоритму.

Якщо комп'ютерні ресурси обмежені, треба суттєво зменшувати число частинок в моделі. Якщо число частинок недостатньо велике, модель може не відображувати реальні статистичні розподіли енергії, температури, тиску та ін. Найбільш інтенсивне завантаження процесорів пов'язане з розрахунками потенціалу (правих частин в (1)), особливо його нековалентною частиною, яка суттєво залежить від координат всіх (особливо найближчих) частинок в системі. Для розв'язання цієї проблеми використовуються алгоритм Евальда (Ewald summation method) підсумовування потенціалів та методи відсічі з урахуванням взаємодії тільки з найближчими сусідніми частинками (рис. 18). В цьому методі періодичний розподіл заряду у просторі (внаслідок періодичних граничних умов) розкладається у ряд Фур'є, а у Фур'є-просторі (frequency domain) підсумовування стає швидким.

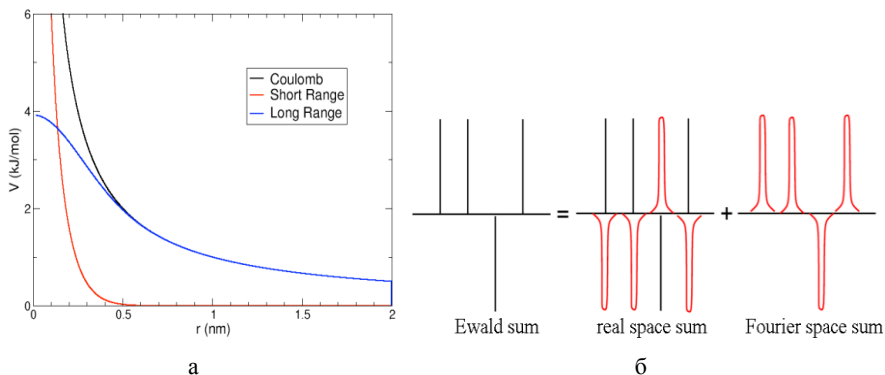


Рис. 18. Розділення електростатичного силового поля на далеко діючу і короткодіючу складові (а), а сума Евальда обчислюється через ряд Фур'є (б)

Якщо проводиться моделювання взаємодії частинок у розчиннику (молекули у воді, спиртах, гліцерині), можна використовувати явні або неявні моделі розчинника. В явних моделях сили взаємодії молекул розчинника розраховуються за (1) як силове поле, що потребує величезних комп'ютерних ресурсів. Неявні моделі використовують підхід середнього поля для молекул розчинника і більш швидкі.

Для уникнення впливу граничних умов на результати розрахунків, найчастіше в ММД використовуються періодичні граничні умови, але і при цьому оцінка впливу має бути перевірена. Використання періодичних граничних умов означає, що протилежні грані звичай кубічної просторової області, в якій моделюється система, ототожнюються. Внаслідок цього формально розглядається нескінченно велика система, але вона залишається замкненою без запровадження додаткових алгоритмів.

Як вже зазначалось вище, способом перевірки (валідації) молекулярної моделі, є оцінка статистичних параметрів системи (43), (44), які можуть вимірюватися експериментально в реальних газах, рідинах і твердих матеріалах, включаючи біологічні тканини / матеріали [7]. Проблемою ММД в цій перевірці є обчислення температури як середньої кінетичної енергії руху достатньої кількості частинок, яка припадає на один ступінь свободи (9). Оскільки температура – це величина статистична, то потрібна достатня кількість частинок в системі і числі кроків за часом, щоб обчислена середня кінетична енергія відповідала статистичній температурі.

Часто при моделюванні макромолекулярної системи в середовищі розчинника, такому як молекули води та солі, ми стикаємося з різними наборами ступенів свободи, що характеризуються або дуже різними характерними частотами, або дуже різними швидкостями нагрівання. У цьому випадку зв'язок усіх ступенів свободи з одним термостатом може спричинити різні температури конвергенції для різних підмножин ступенів свободи системи через різні швидкості обміну кінетичної енергії між підмножиною ступенів свободи та термостатом. Зокрема, у моделюванні макромолекулярних систем це називається проблемою «гарячого розчинника» та «холодної розчиненої речовини». Оскільки на динаміку розчинника суттєво впливає використання далекосяжного скорочення, зокрема, електростатичні взаємодії, і, отже, зв'язок усієї системи з одним термостатом може призвести до того, що середня температура розчиненої речовини буде нижчою за середню температуру розчинника. У цьому випадку для усунення проблеми можна використати роздільне зв'язування ступенів свободи розчиненої речовини та розчинника з двома різними термостатами.

Інша проблема виникає, якщо ми використовуємо програму МДД, яка неправильно пов'язує термостати безпосередньо з атомними швидкостями, коли спочатку загальний лінійний і кутовий моменти системи не встановлені на нуль. У такій ситуації лінійний і кутовий моменти системи не зберігаються, що не відповідає законам механіки збереження імпульсу і моменту сил в системі.

Залежно від числового інтегратора, який використовується, ці величини не зберігаються через чисельні помилки, навіть якщо спочатку вони були встановлені на нуль. У цьому випадку моделювання показало, що термостат передає кінетичну енергію від високих частот до низькочастотних ступенів свободи; це явище відоме під назвою «ефект літаючої крижинки» (flying ice cube effect³).

Для виконання моделювання в ансамблі NVT можна спочатку запустити врівноваження за допомогою термостата Берендсена при малому значенні τ (наприклад, $\tau = 0,01$ пс). Після того як система стане врівноваженою, слід збільшити τ , щоб отримати достатню рівноважну роботу всіх сил в системі. Тоді, якщо необхідний належний канонічний ансамбль із правильними коливаннями при цільовій температурі, можна використовувати не один, а ланцюжок термостатів Нозе-Хувера.

2. МОДИФІКАЦІЇ І УЗАГАЛЬНЕННЯ ММД

Класичний ММД, підходи і алгоритми якого детально обговорені у попередньому розділі, протягом десятиріч широко використовувався для моделювання різних молекулярних і мезоскопічних систем в різних експериментальних умовах. Класичний ММД, який був по суті розроблений наприкінці 70-х років, починав з моделювання кількох сотень атомів. Історично перші моделювання були проведені для структури і динаміки молекул білків, які розглядали як ізольовані ланцюги у вакуумі. Пізніше до цих моделей були додані молекули води як розчинника. Потім стали братися до уваги сусідні білкові молекули, як це робилося для атомів у кристалічних структурах. Для економії часу і ресурсів для обчислень, розглядалися частинки в деякому репрезентативному елементарному об'ємі (representative elementary volume, REV), який вміщує всі типи молекул у пропорційному складі і фазі, які присутні в системі (тверді, рідкі, газоподібні), у вигляді, як правило, кубічної решітки (рис. 3). Потім REV періодично дублювався в усіх напрямках з періодичними граничними умовами (рис. 19), щоб

представити по суті нескінченну систему або речовину (матеріал). Атоми за межами центрального REV є тільки зображеннями атомів, які змодельовані у самому REV. Так звані періодичні граничні умови гарантують, що всі змодельовані атоми оточені сусідніми атомами, незалежно від того, чи є ці сусіди зображеннями чи ні. Так звана умова мінімального зображення гарантує, що подвійні взаємодії між атомами не враховуються шляхом обчислення лише однієї попарної взаємодії для кожної пари атомів. Для атомів з номерами i та j взаємодія полягає у взаємодії між оригінальним атомом i та будь-якою копією атома j , оригіналом чи зображенням, яка є найближчою до атома з номером i .

Сучасні комп'ютерні системи дозволяють проводити симуляцію систем, що містять більше 10^6 частинок, а дослідження динаміки систем з приблизно 50000–500000 атомів тепер є майже рутинною задачею. З підвищенням технічних можливостей комп'ютерів з'явилися потреби в моделюванні більш складних взаємодій і стало ясно, що багато фізичних явищ не вдається описати в рамках стандартного ММД. У зв'язку з цим були запропоновані, протестовані, валідовані і широко розповсюджені численні узагальнення і модифікації класичного ММД.

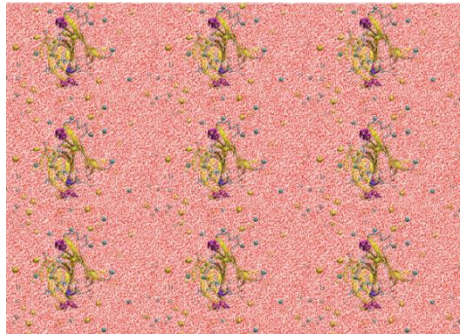


Рис. 19. Приклад повторення REV (молекула білка у оточенні водного розчину) в квадратній решітці

2.1. Керована молекулярна динаміка (КМД)

Більш узагальнений підхід КМД базується на симуляції силового зонду (наприклад, кінця зонду атомно-силового мікроскопу, рис. 20а), який прикладає до молекули силу з ціллю впливу на її геометричну структуру в просторі (рис. 20б). Такі комп'ютерні експерименти найчастіше використовуються для дослідження структури молекул білків та інших біологічних макромолекул, виявлення їх структурних змін на атомарному рівні, для моделювання процесів механічне згортання (folding, фолдінг), розгортання (unfolding) або розтягування (stretching) молекул. Відомо, що всі білки утворюються в біологічних клітинах як ланцюги з амінокислот (АК), яких (АК) існує 20. Складні специфічні комбінації АК в ланцюгах формують первинну структуру білка (рис. 21а). Одразу після закінчення додавання нових АК ланцюг змінює свою геометрію, наприклад, утворює просторову спіраль α -helix (рис. 21б). Пізніше за рахунок більш складних (поліпептидних) зв'язків утворюється третинна структура (рис. 21в), а за рахунок просторового асемблювання – четвертинна структура білка (рис. 21г).

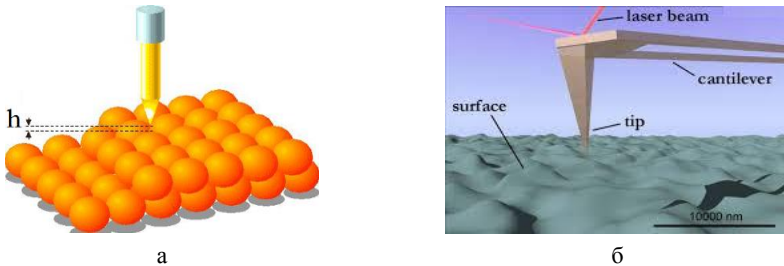


Рис. 20. Принцип роботи атомно-силового мікроскопу: зонд над досліджуваною системою молекул поверхні (а) і вимірювання сили за допомогою відбиття лазерного променя від рухомого важеля (б)

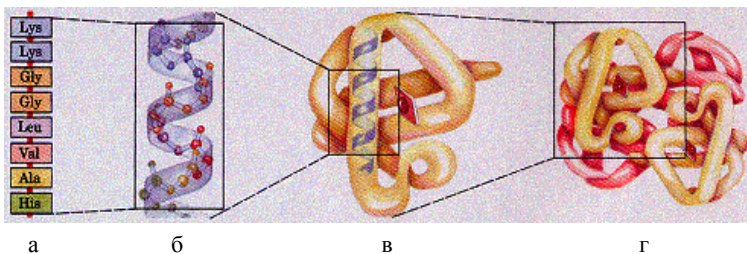


Рис. 21. Первинна (а), вторинна (б), третинна (в) і четвертинна (г) структури (конформації, геометрії) молекули білка

Розроблені і використовуються два основних протоколи КМД – з постійною швидкістю руху, і з постійною прикладеною силою, тобто зонд

задає вплив з постійною швидкістю або силою до конкретних атомів поверхні або розчину. Решта атомів може бути зв'язана силами взаємодії і лише опосередковано «відчували» локальний вплив зонду. Для прикладання сили / переміщення вздовж вибраної координати або ступеню свободи використовується **парасолькова вибірка** – це метод обчислювальної фізики і хімії, який використовується для покращення статистичної вибірки за умовами дуже неоднорідного розподілу енергії в системі. Цей підхід був запропонований в 1977 р. і є прикладом статистичної вибірки за важливістю до фізичних систем. За допомогою парасолькової вибірки адекватно відбираються всі конфігурації системи як високоенергетичні, так і низькоенергетичні. Тоді зміна вільної енергії кожної геометричної конфігурації частинок розраховується на основі теорії потенціалу середнього силового поля, що дозволяє зберігати обчислювальний ресурс (комп'ютерну потужність і час).

2.2. Методи квантової механіки та змішані КМ / ММ методи

Як вже зазначалося вище, насправді структура атомів складна і включає елементарні частинки (електрони, протони, бозони, ферміони та ін.), які мають заряди, спин та ще низка специфічних властивостей, які описуються за допомогою квантово-механічних сил / полів. Безпосереднє застосування ММД до складових кожного атома призводить до суттєвого ускладнення моделі, невизначеності у виборі параметрів в рівняннях типу (38) або інших, значному підвищенню вимог до часу і комп'ютерної міцності для проведення чисельних розрахунків. Тому привабливими та більш економними є змішані методи квантово-механічних і молекулярно-механічних силових полів (КМ / ММ).

На жаль, пряме моделювання молекулярної динаміки за допомогою КМ / ММ методів *ab initio* (від самого початку, лат.) є дуже потужним для вивчення механізму хімічних реакцій у складному середовищі, але також вимагає багато часу. Однак обчислювальні витрати на обчислення КМ/ММ під час моделювання молекулярної динаміки можна значно зменшити за допомогою напівемпіричних методів КМ / ММ з меншою точністю. Щоб досягти вищої точності на початковому рівні КМ / ММ, був висунутий алгоритм корекції існуючої напівемпіричної моделі КМ / ММ за допомогою нейронної мережі для прогнозування різниці потенційної енергії між напівемпіричним і *ab initio* КМ / ММ підходами. Цей підхід виходить за рамки даного посібника, але бажаючі можуть ознайомитися з ним у фундаментальному підручнику [10].

2.3. Моделі для специфічних зовнішніх полів

Нові експерименти з газами, рідинами, твердими, в'язкопружними, композитними та іншими матеріалами і біологічними клітинами, тканинами і їх заміниками в різних зовнішніх полях вимагають від ММД розробки

нових підходів і додавати сили, які виникають під час впливу теплових (нагрів або заморожування), акустичних, ультразвукових полів або радіації. Остання викликає радіаційно-стимульовані хімічні перетворення складних молекулярних систем, які не враховані в класичних ММД. Молекулярна динаміка, яка керована опроміненням, пояснює випадкові локальні квантові перетворення в системі (наприклад, розриви молекулярних зв'язків, іонізацію та зміни молекулярної топології) під час опромінення (рис. 22). Один з реалізованих модифікованих підходів можна знайти у статті [11].

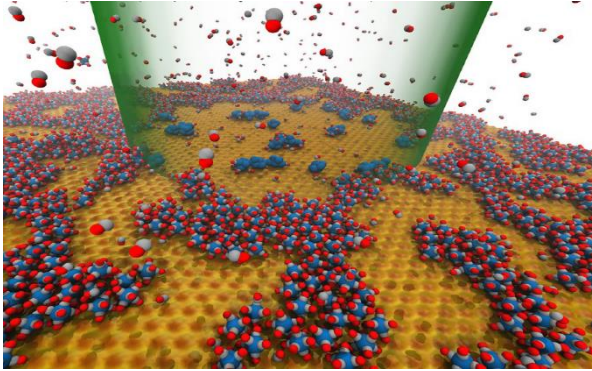


Рис. 22. Результати моделювання поведінки молекул на шарі графену під час опромінення [11]

2.4. Стратегії вдосконалення формування ансамблю

Підвищення кількості періодів обчислень рівнянь (1), тобто просто довшого моделювання, недостатньо для розширення конформаційної вибірки в біомолекулярних системах. Складна форма ландшафту вільної енергії змушує більшість симуляцій досліджувати лише невелику область навколо мінімуму енергії, найближчого до початкової конформації. З появою high-performance computing (HPC)-систем очевидною стратегією є виконання серії паралельних симуляцій з кількома початковими конформаціями, що потребує спеціальних знань про досліджувану систему частинок (атомів, молекул, клітин). Цей підхід особливо корисний, коли доступно кілька кристалічних структур (наприклад, у випадку регульованих ферментів).

Друга проблема, яка виникає під час обчислення наборів паралельних симуляцій, полягає у створенні придатного для використання ансамблю з отриманих траєкторій. Нещодавно з цією метою була використана **теорія моделі стану Маркова (MSM)**. Теорія MSM дискретизує конформаційний ансамбль у наборі станів і будує матрицю з обчисленими ймовірностями переходу між ними. Аналіз такої матриці дозволив би реконструювати глобальну поведінку системи. Оскільки темпи переходу збігаються швидше,

ніж чисельність частинок в підсистемах, цей підхід має перевагу в тому, що набір симуляцій не повинен бути особливо тривалим. Цей підхід використовувався в основному для вивчення процесів згортання білків, а також для кінетичної характеристики утворення білкових комплексів. Інші підходи були розроблені для збільшення простору вибірки в окремих симуляціях як, наприклад, *метадинаміка*, де підтримуються зважені ансамблі, в яких додаткове моделювання запускається при відвідуванні нових конформаційних станів, а вже відвідані конфірмаційні стани штрафуються.

Прискорений ММД базується на підході, в якому енергетичні бар'єри штучно знижені для підвищення швидкості циклу розрахунків. Однак навіть за ідеальної вибірки ММД не може подолати бар'єри в енергетичному ландшафті, які перевищують загальну енергію, додану до системи. Отриманий ансамбль при одному моделюванні обмежений тими станами, які доступні при температурі моделювання. *Моделювання при високих температурах* було звичайним явищем у класичному ММД, але воно призводило до нереалістичних траєкторій, і, отже, їх слід поєднувати з прогонами при кімнатній температурі. Цей підхід, який називається *імітованим відпалом*, був значною мірою замінений *методами обміну копіями*. Такі методи запускають паралельне моделювання в різних умовах. Найпоширенішим варіантом є температура моделювання. Здатність симуляції до вибірки збільшується з температурою. Симуляції при вищих температурах можуть подолати енергетичні бар'єри та дослідити нові регіони ансамблю. Періодично порівнюються енергії різних моделювань і структури міняються місцями відповідно до їх енергетичного рангу. Основна відмінність цього підходу від імітованого відпалу полягає в тому, що отримується реалістичний ансамбль системи, а термодинамічні параметри (тиск, об'єм, температура) можуть бути отримані з результатів симуляцій. Ідея була поширена на інші стратегії моделювання. Найдивовижніше те, що репліки, засновані на відмінностях в *гамільтонівському обміні репліками*, включаючи розрахунки вільної енергії системи або моделювання постійного хімічного потенціалу рН, стають все більш популярними в прикладних задачах хімії, біохімії, біології і медицини (комп'ютерний дизайн ліків).

3. МОДЕЛІ СИЛОВИХ ПОЛІВ

Силове поле ММ визначається не тільки його функціональною формою, але й параметрами, наприклад, в рівняннях (38). Суттєвою особливістю ММ є те, що два силових поля можуть мати однакову функціональну форму, але в той же час дуже різний набір параметрів. Крім того, різні силові поля можуть давати приблизно однакову точність розрахунків. Різні силові поля не можна змішувати; тобто суворо не дозволяється розділяти енергію на окремі члени та брати деякі параметри з одного силового поля та змішувати їх з параметрами з іншого силового поля. Однак деякі з членів силового поля ММ достатньо незалежні від інших (наприклад, терміни розтягування зв'язку та кутового згину), щоб зробити це наближення прийнятним. В ММ існують три основні проблеми параметризації поля сил: мінімальна різниця вільних енергій за наявності великого числа взаємодій, ентропійні ефекти, і різноманітність атомів і молекул в системі, що розглядається.

Варто зазначити, що силові поля молекулярної механіки є емпіричними; тобто не існує «правильної» форми для силового поля. У принципі, якщо одна функціональна форма потенціальної енергії працює краще, ніж інше силове поле, то, швидше за все, цьому силовому полю віддадуть перевагу. Було багато спроб порівняти точність різних типів силових полів. Функціональна форма потенційної енергії, яка використовується в силовому полі, є компромісом між точністю та обчислювальною ефективністю. Таким чином, можна очікувати, що зі збільшенням продуктивності комп'ютера більш складні функціональні форми можна буде включити в силові поля ММ. Крім цього, нові функціональні форми повинні дозволяти швидкі обчислення похідних першого та другого порядку від функції потенціальної енергії відносно координат атомів з метою використання таких методів, як мінімізація енергії системи.

Методи грубої деталізації (coarse-grain, CG), що застосовуються до молекулярного моделювання, дозволяють значно розширити доступні масштаби розмірів та часу зі зберіганнями фізичного і хімічного змісту фундаментальних взаємодій, які керують структурою та динамікою біомолекулярних систем. Ці методи можна розділити на підходи «знизу догори» і «згори донизу». У першому випадку параметри взаємодії для даного молекулярного уявлення розробляються шляхом припасування до моделювання вищого рівня (зазвичай атомістичного). У низхідному підході параметри вибираються відповідно до структурних або енергетичних експериментальних даних.

Нижче наведені найбільш поширені моделі силових полів, які включають певний набір потенціалів і були неодноразово перевірені з використанням експериментальних даних (валідація). Часто моделі полів включають і відповідні комп'ютерні програми / пакети для чисельних розрахунків з тією ж назвою.

3.1. REBO

REBO (Reactive Empirical Bond-Order) – модель реактивного емпіричного порядку зв'язків, яка враховує потенційну енергію ковалентних зв'язків пари атомів, тобто їх хімічну взаємодію. У цій моделі повна потенціальна енергія системи є сумою взаємодій всіх пар найближчих сусідів, причому ці взаємодії залежать не тільки від типу атомів і відстані між ними, але й від їхнього локального атомного оточення. Модель використовує потенціал Дж.Терсофа⁹, який був уточнений в роботах Бреннера¹⁰. Потенціал Терсофа відповідає енергії одинарних, подвійних і потрійних зв'язків у вуглецевих структурах, таких як алмаз, графен, вуглецеві нанотрубки, фулерени і вуглеводні. Енергія хімічних зв'язків в моделі REBO обчислюється за формулою

$$E_{REBO} = \sum_{i,j>i} (U^r(r_{ij}) - b_{ij}U^a(r_{ij})), \quad (45)$$

де U^r і U^a – потенціали сил відштовхування і притягання між парою атомів (ij) на відстані r_{ij} :

$$U^r(r) = f(r) \left(1 + \frac{q}{r} \right) \exp(-ar), \quad U^a(r) = f(r) \sum_{n=1,3} b_n \exp(-\beta_n r). \quad (46)$$

Бреннер додав два члени у функцію порядку зв'язків і тим поширив потенціал Терсоффа на спряжені вуглеводневі зв'язки.

Модель REBO широко використовується для моделювання механічних і теплових властивостей вуглецевих нанотрубок. Порівняно з класичними підходами, ця модель потребує менше часу для обчислень, оскільки розглядає лише взаємодії двох найближчих сусідів, що важливе для систем з 10^3 - 10^6 атомів. Проте потенціал Терсофа є недостатнім для опису реальних сил взаємодій, особливо на малих відстанях. Крім того, модель не враховує витрати енергії на зіткненнях частинок, що складає обмеження моделі REBO.

3.2. AIREBO

Незважаючи на успішні застосування моделі REBO, повідомлялося про її недоліки, для подолання яких було запропоновано розширення потенціалу Бреннера з використанням потенціалу адаптивного міжмолекулярного реактивного порядку зв'язків (Adaptive Intermolecular Reactive Empirical Bond-Order, AIREBO), у якому потенціали як сил відштовхування, так і притягання пар у відповідних функціях REBO були модифіковані відповідно до

⁹ Tersoff J. New empirical approach for the structure and energy of covalent systems. Physical Review. Ser.B. 1988;37(12):6991–7000.

¹⁰ Brenner D.W. Empirical potential for hydrocarbons for use in simulating the chemical vapor deposition of diamond films. Physical Review. Ser.B. 1990;42(15):9458–9471.

властивостей зв'язків. Крім того, були включені дальні взаємодії атомів і торсійні взаємодії (деформації скручування) одинарних зв'язків.

3.3. CHARMM

CHARMM (Chemistry at Harvard Macromolecular Mechanics) [12] – це абревіатура розробленого в Гарварді цілого набору силових полів для молекулярної динаміки, а також пов'язаного з ними комп'ютерного програмного забезпечення для моделювання та аналізу молекулярної динаміки, які широко використовуються в прикладних дослідженнях як сукупностей малих, так і великих біологічних молекул. Силкові поля включають не тільки потенціали Ленарда-Джонса, Кулона і ковалентних зв'язків, але й вигин поза площиною будь-якого набору чотирьох атомів, які не з'єднані послідовно, а також потенціал Урі-Бредлі (Urey-Bradley). Програма CHARMM – це одна із найстаріших програм молекулярної динаміки, яка дозволяє створювати та аналізувати широкий спектр молекулярних моделей із застосуванням таких інструментів, як мінімізація енергії заданої структури, обурення вільної енергії (FEP), оцінку квазігармонійної ентропії, кореляційний аналіз та комбіновані методи квантової і молекулярної механіки (QM/MM). Зараз пакет включає силові поля об'єднаного атома (CHARMM19) і всіх атомів (CHARMM22, CHARMM27, CHARMM36), які постійно доповнюються і оновлюються.

3.4. GROMACS

GROMACS (GRONingen MACHine for Chemical Simulations, гронінгенська машина для хімічного моделювання) [13] – це універсальний пакет програм для моделювання фізико-хімічних процесів у молекулярній динаміці, який був розроблений командою Германа Берендсена, яка працює в Гронінгенському університеті і надалі підтримується співробітниками університету й ентузіастами. Пакет призначений для моделювання біологічних макромолекул (білків, ліпідів, ДНК та ін.), які мають багато зв'язаних взаємодій між атомами і вважається одним із найшвидших інструментів ММД. Пакет є вільним програмним забезпеченням із відкритим вихідним кодом.

3.5. GROMOS

GROMOS (GRONingen MOlecular Simulation [14,15]) – це модель силового поля, яка входить до однойменного пакету комп'ютерного моделювання молекулярної динаміки для біомолекулярних систем. Силове поле GROMOS було розроблено в 1978–1990 рр. для застосування до водних або неполярних розчинів білків, нуклеотидів та цукрів. Також є версія для моделювання молекул, які є ізольованими в газі. Модель може бути застосована до хімічних і фізичних систем, починаючи від скла і рідких кристалів, до полімерів, твердих кристалів і розчинів біомолекул.

3.6. AMBER

AMBER (Assisted Model Building with Energy Refinement [16]) включає будову геометрії і уточнення значення повної енергії для молекул білків і ДНК на основі потенціалу Ленарда-Джонса, ковалентних зв'язків і електростатичних сил; дозволяє також поляризаційні взаємодії. Модель була розроблена наприкінці 1970-х років і підтримується спільнотою розробників. Вона містить як набори молекулярних механічних силових полів для моделювання біомолекул, так і пакет програм молекулярного моделювання.

3.7. LAMMPS

LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator [17]) – це класичний код молекулярної динаміки для моделювання матеріалів у твердому стані (метали, напівпровідники) і м'якої речовини (біомолекул, полімерів), а також для грубозернистих або мезоскопічних систем. Його можна використовувати для моделювання атомів або, більш загально, як симулятор паралельних частинок в атомному, мезо- або макроскопічному масштабах.

3.8. NAMD

NAMD (NAnoscale Molecular Dynamics [18]) – це паралельний код ММД, біомолекулярних систем на основі силового поля Charm++. NAMD може використовувати до сотень ядер у типових симуляціях і близько 500 000 ядер – у великих симуляціях ($>10^6$ частинок). Пакет сумісний з AMBER, CHARMM і низкою інших. NAMD використовує популярну програму молекулярної графіки VMD для налаштування симуляцій та аналізу траєкторій. Вхідний код поширюється безкоштовно і має готові файли для різних платформ. Пакет отримав премію Гордона Белла 2002 р. і 2020 р., а також премію Сідні Фернбаха 2012 р.

3.9. Інші моделі

Інші популярні моделі силових полів були створені як для розв'язання конкретних «вузких» задач (наприклад, тільки для течій рідин), так і для цілого спектру задач з різними типами частинок.

CFF (Consistent Force Field) – це сімейство силових полів, які адаптовані до широкого спектру органічних сполучень, включаючи полімери, метали та ін., який був розроблений ізраїльсько-американським біохіміком і біофізиком Ар'є Уоршелем із співробітниками як загального методу для об'єднання досліджень енергії, структури і вібрації звичайних молекул та молекулярних кристалів. Програма CFF, яка була розроблена Левіттом і Уоршелем, базується на декартовому уявленні всіх атомів; вона стала основою для

багатьох наступних програм ММД. У 2013 р. Ар'є Уоршель разом з Майклом Левіттом і Мартіном Карплюсом отримали Нобелівську премію з хімії за «розробку багатомасштабних моделей для складних хімічних систем».

COSMOS-NMR (Nuclear Magnetic Resonance, NMR) – гібридне квантово-механічне-механічне силове поле, яке адаптоване до різних неорганічних і органічних сполучень та біологічних макромолекул, включаючи напівемпіричний розрахунок властивостей атомних зарядів. Пакет оптимізований для з'ясування структури на основі результатів вимірювань за допомогою NMR (ядерний магнітний резонанс, ЯМР) та реалізований у пакеті молекулярного моделювання COSMOS. Гібридне силове поле проводить явний квантово-механічний розрахунок електростатичних властивостей із використанням локалізованих орбіталей зв'язку з урахуванням флуктуації атомного заряду на кожному кроці ММД розрахунків.

CVFF (Consistent Valence Force Field) – це узагальнене силове поле для частинок з ковалентними зв'язками як малих молекул, так і великих біологічних макромолекул. Зручна вбудована база даних молекул включає водень, вуглець, азот, кисень, сірку, фосфор, галогени, аргон, кремній та їх підвиди. Наприклад, елемент «водень» включає підвиди: загальний дейтерій, дейтерій у важкій воді, загальний водень, водневий зв'язок с N, O, P, S, C, водень у воді, заряджений водень у катіоні та ін.

IFF (Interface Force Field, інтерфейсне силове поле) – це перше розроблене силове поле для металів, мінералів, полімерів, 2D-матеріалів (графен) і їх комбінаціями з найвищою точністю та сумісністю з багатьма іншими силовими полями (CHARMM, AMBER, CFF, CVFF, GROMOS). IFF було розроблено як перше узгоджене силове поле для сполучень майже зі всієї періодичної таблиці. Цей підхід долає відомі обмеження, які пов'язані зі штучним призначенням узгоджених зарядів, використовує стандартні умови як еталонний стан, відтворює структури, енергії та похідні енергії, а також кількісно визначає обмеження для всіх включених сполучень.

MM2 – це силове поле було розроблене Норманом Алінджером для конформаційного аналізу вуглеводнів та інших невеликих органічних молекул. Поле призначене для максимально точного відтворення рівноважної геометрії ковалентних молекул. Підхід реалізує великий набір параметрів, які постійно уточнюються та оновлюються для різних класів органічних сполук.

OPLS (Optimized Potentials for Liquid Simulations) – був розроблений професором Вільямом Л. Йоргенсеном як оптимізований потенціал для моделювання статичної і динаміки конденсованої фази (рідин). Останні open-source версії безкоштовні і призначені для використання на базі різних пакетів ММД і платформ.

GFN-FF (Geometry, Frequency, Non-covalent Force Field) – повністю автоматизоване загальне силове поле, що частково поляризується, для точного опису структур й динаміки великих молекул в періодичній таблиці.

EVБ (Empirical Valent Bond, емпіричний валентний зв'язок) – це реактивне силове поле для моделювання хімічних реакцій в різних середовищах, яке полегшує розрахунок вільних енергій активації в конденсованих фазах та ферментах.

DPD (Dissipative Particle Dynamics, дисипативна динаміка частинок) – це підхід, який зазвичай застосовується у хімічній інженерії, а також для вивчення гідродинаміки різних простих і складних рідин, які вимагають розгляду в більших масштабах часу і довжини, ніж ті, які доступні для класичного ММД. Нещодавно було розпочато роботу з автоматичним долученням деяких хімічних сполучень, що належать до розчинів з метою автоматизованої параметризації потенціалів взаємодії DPD з експериментальними даними.

SAFT (Statistical Associating Fluid Theory) – це модель силового поля, яка заснована на теорії статистично асоційованої рідини для гетероядерних молекул, яка була розроблена групою Molecular Systems Engineering в Імперському коледжі Лондона [19]. Модель адаптована до щільності рідкої фази та тиску парів чистих сполучень з використанням рівняння стану SAFT.

SIRAH (Southamerical Initiative for a Rapid and Accurate Hamiltonian [20]) – це силове поле, яке було розроблене співробітниками групи біополекулярного моделювання Інституту Пастера в Монтевідео, Уругвай для симуляцій молекулярної динаміки води, електролітів, ДНК та білків. Молекулярні взаємодії оцінюються з використанням класичного гамільтоніана, що полегшує його реалізацію у звичайних пакетах молекулярної динаміки. Поле включено до пакетів AMBER, GROMACS і деяких ін.

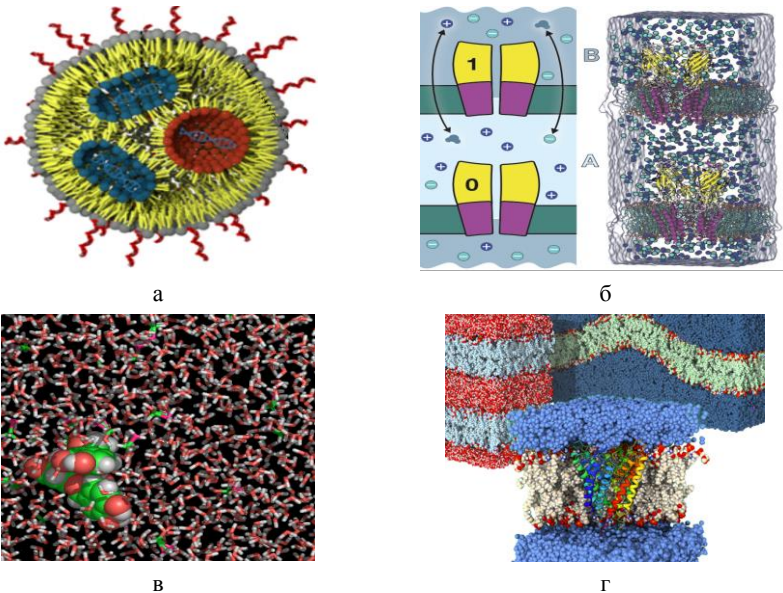
VAMM (Virtual Atom Molecular Mechanics, молекулярна механіка віртуального атома) - це силове поле, яке було розроблене для розрахунків молекулярної механіки, таких як великомасштабні конформаційні переходи, які базуються на віртуальних взаємодіях.

Найпопулярніші коди ММД (AMBER, CHARMM, GROMACS, NAMD) вже давно сумісні з інтерфейсом передачі повідомлень (messaging passing interface, MPI). Якщо одночасно можна використовувати велику кількість комп'ютерних ядер, MPI може значно скоротити час обчислення. Щоб забезпечити локальність взаємодії, загальна стратегія полягає в тому, щоб розподілити систему для моделювання між процесорами. Ця стратегія називається просторовою декомпозицією. Лише невеликий фрагмент системи повинен бути змодельований у кожному процесорі. Найефективніший поділ базується не на списку частинок, а на їх положенні в просторі. Кожен процесор має справу з областю простору, незалежно від того, які частинки там присутні. Зв'язок між процесорами також зменшується, оскільки лише ті, що імітують сусідні регіони, мають обмінюватися інформацією. Як було зазначено, використання прискорювачів, головним чином GPU, стало великим проривом у моделюванні кодів. Спочатку розроблені для роботи з комп'ютерною графікою, графічні процесори еволюціонували в універсальні, повністю програмо-

вані, високопродуктивні процесори та являють собою серйозне технічне вдосконалення для виконання атомістичної МД. Більшість основних МД-кодів уже підготовлено для GPU, і навіть розроблено МД-коди, написані спеціально для використання на GPU (ACEMD61). Симуляція на графічних процесорах окремо або в поєднанні з MPI наразі є стратегією за замовчуванням для високопродуктивного моделювання МД. Помітно, що в той час як моделювання було найпопулярнішим використанням НРС у науках про життя, зростання потужності та складності графічних процесорів призводить до більшого використання персональних робочих станцій із порівнянною продуктивністю. Приклади моделювання в деяких з перелічених вище пакетів наведені на рис. 23, а частотний аналіз використання різних пакетів з часом – на рис. 24.

3.10. Методи машинного навчання і нейронних мереж у ММД

В останнє десятиріччя стали популярними і показали свою значущість методи з використанням *глибокого машинного навчання* (Machine Learning, ML), *нейронних мереж* (Neural Networks, NN) і *штучного розуму* (Artificial Intelligence, AI). Вже майже не існує проблеми або теми, для розв'язання задачі якої не були б використані різні типи ML, NN і AI. Це задачі медичної діагностики, розпізнавання обличчя і мови, автоматичного перекладу, логістики і стану фінансових ринків, розпізнавання і класифікації. Методи і підходи ML, NN і AI реалізовані і в низці пакетів ММД, наприклад:



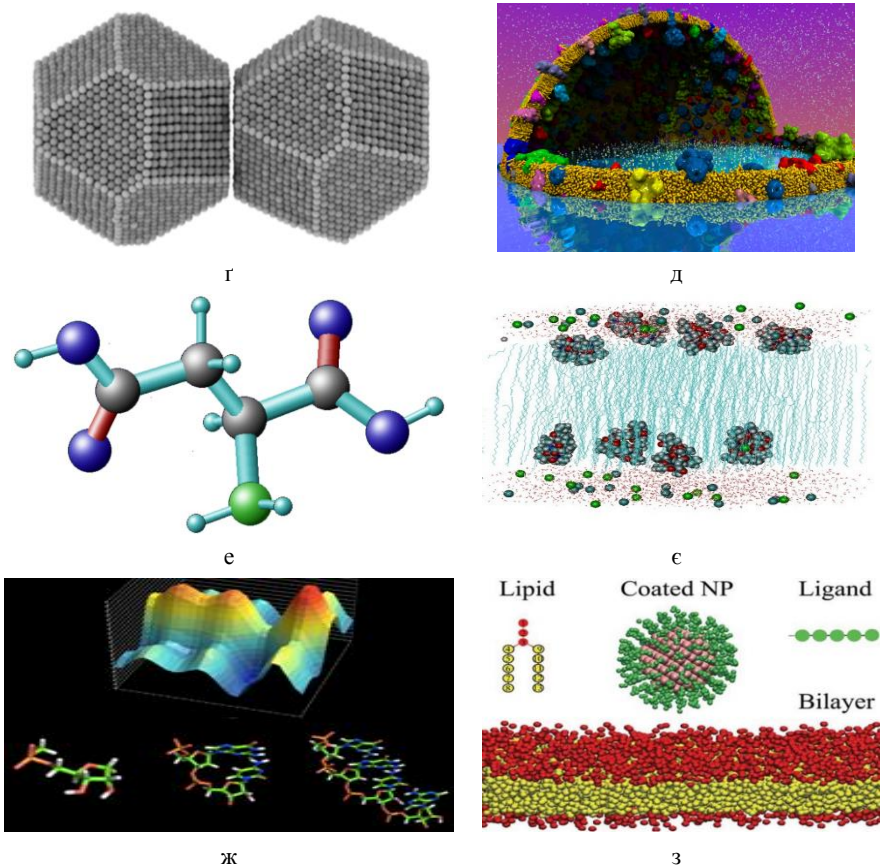


Рис. 23. Приклади візуалізації молекулярних моделей в різних пакетах: мембранні ліпідні наночастинки в CHARMM (а), іонний канал в GROMACS (б), молекула у розчиннику GROMACS (в), молекула етана в AMBER (г), срібні наночастинки в LAMMPS (г), клітинна мембрана в NAMD (д), молекула в COSMOS-NMR (е), мембранні білки в MM2 (е), енергетичні розподілення атомів в OPLS (ж), мембранні компоненти в DPD (з).

ANI (Artificial Narrow Intelligence) – це софт для розрахунків за допомогою нейронної мережі потенціалу взаємодії частинок, який побудований з векторів всіх атомів (частинок) системи і здатний забезпечити точність розрахунків з точки зору закону зберігання енергії. Без використання NN такі розрахунки потребують штучної корекції на кожному циклі алгоритму (рис. 15) ММД. В більш глобальному визначенні вузький штучний інтелект (ШІ) – це особливий тип ШІ, у якому алгоритм навчання розроблено для виконання одного завдання, а будь-які інші знання, які були отримані під

час виконання цього завдання, не будуть автоматично застосовані до інших завдань.

FFLUX (перша назва **Quantum Chemical Topology Forse Field, QCTFF**) – це методологія, яка поділяє будь-яку систему, чи то окрему молекулу, чи збірку молекул (у присутності іонів, з молекулами розчинника чи без них) на так звані топологічні атоми. Це розбиття не містить параметрів, а отримані атоми мають кінцеві межі. Топологічні атоми заповнюють простір: вони не перекриваються і не залишають проміжків між собою. Їхня внутрішня електронна густина характеризується набором атомних мультипольних моментів. З віддаленням від атома для опису його електронної густини, а отже, електростатичного потенціалу або енергії електростатичної взаємодії з іншими атомами, необхідно менше мультипольних моментів. На відміну від популярних силових полів, таких як AMBER або CHARMM, FFLUX «бачить електрони». Іншими словами, FFLUX «знає» про електронну густину атома. Крім того, FFLUX враховує кінетичну енергію атома, унікальне значення якої є найважливішою властивістю, яка спонукала дослідників до розробки першої версії QCTFF. В принципі, FFLUX також «знає» про зменшену матрицю густини другого порядку, яка містить усі ефекти обміну та електронної кореляції. Як наслідок, FFLUX включає всю необхідну інформацію для прогнозування енергетичної поведінки ковалентних зв'язків, а також низки слабших міжатомних взаємодій. FFLUX також має здатність справлятися з ефектами дисперсії, не залучаючи, наприклад, потенціал Леннарда-Джонса, який потребує додаткової енергетичної компоненти. Нарешті, FFLUX зберігає внутрішню енергію атома, яка складається з його кінетичної енергії, кулонівської, обмінної та кореляційної енергії між його електронами, а також енергії їх взаємодії з ядром даного атома.

TensorMol (Tensor Molecular Neural Network) – це змішана модель, нейронна мережа якої забезпечує потенціал ближньої дії, тоді як більш традиційні потенціали додають екрановані терміни дальньої дії. Традиційні силові поля не можуть моделювати хімічну реакційну здатність і страждають від низької загальності без повторного налаштування. Потенціал нейронної мережі вирішує ці проблеми, пропонуючи обчислення енергії і сил з високою точністю майже з самого початку розрахунків (перші цикли використання алгоритму рис. 15). Однак підхід, який керований даними, є природно неефективним для міжатомних сил. Гібридна хімічна модель TensorMol, яка виконана на мові Python з відкритим вихідним кодом, складається з потенціалу короткозорної нейронної мережі з екранованою далекобійною електростатикою та фізикою Ван-дер-Ваальса. Пакет підтримує багато типів моделювання, які зазвичай використовуються для вивчення хімії, а саме: оптимізація геометрії, гармонійні спектри, відкрита або періодична молекулярна динаміка, Монте-Карло та розрахунки еластичної стрічки.

Δ-ML – це підхід комбінованого методу квантової механіки / молекулярної механіки QM /MM, в якому рівняння і алгоритми корекції QM були

замінені моделями машинного навчання ML. Метод був розроблений для молекулярних систем, де необхідний чіткий опис змін в електронній структурі молекул під час взаємодії. Оскільки QM/MM є обчислювально дорогим порівняно з класичним ММД, тому що в ньому всі валентні електрони розглядаються явно, і потрібна процедура самоузгодженого поля (SCF), однак системи конденсованої фази становлять проблему для цих підходів через дальні взаємодії. В пакеті Δ -ML робочий процес включає середовище MM як тип елемента в потенціал нейронної мережі високого розміру (high dimension neuron network potential, HDNNP). Підігнаний HDNNP описує поверхню потенціальної енергії частинок QM за допомогою електростатичної схеми вбудовування. Таким чином, частинки MM «відчувають» силу від поляризованих частинок КМ. Щоб досягти хімічної точності, навіть прості системи вимагають моделей із сильною градієнтною регуляризацією, великою кількістю точок даних і значною кількістю параметрів. Щоб вирішити цю проблему, була використана схема Δ -навчання, де модель ML вивчає різницю між еталонним методом (теорія функціоналу щільності DFT) і дешевшим напівемпіричним методом (функціонал щільності з жорстким зв'язком DFTB). Така схема досягає точності еталонного методу ДПФ, вимагаючи значно менших параметрів. Крім того, схема Δ -навчання здатна правильно включати взаємодії на великій відстані в межах $\sim 1,4$ нм.

SchNet – це архітектура глибокого машинного навчання, яка була спеціально розроблена для моделювання атомістичних систем шляхом використання згорткових шарів безперервного фільтра у хімічному просторі для молекул і матеріалів, де модель вивчає хімічно правдоподібні вбудовування типів атомів у періодичну таблицю. SchNet також використовується для прогнозування поверхонь потенційної енергії та енергозберігаючих силових полів для молекулярно-динамічного моделювання малих молекул. Так, з використанням SchNet, було виконано дослідження квантово-механічних властивостей фулерену C₂₀ (наночастинка), яке було б неможливо здійснити з використанням пакетів звичайної молекулярної динаміки.

PhysNet – це архітектура глибоких нейронних мереж (deep neuron networks, DNN), яка розроблена для прогнозування енергій, сил і дипольних моментів хімічних систем. В останні роки методи ML стають все більш популярними в обчислювальній хімії. Після навчання на відповідних довідкових даних ab initio ці методи дозволяють точно прогнозувати властивості хімічних систем, обходячи потребу в явному розв'язанні рівняння Шредінгера. Через свою обчислювальну ефективність і масштабованість до великих наборів даних глибокі нейронні мережі (DNN) є особливо перспективним алгоритмом ML для хімічних застосувань. Явне включення електростатики венергетичні прогнози є вирішальним для якісно правильного опису асимптотичних областей поверхні потенційної енергії. Моделі PhysNet, які навчені на систематично створеному наборі невеликих фрагментів (щонайбільше вісім важких атомів), здатні до узагальнення на значно більші за

розміром білки. За допомогою об'єктивного ММД на PhysNet-PES у газовій фазі виявлено, що замість спіральної структури молекула може згортатися у «вінкоподібну» конфігурацію, яка є більш стабільною, ніж спіральна форма. Таким чином, відповідне програмне забезпечення з використанням ML відкриває нові горизонти у моделюванні з використанням ММД.

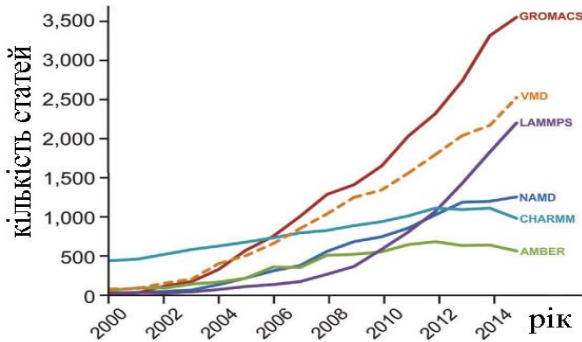


Рис. 24. Еволюція використання деяких із найбільш популярних інструментів ММД [21]

4. ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ММД

4.1. Структура і складові пакетів програм

На сьогодні існує велика кількість програмних продуктів, які забезпечують виконання чисельних розрахунків ММД і МКД, використовують перелічені вище моделі силових полів ММ або КМ. Важливими складовими пакетів є:

- 1) бібліотека частинок (малі й великі молекули у вигляді координат центрів атомів $\{(x_j, y_j, z_j)\}_{j=1}^n$, наприклад, у вигляді матриць (рис. 25) і їх фізико-хімічних властивостей, параметри яких потрібні для обчислень силового поля);
- 2) одна або кілька доступних моделей силового поля (з поляризацією або без, з розчинником або без та ін.);
- 3) можливість залучати інформацію як розробника пакету, так і з відкритих джерел, публікацій в журналах і used-defined data;
- 4) вибір типу статистичного ансамблю, термостату і баростату для контролю фізичної релевантності результатів розрахунків;
- 5) візуалізація результатів розрахунків у вигляді об'єктів lines, sticks, spheres, surface, mesh, dots, ribbons, cartoon та ін. (рис. 26а) і анімацій. В основі геометричних перетворень лежать алгоритми заміни груп частинок в детальній атомістичній моделі (рис. 26б) на більш «грубі» об'єкти в coarse-grain моделі (рис. 26в);
- 6) використання методів машинного навчання ML для прогнозування властивостей системи за допомогою алгоритмів прийняття рішень на основі деяких попередньо визначених характеристик, які характеризують властивості системи, а також для прогнозування відсутніх даних і виявлення нових шаблонів під час процесу інтелектуального аналізу даних;
- 7) використання методів нейронних мереж NN, які розглядають великий набір навчальних даних для такого типу задачі, яка розглядається, а потім будує систему правил для розпізнавання шаблонів у наборі даних.

Рис. 25. Приклад вхідного файлу для завантаження

геометрії системи, яка містить два атоми вуглецю і чотири – водню з радіусами відповідних сфер

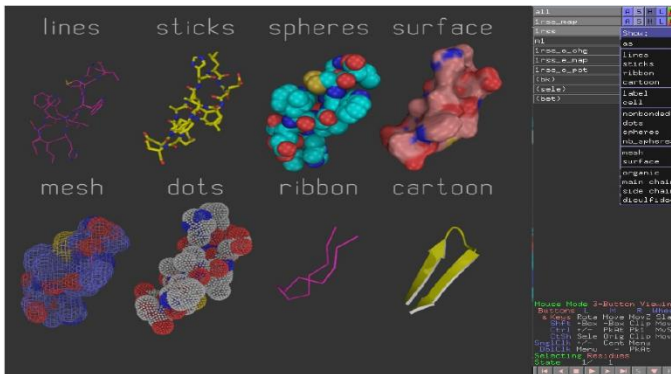
```
Cartesian coordinates
C      0.98770      -0.03260      -0.09450
C      2.32350      -0.03260      -0.09450
H      0.42790      0.73340      0.43290
H      0.42790      -0.79870      -0.62190
H      2.88330      0.73340      0.43290
H      2.88330      -0.79870      -0.62190

Z-matrix
C  1  r1
H  1  r2  2  a1
H  1  r2  2  a1  3  d1
H  2  r2  1  a1  3  d1
H  2  r2  1  a1  3  d1

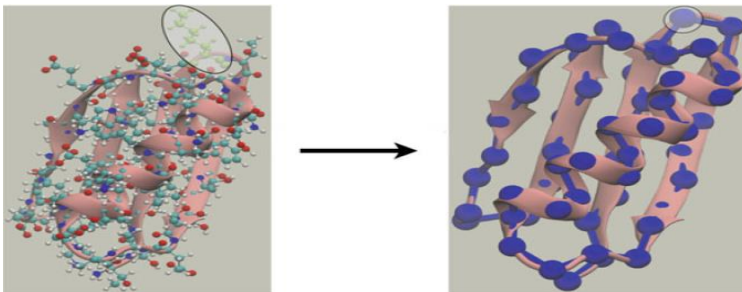
r2= 1.3358
r3= 1.0855
a1= 120.00
d1= 180.00
```

4.2. Приклади програмних пакетів для різних прикладних задач

Abalone [22] – це програма загального призначення для обчислень ММД і візуалізації 3D-молекулярної графіки біомолекул у періодичних граничних умовах у явних (гнучка SPC модель води) або неявних моделях води. Пакет, в основному, призначений для моделювання згортання білка та комплексів ДНК-ліганд у силовому полі. Пакет має автоматичний генератор силового поля для атомів H, C, N, O біологічних молекул, побудові та редагування хімічних структур, бібліотеку



а



б

в

Рис. 26. Різні типи геометричного уявлення молекул як частинок в ММД (а), перетворення атомістичної геометричної моделі (б) в coarse-grain модель (в); як приклад наведена група атомів в еліпсі зліва та відповідний об'єкт в колі праворуч

будівельних блоків, власні узагальнені моделі силових полів з уточненням енергії, оптимізовані можливості для моделювання рідин (OPLS), гібридний метод Монте Карло і ММД з багатокроковим інтегратором, інтерфейс

з квантовою хімією – ORCA, NWChem, Firefly (PC GAMESS), CP2K, і GPU-прискорене молекулярне моделювання.

Avogadro [23] – це розширений редактор молекул і візуалізатор, який був розроблений для кросплатформного використання в обчислювальній хімії, молекулярному моделюванні, біоінформатиці, матеріалознавстві та суміжних областях. Він пропонує гнучку високоякісну візуалізацію та потужну архітектуру плагінів з використанням молекулярного інструктора / редактора для Windows, Linux і Mac OS X, візуалізації, інтерактивних інструментів, команд та сценаріїв Python з підтриманням багатопотокового відтворення та обчислення, з можливістю імпорту хімічних файлів Open Babel, створення вхідних даних для кількох пакетів обчислювальної хімії, кристалографії та біомолекул. Пакет безкоштовний з відкритим вихідним кодом і детальною документацією з перекладом на різні іноземні мови, є інтуїтивно зрозумілим і створеним для зручності роботи як для студентів, так і для досвідчених дослідників.

BALL (Biochemical ALgorithms Library [24]) – це структура класу C++ і набір алгоритмів і структур даних для молекулярного моделювання та обчислювальної структурної біоінформатики з використанням інтерфейсу Python. Використовує **BALLView** [25] – набір моделей для візуалізації атомів, сполучень і поверхонь, а також візуалізацію на основі сітки, наприклад електростатичні потенціали. Засіб перегляду молекул BALLView є програмою C++ для BALL із використанням Qt і OpenGL із трасувальником променів у реальному часі RTFact як механізмами візуалізації. Для них BALLView пропонує тривимірну та стереоскопічну 3D-візуалізацію в кількох різних режимах, а також безпосередньо застосовуючи алгоритми бібліотеки BALL через графічний інтерфейс користувача.

BALL і BALLView орієнтовані на операційні системи Linux, macOS, Solaris і Windows. Зараз BALL – це безкоштовне програмне забезпечення з відкритим вихідним кодом, яке поширюється за ліцензією GNU General Public License (GPL).

Chimera [27] – це розширена програма для інтерактивної візуалізації і аналізу молекулярних структур та зв'язаних з ними даних, включаючи карти щільності, супрамолекулярні збірки, вирівнювання послідовностей, результати стикування, траєкторії та конформаційні ансамблі, яка дозволяє створювати високоякісні зображення та анімації. Chimera містить повну документацію, яку можна завантажити безкоштовно для некомерційного використання. Пакет дозволяє проводити автоматичну ідентифікацію атомів, додавати атоми водню та розподілення часткового заряду, виявляти водневі зв'язки, вимірювати відстані (рис. 27), кути, площі поверхні і об'єми, обчислювати центроїди, відтворювати траєкторії молекулярної динаміки в різних форматах, графіки зміни відстані та кутів з часом. Інструмент ViewDock дозволяє інтерактивний перегляд результатів стикування молекул і структур.

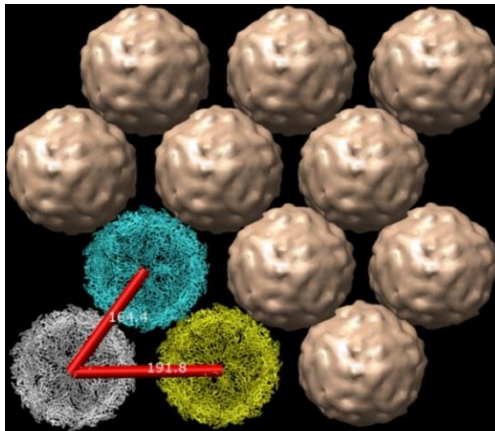


Рис. 27. Модель Chimera вірусу тютюнової мозаїки у вигляді вуглецевої основи (ліворуч унизу) і як контурні поверхні атомної щільності (решта частинок) для порівняння з даними атомно-силового мікроскопа

Gabedit [27] – це графічний інтерфейс користувача, який використовується в пакетах обчислювальної хімії GAMESS, Gaussian, MOLCAS, MOLPRO, MPQC, OpenMopac, PC GAMESS, ORCA та Q-Chem. Він дозволяє будувати молекули за атомом, кільцем, групою, амінокислотою та нуклеозидом, відображати молекулярні орбіталі та електронну густину як контурні діаграми або тривимірні сітки, робити анімацію молекулярних коливань, контурів, ізоповерхонь та обертань молекул, молекулярних систем і кристалічних структур.

Jmol [28] – це комп'ютерне програмне забезпечення для 3D-моделювання молекулярних хімічних структур, які можна повертати в просторі для кращої візуалізації і використовувати для навчання та наукових досліджень у хімії та біохімії. Пакет написаний на мові програмування Java, тому може працювати в операційних системах Windows, macOS, Linux і Unix, якщо встановлено Java. Це безкоштовне програмне забезпечення з відкритим вихідним кодом, випущене за ліцензією GNU Lesser General Public License (LGPL). Існують окрема програма та комплект розробки програмного забезпечення (SDK), які можна інтегрувати в інші програми Java, такі як Bioclipse і Taverna. Популярною функцією є аплет **JSmol** – це інтерактивний об'єкт, який можна інтегрувати у вебсторінки, щоб відображати молекули різними способами. Наприклад, молекули можна відображати як моделі кульок і паличок, моделі заповнення простору, або стрічкові діаграми. Jmol підтримує широкий спектр хімічних форматів файлів, включаючи Protein Data Bank (pdb), Crystallographic Information File (cif), MDL Molfile (mol) і Chemical Markup Language (CML) веб-браузера HTML5, який можна вставляти у вебсторінки.

MDynaMix [29] – це код загального призначення ММД для *моделювання сумішей* твердих або гнучких молекул, які взаємодіють за допомогою силового поля, що складається з потенціалів Леннарда-Джонса, електростатичних, ковалентних зв'язків, кутів і торсійних кутів, а також деяких додаткових. У випадку гнучких молекулярних моделей використовується алгоритм подвійного кроку в часі. Реалізовано алгоритми для статистичних ансамблів NVE, NVT, NPT і анізотропний NPT. Програма може запускатися як у послідовному, так і паралельному виконанні.

Molden [30] – це пакет для обробки загальної молекулярної та електронної структури атомів, молекул, кристалічних структур з можливістю відображення молекулярних орбіталей або електронної густини у вигляді контурних діаграм або тривимірної сітки, анімації реакційних шляхів та молекулярних коливань. Працює на платформах Linux, Windows NT, Windows95, Windows2000, WindowsXP, MacOSX, Silicon Graphics IRIX, Sun SunOS і Solaris. Як загальне силове поле використовує модель Amber, а також допускає білкові силові поля GAFF, яке засноване на атомних зарядах і тому потребує великої кількості обчислень.

Molekel [31] – це багатоплатформна (Mac OS X, Windows, Linux) програма молекулярної візуалізації з відкритим кодом, яка використовує різні методи прискорення візуалізації молекул, орбіталей, ізо-поверхонь матриці густини, поверхні Ван дер Ваальса і змочуваної поверхні (рис. 28), анімацію молекулярних поверхонь і вібраційних мод. Пакет дозволяє експорт зображення високої роздільної здатності (300+ DPI) у PostScript, PDF і gif-анімацію. Він має також плагін для візуалізації скалярного поля, при цьому деяку площину можна вільно переміщувати в 3D-просторі, а точки на площині будуть забарвлені відповідно до значення скалярного поля; курсор можна переміщати по плоскій поверхні, щоб показати точне значення поля в певній точці простору.

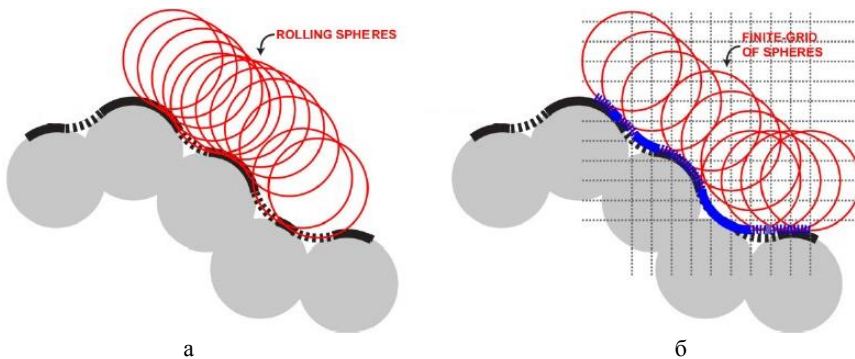


Рис. 28. Візуалізація поверхні молекули, її поверхні Ван-дер-Ваальса і змочуваної поверхні шляхом прокочування сфери з відповідним радіусом (а) або будови решітки сфер (б)

ms2 [32] – пакет для розрахунків термодинамічних властивостей рідин у стані рівноваги, який включає як ММД, так і метод Монте-Карло. Він підтримує розрахунки пар-рідинних рівноваг чистих рідин і багатокомпонентних сумішей, описаних жорсткими молекулярними моделями на основі методу загальної рівноваги. Пакет здатний відбирати різні класичні ансамблі та обчислювати численні термодинамічні властивості рідин. Для оцінки хімічного потенціалу реалізовано метод тестової молекули Відома (Widom's test). Транспортні властивості рідин проводяться шляхом моделювання рівноваги ММД відповідно до формалізму Гріна-Кубо. Пакет написаний на Fortran90 та оптимізований для швидкого виконання на широкому діапазоні комп'ютерних архітектур, починаючи від ПК з одним процесором, ПК-класерів і векторних комп'ютерів до високоякісних паралельних машин.

Ninithi [33] – це безкоштовне програмне забезпечення з відкритим кодом для моделювання, візуалізації та аналізу вуглецевих наноматеріалів, які використовуються в сучасних нанотехнологіях. Користувачі можуть візуалізувати 3D молекулярні структури графену, нанострічок, одностінних та багатостінних вуглецевих нанотрубок і фулеренів. Пакет також надає функції для моделювання електронних зонних структур графену та вуглецевих нанотрубок. Ninithi написаний мовою програмування Java і доступний для платформ Microsoft Windows і Linux. Розповсюджується в рамках умов ліцензування GPL.

NWChem [34] – містить інструменти квантової і класичної обчислювальної хімії, які дозволяють проводити розрахунки з біомолекулами, наноструктурами та твердими матеріалами в станах релаксації і збудження для обчислення їх властивостей з можливістю урахування релятивістських ефектів за рахунок використання від одного до 1000 процесорів. Пакет постійно оновлюється новими даними з кінетики та динаміки хімічних перетворень, хімії на поверхнях розділу (наприклад, процеси на мембранах і у тонких плівках) та в конденсованій фазі.

QuteMol [35] – це інтерактивна система молекулярної візуалізації з відкритим кодом, яка використовує поточні можливості сучасних графічних процесорів через шейдери OpenGL для виконання низки інноваційних візуальних ефектів, такі як коливання в реальному часі, покращення силуету з урахуванням глибини, миттєві знімки зі згладжуванням високої роздільної здатності для створення зображень високої якості для публікацій, інтерактивне відтворення великих біологічних (~100 тис. атомів) з відкритого банку даних білків Protein Data Bank. Методи візуалізації QuteMol спрямовані на підвищення чіткості та легше розуміння тривимірної форми та структури великих молекул, складних білків і молекулярних структур.

RasMol [36] – це пакет візуалізації молекулярної графіки, призначений, головним чином, для зображення та дослідження структур біологічних макромолекул, які знаходяться в банку даних білків PDB. Пакет є важливим інструментом для молекулярних біологів, оскільки надзвичайно оптимізована

програма дозволяє працювати навіть на малопотужних ПК, проводити дослідження у структурній біології та використовуватися для навчання. RasMol містить мову сценаріїв для виконання багатьох функцій, таких як вибір певних білкових ланцюгів, зміна кольорів тощо. Програмне забезпечення Jmol і Sirius включило цю мову до своїх команд.

SAMSON (Software for Adaptive Modeling and Simulation Of Nanosystems) – це платформа комп'ютерного програмного забезпечення для молекулярного проектування з модульною архітектурою для моделювання наносистем (рис. 29), багатокomпонентних матеріалів, біологічних систем і молекулярному дизайні ліків. Програмний комплекс SAMSON включає програми з графічним інтерфейсом користувача, редактори для створення і корекції моделі, деформації молекулярної структури тощо, вбудовані моделі наносистем, і парсери (parsers) – класи, які можуть аналізувати файли для додавання вмісту до графу даних. SAMSON будує моделі систем за допомогою структурних моделей (описують геометрію і топологію), візуальних моделей (генерують графічні зображення), динамічні моделі (описують динамічні ступені свободи системи), моделі взаємодії (сили і загальна енергія), моделі додаткових властивостей (магнітні, оптичні та ін.), і симулятори (потенційно інтерактивні класи, які використовуються для створення фізичних моделей об'єктів і прогнозування їх властивостей). Усі моделі та симулятори об'єднані в ієрархічну багаторівневу структуру, яка формує граф даних SAMSON. Механізм сигналів і слотів дає змогу вузлам графів даних надсилати події, коли вони оновлюються, що дає змогу розробляти, наприклад, адаптивні алгоритми моделювання.

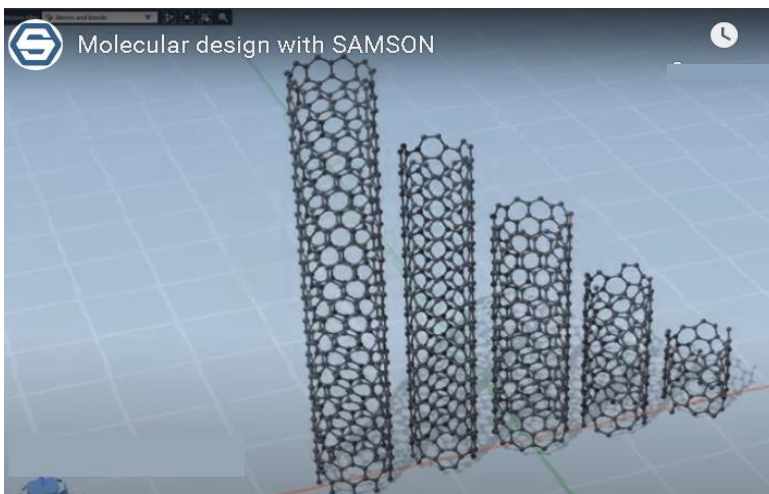
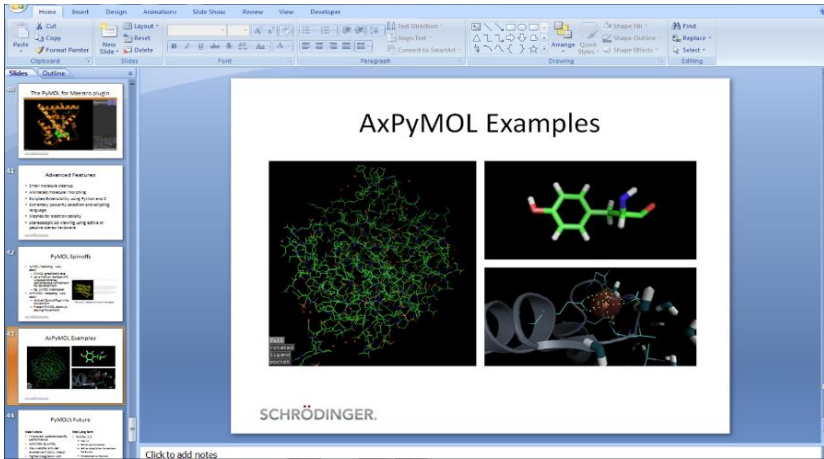
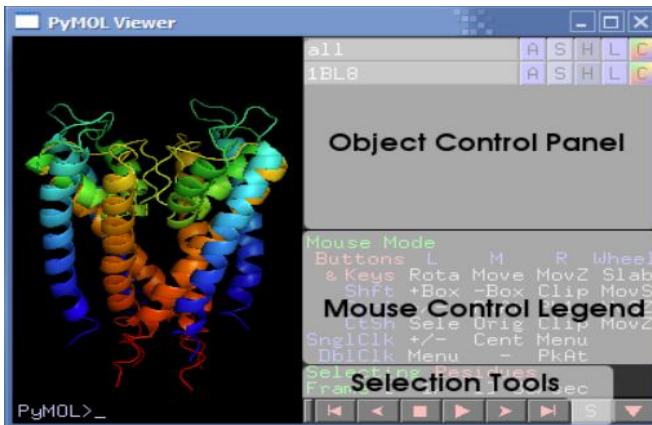


Рис. 29. Візуалізація нанотрубок з різною структурою в пакеті SAMSON



б



в

Рис. 30. Пакет PyMOL: приклад візуалізації молекули білка у вікні Desktop PyMOL (а), експорт згенерованих зображень і результатів обчислень в презентацію ppt (б), вікно PyMOL Viewer з інструментами (в)

4.3.1. Інтерфейс програми

Після інсталяції останньої версії пакету (з [37]) і запуску PyMOL відкриваються два вікна. Менше вікно, яке називається «Зовнішній графічний інтерфейс», містить рядок меню (Файл, Правка, Довідка, Відображення тощо), кнопки швидкого доступу для типових команд і командний рядок. Друге вікно – це вікно перегляду PyMOL Viewer (рис. 30в), де можна відобразити 3D-моделі і маніпулювати з ними (повертати, змінювати).

Об'єкти, які PyMOL відображає в 3D, завантажуються з файлів координат, які описують розташування атомів у вибраній молекулі. PyMOL може відображати більше одного об'єкта одночасно та надає панель керування об'єктами для налаштування режимів перегляду, кольорів, міток, приховування і всього, що стосується об'єктів. Після назви кожного об'єкта є набір командних кнопок, які керують об'єктом (рис. 30а,в). Ось кнопки та деякі з їх опцій:

A – Actions: перейменувати, створити дублікат, видалити, застосувати стилі (такі як "ball-and-stick" або "publication") і виконати обчислення;

S – Show: змінює спосіб відображення молекул, наприклад, у вигляді ліній, паличок, сфер, поверхні (рис. 26а);

H – Hide: структури, які показані за допомогою Show, накопичуються і не замінюють автоматично останній перегляд, тоді як опція Hide приховує непотрібні зображення;

L – Label: дозволяє позначати атоми, залишки і т.д.;

C – Color: змінює колір атомів і груп на інший.

Правий нижній кут в'ювера (Viewer) містить посібник із використання 3D-миші, а також потужний інструмент вибору. У нижній частині вікна перегляду є ще один командний рядок PyMOL>_ (рис. 30в). Щоб запустити будь-яку текстову команду, введіть її в командному рядку PyMOL> і натисніть [Enter].

4.3.2. Візуалізація молекул

Треба вибрати і завантажити файл PDB з інформацією про молекулу, приміром, з банку даних про білки RCSB. Наприклад, розглянемо калієвий канал із *Streptomyces Lividans* (файл 1BL8). Введіть

```
fetch 1bl8
```

і PyMOL знайде і відкриє файл, або використайте опції меню

File => Open...

Структура білка з'явиться у форматі за замовчуванням у вигляді ліній зв'язку.

Права частина вікна показує завантажений PDB як об'єкт, а також його командні кнопки. Кожна кнопка містить підменю з додатковими параметрами. Натисніть «S», а потім «Cartoon», щоб показати вторинну структуру білка у популярній формі мультфільму.

Зауважте, що зображення ліній все ще видно зображення мультфільму. Щоб приховати лінії, натисніть H, а потім Lines.

Щоб змінити колір кожного білкового ланцюга, натисніть «C», а потім виберіть «Chainbows» в меню «By chain». Ця опція забарвлює залишки в кожному білковому ланцюжку у веселку, яка починається синім і закінчується червоним.

Інший поширений метод фарбування призначає один колір кожному ланцюжку. Клацніть «С», а потім виберіть «By chain» у меню «By chain».

Натисніть і перетягніть білок, щоб змінити його вигляд. Під панеллю керування об'єктом знаходиться список кнопок миші:

- Rota: обертання;
- Move: переміщення об'єкта вздовж осі;
- MoveZ: переміщення з зумом;
- Sele: вибір;
- Slab: сплющення;
- Cent: центрування.

Можна змінити розмір вікна перегляду та збільшити / зменшити масштаб, щоб зробити зображення молекули потрібного розміру. Наприклад, зображення для друку та презентації мають бути більшими, ніж зображення для публікації на вебсайті. Оскільки PyMOL був розроблений для роботи на старих комп'ютерах, максимальна якість зображення не ввімкнена за замовчуванням. Щоб змінити це, виберіть Display → Quality → Maximum Quality. Зауважте, що круглі предмети стали круглішими, вигини – вигнутішими, а затінення кольорів - плавнішими.

Знайшовши відповідне подання, збережіть зображення за допомогою File → Save Image. За замовчуванням зображення буде збережено у форматі PNG. Варто перед збереженням використання команду «Ray», що створить зображення ще вищої якості з тінями. Процес рендерінгу може зайняти час залежно від розміру зображення та швидкості комп'ютера, але зображення, які створені після трасування променів, зазвичай дуже реалістичні і використовуються для презентацій.

Іноді може бути корисним вибрати частину об'єкта та безпосередньо змінити її колір. У нижньому правому куті вікна перегляду (рис. 30в) поруч із елементами керування анімацією є кнопка «S» (Selection), яка активує інструмент вибору. Цей інструмент можна змінити, щоб вибрати атоми чи залишки, натиснувши «Selecting Residues» або щось інше, доки не з'явиться потрібний режим. Після активації вибору у верхній частині засобу перегляду з'являється список частин для вибору. Виберіть предмети, вказуючи на них курсором або перетягуючи. Вибором можна керувати індивідуально на панелі керування об'єктом, як і будь-яким іншим об'єктом, а щоб зберегти вибір, треба натиснути «Rename» в меню «A».

4.3.3. Опції збереження в PyMOL

Пакет дозволяє зберігати як цілий проект, зображення (див. 4.3.2), молекули, сеанси, анімації та інші компоненти.

Сеанси PyMOL зберігають стан вашого PyMOL на даний час праці. Ви можете зберегти сеанс і завантажити його пізніше. Ви можете налаштувати складну сцену з переходами і зберегти її як файл сеансу PyMOL (.pse).

Використовуйте File=>Save Session або Save Session As... Завантажити сеанси можна за допомогою: File=>Load та виберіть файл сеансу.

Molecules дозволяє зберегти окремі молекули з вашого проекту. Оберіть File=>Save Molecule і з'явиться діалогове вікно, у якому ви зможете вибрати молекулу для збереження. Замість того можна набрати команду «Save» у діалоговому вікні і вказати ім'я файлу і папку для збереження:

```
save fileName, objSel
```

Movie дозволяє зберігати анімації як автоматично у форматі MPEGЮ, так і у вигляді серії файлів PNG для з'єднання у фільм пізніше.

4.3.4. Скрипти PyMOL with Python

Скрипти варіюються від дуже простих до дуже складних. Можна зробити простий цикл (наприклад, виконати обертання візуалізованої молекули на один градус за крок часу) або виконати повнофункціональні сценарії. Оскільки PyMOL побудовано на основі інтерпретатора Python, будь-яка команда, яку PyMOL не розпізнає, передається на виконання Python. Це дуже зручна функція, тому що, по суті, ми працюємо з інтерпретатором Python, з яким можна безпосередньо взаємодіяти, що значно полегшує ваше життя. Наприклад, цей сценарій

```
f = 10.
for x in range(0,100,10):
    cmd.set("spec_direct_power", float(float(x) / f))
    cmd.png("spec_dir_power" + str(x) + ".png", ray=1)
```

створить 10 зображень, кожне зі зміненим параметром Spec_direct_power для анімованого GIF-файлу. Зверніть увагу, що f у першому рядку *for x* і у другому рядку не є командами або символами PyMOL. Таким чином, можна написати код Python, який взаємодіє з PyMOL. Фрагмент коду (10–20 рядків), який виконує якесь конкретне завдання, треба розмістити між командами «Python» і «Python end» (міні-оболонка Python). Якщо у вашому сценарії буде помилка, можна перервати роботу за допомогою команди «end». Приклад використання оболонки з попереднім сценарієм:

```
python
f = 10.
for x in range(0,100,10):
    cmd.set("spec_direct_power", float(float(x) / f))
    cmd.png("spec_dir_power" + str(x) + ".png", ray=1)
python end
```

Команда «Python» надає повний доступ до оболонки «Python», а команда «Python end» повертає вас до оболонки PyMOL:

```
# enter mini python shell
python
ff = 10.
python end

# now we're back in the normal PyMOL shell, where PyMOL knows about the value
print(ff)

# in the mini shell, Python still knows about ff.
python
print(ff)
python end
```

4.3.5. Графік потенціальної енергії

Використана модель силового поля визначає потенціальну енергію взаємодії частинок у моделі, графік якої треба візуалізувати, щоб зрозуміти особливості моделі і перевірити її фізичну релевантність. Для цього корисним є використання бібліотеки *matplotlib.pyplot*. Нижче наведений приклад коду для порівняння рідин з потенціалом Ленарда-Джонса та з модифікованим потенціалом.

```
import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline

r = np.linspace(0.01,3.0,num=500) #створюємо радіус-вектор
epsilon = 1 #енергетичний мінімум
sigma = 1 #радіус, на якому потенціал дорівнює нулю
E_LJ = 4*epsilon*((sigma/r)**12-(sigma/r)**6) #потенціал Леннарда-Джонса

plt.figure(figsize=[6,6])
plt.plot(r,E_LJ,'r-',linewidth=1,label=r"$LJ$; pot$") #червона лінія - це потенціал ЛД

#значення відсічення та зсуву
Rcutoff = 2.5
phicutoff = 4.0/(Rcutoff**12)-4.0/(Rcutoff**6) #зміщуємо потенціал так, щоб потенціал звертався до нуля

E_LJ_shift = E_LJ - phicutoff #віднімаємо значення потенціалу при r=2,5

plt.plot(r[:415],E_LJ_shift[:415],'b-',linewidth=1,label=r"$LJ$; pot$; shifted$") #синя лінія - це зрушений потенціал ЛД

#оформлення графіку
plt.rc('text', usetex=True)
plt.rc('xtick', labelsz=20)
plt.rc('ytick', labelsz=20)
plt.title(r"$Lennard-Jones$; potential$")
plt.xlim([0.0,3.0])
plt.ylim([-1.5,1.5])
plt.ylabel(r"$E_{LJ}/\epsilon$")
plt.xlabel(r"$r/\sigma$")
plt.axhline(0, color='grey',linestyle='--',linewidth=2)
plt.axvline(1, color='grey',linestyle='--',linewidth=2)
```

4.3.6. Ініціалізація частинок і розрахунок поля сил

Ініціалізуємо частинки завантаженого файлу моделі молекули у масиві даних:

```
DIM = 2
N = 6000

BoxSize = 10.0

volume = BoxSize**DIM
density = N / volume
print("volume = ", volume, " density = ", density)

pos = np.zeros([N,DIM])

pos = np.random.rand(DIM,2)
pos = pos[:, :DIM]/BoxSize

MassCentre = np.sum(pos,axis=0)/N

for i in range(DIM):
    pos[:,i] = pos[:,i]-MassCentre[i]

pos #отримаємо масив з початковими координатами 2-х молекул

volume = 100.0 density = 60.0

array([[0.0549845 , 0.04171987],
       [0.03494417, 0.08455425]])
```

Розрахунки сил взаємодії частинок на кожному кроці проводимо за допомогою опції Compute_Forces:

```
def Compute_Forces(pos,acc,ene_pot,epsilon,BoxSize,DIM,N):
    #Обчислюємо сили на позиціях, використовуємо потенціал Леннарда-Джонса
    Sij = np.zeros(DIM) #масштабовані одиниці
    Rij = np.zeros(DIM) #реальні одиниці простору

    #встановлюємо всі змінні як нуль
    ene_pot = ene_pot*0.0
    acc = acc*0.0
    virial=0.0

    #цикл по всіх парах частинок
    for i in range(N-1):
        for j in range(i+1,N): #від i+1 до N гарантує відсутність подвійного підрахунку
            Sij = pos[i,:]-pos[j,:] #відстань у масштабованих одиницях
            for l in range(DIM): #періодичні взаємодії
                #якщо відстань більша за 0.5, то віднімаємо 0.5, щоб знайти відстань періодичної взаємодії.
                if (np.abs(Sij[l])>0.5):
                    Sij[l] = Sij[l] - np.copysign(1.0,Sij[l])

            Rij = BoxSize*Sij #масштабуємо поле до реальних одиниць, у випадку зменшених одиниць ЛД
            Rsqij = np.dot(Rij,Rij) #обчислюємо квадрат відстані

            if (Rsqij < Rcutoff**2):
                #розрахуємо внутрішній пороговий потенціал ЛД
                rm2 = 1.0/Rsqij #1/r^2
                rm6 = rm2**3.0 #1/r^6
                rm12 = rm6**2.0 #1/r^12
                phi = epsilon*(4.0*(rm12-rm6)-phicutoff) #4[1/r^12 - 1/r^6] - phi(Rc) - використовуємо змінений потенціал ЛД
                dphi = epsilon*24.0*rm2*(2.0*rm12-rm6) # 24[2/r^14 - 1/r^8]
                ene_pot[i] = ene_pot[i]+0.5*phi #накопичуємо енергію
                ene_pot[j] = ene_pot[j]+0.5*phi #накопичуємо енергію
                virial = virial + dphi*np.sqrt(Rsqij) #необхідно для розрахунку тиску
                acc[i,:] = acc[i,:]+dphi*Sij #накопичуємо сили
                acc[j,:] = acc[j,:]-dphi*Sij # (Fji=-Fij)
    return acc, np.sum(ene_pot)/N, -virial/DIM #вертаємо вектор прискорення, потенціальну енергію та віріальний коефіцієнт
```

4.3.7. Інтегрування рівнянь руху

Проводимо інтегрування рівнянь динаміки Ньютона у 2D- або 3D-області за допомогою одного з алгоритмів (Верле, чехарди і т.д., див. 1.4 – 1.7)

```

NSteps=100000 #кількість кроків
deltat = 0.0032 #крок часу в скорочених одиницях часу
TRequested = 0.5# #знижена температура
DumpFreq = 100 #зберігайте позицію для файлу кожного кроку
epsilon = 1.0 #параметр ЛД для енергії між частинками

def main(pos,NSteps,deltat,TRequested,DumpFreq,epsilon,BoxSize,DIM):
    #вектори для зберігання значень параметрів на кожному кроці
    N = np.size(pos[:,1])
    ene_kin_aver = np.ones(NSteps)
    ene_pot_aver = np.ones(NSteps)
    temperature = np.ones(NSteps)
    virial = np.ones(NSteps)
    pressure = np.ones(NSteps)
    ene_pot = np.ones(N)

    vel = (np.random.randn(N,DIM)-0.5)
    acc = (np.random.randn(N,DIM)-0.5)
    f = open('traj.xyz', 'w') #відкриваємо файл, у який ми будемо зберігати результати

    for k in range(NSteps):
        for i in range(DIM): #згортаємо позиції відповідно до періодичних граничних умов
            period = np.where(pos[:,i] > 0.5)
            pos[period,i]=pos[period,i]-1.0
            period = np.where(pos[:,i] < -0.5)
            pos[period,i]=pos[period,i]+1.0

        pos = pos + deltat*vel + 0.5*(deltat**2.0)*acc #r(t+dt) змінюємо положення відповідно до швидкості та прискорення
        ene_kin_aver[k],temperature[k] = Calculate_Temperature(vel,BoxSize,DIM,N)#рахуємо температуру
        chi = np.sqrt(TRequested/temperature[k])#змінюємо масштаб швидкості та робимо половину кроку
        vel = chi*vel + 0.5*deltat*acc #v(t+dt/2)
        acc, ene_pot_aver[k], virial[k] = Compute_Forces(pos,acc,ene_pot,epsilon,BoxSize,DIM,N) #обчислюємо сили
        vel = vel + 0.5*deltat*acc #v(t+dt/2)
        ene_kin_aver[k],temperature[k] = Calculate_Temperature(vel,BoxSize,DIM,N) #рахуємо температуру
        pressure[k]= density*temperature[k] + virial[k]/volume #рахуємо тиск
        #записуємо вихідні дані у файл через кожну кількість кроків
        if(k%DumpFreq==0):
            f.write("%s\n" %(N)) #записуємо кількість частинок у файл
            #записуємо усі кількості на цьому кроці у файл
            f.write("Energy %s, Temperature %.5f\n" %(ene_kin_aver[k]+ene_pot_aver[k],temperature[k]))
            for n in range(N): #записуємо позиції у файл
                f.write("x=" ")
                for l in range(DIM):
                    f.write(str(pos[n][l]*BoxSize)+" ")
                f.write("\n")
            if(DIM==2):
                import matplotlib.pyplot as plt
                from IPython import display
                plt.cla()
                plt.xlim(-0.5*BoxSize,0.5*BoxSize)
                plt.ylim(-0.5*BoxSize,0.5*BoxSize)
                for i in range(N):
                    plt.plot(pos[i,0]*BoxSize,pos[i,1]*BoxSize,'o',markersize=20,)
                display.clear_output(wait=True)
                display.display(plt.gcf())
        f.close() #закриваємо файл
    return ene_kin_aver, ene_pot_aver, temperature, pressure, pos

```

Для перевірки релевантності чисельних розрахунків на кроці часу обчислюємо статистичні параметри, такі як температура:

```
def Calculate_Temperature(vel,BoxSize,DIM,N):
    ene_kin = 0.0
    for i in range(N):
        real_vel = BoxSize*vel[i,:]
        ene_kin = ene_kin + 0.5*np.dot(real_vel,real_vel)
    ene_kin_aver = 1.0*ene_kin/N
    temperature = 2.0*ene_kin_aver/DIM
    return ene_kin_aver,temperature
```

Будуємо графіки для всіх обчислених величин

```
#будуємо графіки всіх величин
def plot():
    plt.figure(figsize=[7,12])
    plt.rc('xtick', labels=15)
    plt.rc('ytick', labels=15)
    plt.subplot(4, 1, 1)
    plt.plot(ene_kin_aver, 'k-')
    plt.ylabel(r"$E_{K}$", fontsize=20)
    plt.subplot(4, 1, 2)
    plt.plot(ene_pot_aver, 'k-')
    plt.ylabel(r"$E_{P}$", fontsize=20)
    plt.subplot(4, 1, 3)
    plt.plot(temperature, 'k-')
    plt.ylabel(r"$T$", fontsize=20)
    plt.subplot(4, 1, 4)
    plt.plot(pressure, 'k-')
    plt.ylabel(r"$P$", fontsize=20)
    plt.show()

plot()
```

і виводимо середні значення тиску, температури і енергії системи

```
print("Temperature = ", np.average(temperature), "+-", 2*np.std(temperature))
```

```
Temperature = 0.4999988758109009 +- 0.004937053984546509
```

```
print("Pressure = ", np.average(pressure), "+-", 2*np.std(pressure))
```

```
Pressure = 30.00133306128694 +- 0.29775844318866446
```

```
print("Energy = ", np.average(ene_kin_aver + ene_pot_aver), "+-", 2*np.std(ene_kin_aver + ene_pot_aver))
```

```
Energy = 0.28809109310030223 +- 0.43619956763086576
```

4.3.8. Приклади візуалізації і розрахунків ММД

Приклади з детальними поясненнями (tutorials) доступні на сайті розробника [38] тільки для зареєстрованих користувачів. Численні приклади RuMOL Tutorial for Beginners можна знайти у зручному відеоформаті, наприклад:

1. Protein Visualization Tool –

<https://www.youtube.com/watch?v=7cX5bVENWAE>

2. Basic Tutorial Molecular Visualization of Proteins –

<https://www.youtube.com/watch?v=h5wKppcyzOw>

3. PyMOL tutorial: Building and Manipulating –

<https://www.youtube.com/watch?v=wLIZAm3Alvg>

Приклад 1. Розрахунки мапи густини електронів в молекулі

1. Завантажуємо файл 1w2i.pdb з ресурсу Protein Data Bank (PDB)

File -> Open -> 1w2i.pdb

2. Завантажуємо файл мапи

File -> Open -> 2fofc.map.xplor

It takes a while to load the map file.

3. Активуємо зум обраної локації

PyMOL> select active, (resi 14-20,38) and chain A

PyMOL> zoom active

PyMOL> hide all

PyMOL> show stick, active

4. Вказуємо локації для приєднання молекул розчинника (води) у вигляді сфер

PyMOL> select active_water, ((resi 38 and name ND2 and chain A) around 3.5) and (resn HOH)

PyMOL> show spheres, active_water

Можна змінити розмір сфери за необхідністю, наприклад, до 0.5 Ангстрем:

PyMOL> alter active_water, vdw=0.5

PyMOL> rebuild

5. Візуалізуємо густину електронної хмари навколо обраної локації

PyMOL> isomesh mesh1, 2fofc.map, 1.0, (resi 14-20,38 and chain A), carve=1.6

PyMOL> isomesh mesh1, 2fofc.map, 1.0, active, carve=1.6

6. Змінюємо кольори мапи за необхідністю

PyMOL> color grey, mesh1

7. Для публікації зображень в журналі можна змінити колір фону на білий

PyMOL> bg_color white

8. Генеруємо високоякісне зображення з роздільністю (640x480)

PyMOL> ray

Останній процес може зайняти деякий час, результат наведений на рис. 31.

Окрім загальних типів рендерінгу (рис. 26а) можливі також більш ускладнені і корисні для досліджень і уявлення результатів cartoon, surface, cut-through of surface, highlighted barrels, Qute-Mol like, Goodsel-like, glossy surface, b-factor putty (рис. 32). На рис. 33 наведена періодична таблиця хімічних елементів з назвами, хештегами і кольорами всіх елементів, як вони прийняті в PyMOL.

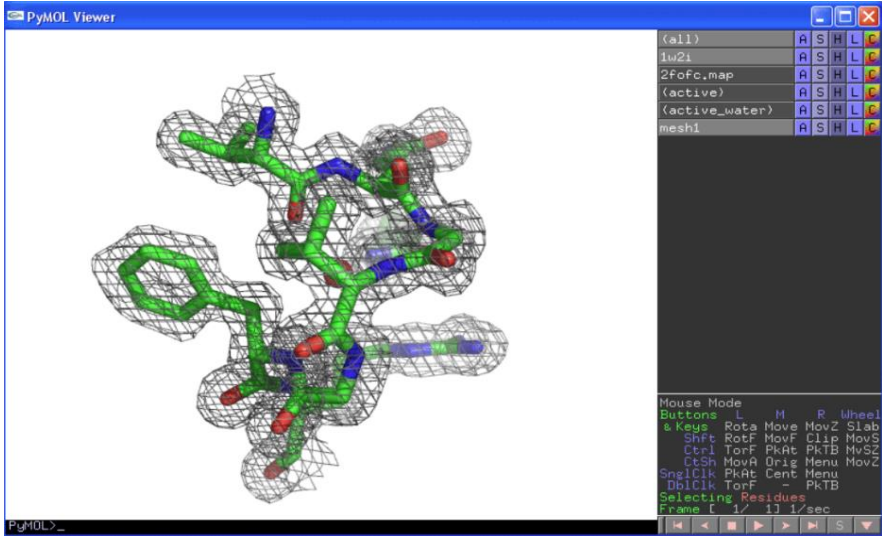


Рис. 31. Результат візуалізації густини електронної хмари молекули ферменту ацилфосфатази за допомогою Desktop PyMOL

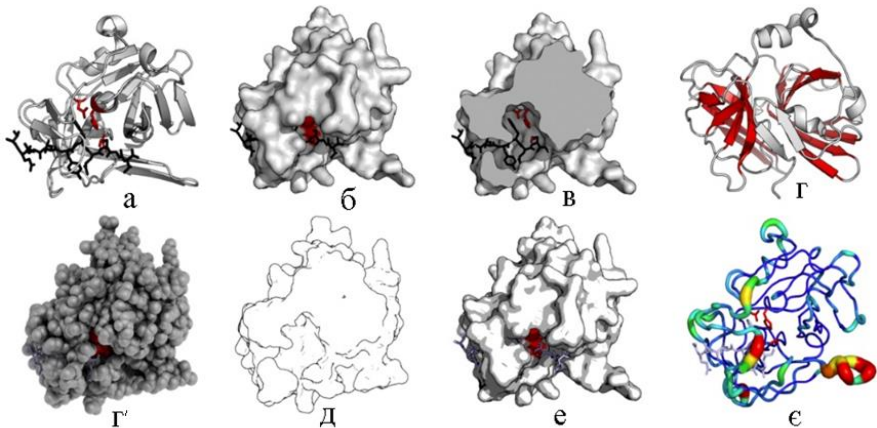


Рис. 32. Ускладнені типи візуалізації молекул і молекулярних систем: мультфільм (а), проста поверхня (б), прорізна поверхня (в), виділені стовбури (г), Qute-Mol тип (г), контури (д), глянцева поверхня (е), пластикна стрічка (е)

Приклад 2. Відображення електростатичної поверхні молекули за допомогою APBS Tools

Для розрахунків електричних властивостей молекули треба видалити із системи (файл 1w2i.pdb) всі молекули води, тобто Кулонові поля електростатичної взаємодії молекули з полярним розчинником (отримуємо файл 1w2i_nowat.pdb). Файли arbs.in і rymol.dx – це вхідний файл з геометрією для чисельних розрахунків і файл мапи електричного поля.

1) Обчислюємо електростатичну мапу

1. Використайте PDB2PQR для конвертації формату PDB у формат PQR:

```
pdb2pqr.py → ff=amber → arbs-in 1w2i_nowat.pdb rymol.pqr.
```

Цей PQR файл буде вихідним для rymol.pqr.

2. Використайте psize.py для визначення розміру сітки для APBS обчислень psize.py rymol.pqr. Зараз ви можете бачити такі результати:

```
Center = 37.468 x 31.798 x 12.177 A
```

```
Coarse grid dims = 53.011 x 58.568 x 65.807 A
```

```
Fine grid dims = 51.183 x 54.452 x 58.710 A
```

```
Num. fine grid pts. = 97 x 97 x 97
```

Збережіть ці параметри (copy-paste).

3. Внесіть зміни до файлу arbs.in з такими параметрами:

```
cgcent 37.468 31.798 12.177
```

```
# Grid Center fgcent 37.468 31.798 12.177
```

```
# Grid Center cglen 53.011 58.568 65.807
```

```
# coarse mesh lengths (A) fglen 51.183 54.452 58.710
```

```
# fine mesh lengths (A) dime 97 97 97
```

```
# Grid Points 4.
```

Запустіть (Run) процедуру APBS як arbs.in Після деякого часу розрахунків ви отримаєте файл rymol.dx з електростатичною мапою.

2) Візуалізуємо електростатичну мапу

1. Відкриваємо (Open) файли PDB і rymol.dx:

```
File → Open → 1w2i_nowat.pdb
```

```
File → Open → rymol.dx
```

2. Виводимо зображення (Display) електростатичної поверхні (рис. 34):

```
Plugin → APBS Tools → Visualization
```

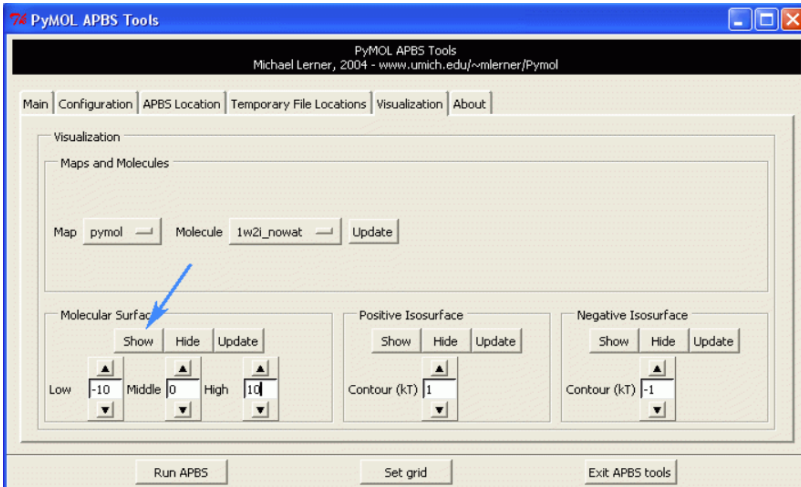
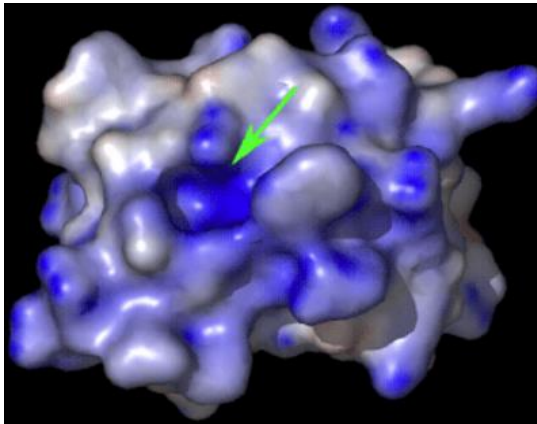
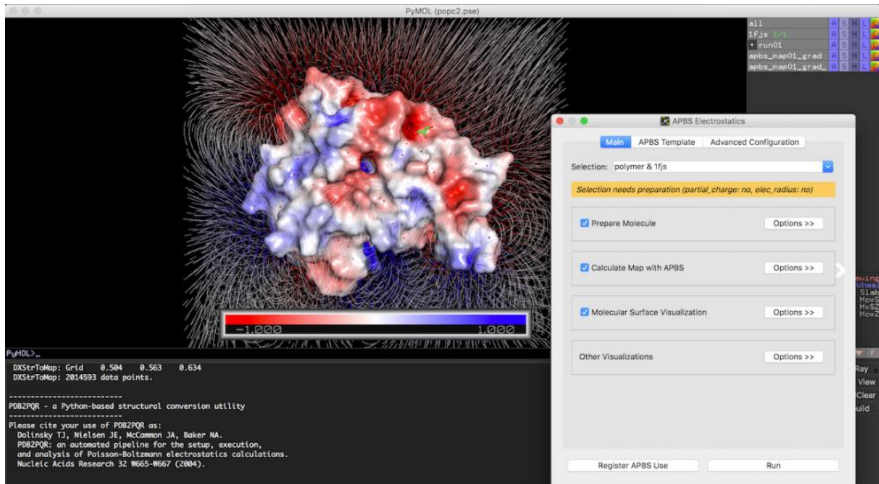


Рис. 34. Вікно PyMOL APBS Tools

Натисніть кнопку Show в меню Molecular Surface. За замовчуванням встановлено: **Blue** surface: +1 kT **Red** surface: -1 kT. Тепер змініть налаштування за замовчуванням на -10 і +10, та знов натисніть Show. Зараз ви можете бачити западину з темно-синьою поверхнею (тобто позитивно зарядженою) на молекулі білка (Рис.35а). Це активний центр молекули ацилфосфатази, який зв'язує негативно заряджений субстрат. Інструмент APBS Electrostatics PyMOL також дає можливість візуалізувати силові лінії електростатичного поля молекули (рис. 35б).





б

Рис. 35. Мапа позитивних електричних зарядів (синій колір) на поверхні молекули (а) і силові лінії поля навколо молекули (б)

Приклад 3. Згортання молекули білку в альфа-спіраль

1. Завантажуємо початкові координати атомів білку з обраного файлу `pdb_file = data/polyALA.pdb` і проводимо візуалізацію (див. 4.3.2) молекули або структури (у даному прикладі молекули поліаланіну `polyALA`).
2. Обираємо модель силового поля і будуємо графік потенціальної функції;
3. Обираємо параметри: температуру, тиск, початкові і граничні умови для частинок (періодичні границі).
4. Обираємо алгоритм розв'язування диференціальних рівнянь руху (інтегратор).
5. Запускаємо чисельні розрахунки і перевірку термодинамічних умов (термостат, баростат). Зберігаємо облічені координати до файлу `.pdb`.
6. Проводимо візуалізацію силових полів, траєкторій, температур і т.д., робимо анімації.
7. Проводимо валідацію результатів комп'ютерного експерименту шляхом порівняння чисельних результатів з даними експериментів (за можливістю).

Наступний код дозволяє подивитися, як виглядає `.pdb` файл:

```
file0 = open(pdb_file, 'r')
counter = 0
```

```

forlinein file0:
ifcounter< 10:
print(line)
counter += 1

```

```

REMARK      1  CREATED WITH OPENMM 7.3, 2019-02-21

ATOM        1  N   ALA  A   1   -36.266 -23.426  -2.296  1.00  0.00      N
ATOM        2  H   ALA  A   1   -36.618 -24.370  -2.382  1.00  0.00      H
ATOM        3  H2  ALA  A   1   -36.620 -23.009  -1.447  1.00  0.00      H
ATOM        4  H3  ALA  A   1   -36.591 -22.873  -3.082  1.00  0.00      H
ATOM        5  CA  ALA  A   1   -34.817 -23.426  -2.296  1.00  0.00      C
ATOM        6  HA  ALA  A   1   -34.451 -23.931  -1.402  1.00  0.00      H
ATOM        7  C   ALA  A   1   -34.294 -21.997  -2.296  1.00  0.00      C
ATOM        8  O   ALA  A   1   -34.784 -21.155  -3.046  1.00  0.00      O
ATOM        9  CB  ALA  A   1   -34.255 -24.128  -3.529  1.00  0.00      C

```

За допомогою модуля `nglview` можемо подивитися, як виглядає цей білок (рис. 36а):

```

u = md.Universe(pdb_file)
ng.show_mdanalysis(u, gui=True)

```

Моделюємо згортання поліаланіну до вторинної структури протягом 400 пікосекунд у вакуумному середовищі при температурі $T=300$ К:

```

### 1. Завантажуємо координати центрів атомів
pdb = PDBFile(pdb_file)
### 2. Задаємо параметри силового поля
ff = ForceField('amber10.xml')
system = ff.createSystem(pdb.topology, nonbondedMethod=CutoffNonPeriodic)
### 3. Вибираємо параметри експерименту: температуру, тиск, розмір
області, сольватацію, граничні умови тощо.
temperature = 300*kelvin
frictionCoeff = 1/picosecond
time_step = 0.002*picoseconds
total_steps = 400*picoseconds / time_step
### 4. Обираємо алгоритм розрахунків (integrator)
integrator = LangevinIntegrator(temperature, frictionCoeff, time_step)
### 5. Запускаємо розрахунки (Run) і зберігаємо координати:
### 5а. Створюємо об'єкти моделювання
simulation = Simulation(pdb.topology, system, integrator)

```

```
simulation.context.setPositions(pdb.positions)
### 5b. Мінімізуємо повну енергію системи для знайдення рівноважної
конфігурації
simulation.minimizeEnergy()
### 5c. Зберігаємо обчислені нові координати кожної частинки
simulation.reporters.append(PDBReporter('data/polyALA_traj.pdb', 1000))
simulation.reporters.append(StateDataReporter(stdout, 5000, step=True,
potentialEnergy=True,\ temperature=True, progress=True, totalSteps =
total_steps))
### 5d. Запускаємо розрахунки
simulation.step(total_steps)
### 6. Проводимо візуалізацію траєкторій і отримуємо згорнуту рівноважну
форму молекули (рис. 36б):
sys = md.Universe(pdb_file, 'data/polyALA_traj.pdb')
ng.show_mdanalysis(sys, gui=True)
```

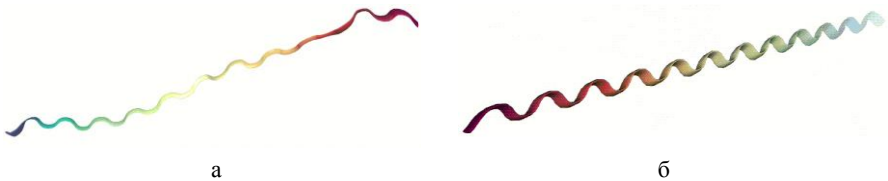


Рис. 36. Початкова геометрія молекули (а)
і згорнута у спіраль (б)

5. ПРИКЛАДИ ЗАСТОСУВАННЯ ММД В СУЧАСНИХ ТЕХНОЛОГІЯХ

1) Молекулярний дизайн ліків.

ММД широко використовується для розробки специфічних молекул (ліків), які діють на конкретні активні центри вірусів, бактерій, отрут і нейтралізують їх на молекулярному рівні. Так, протимікробні пептиди (ПМП) є потенційним рішенням зростаючої загрози резистентності до антибіотиків, але успішний дизайн активних, але нетоксичних ПМП вимагає розуміння механізму їх дії. ММД може надати інформацію на атомному рівні про те, як ПМП взаємодіють із клітинною мембраною (рис. 37).

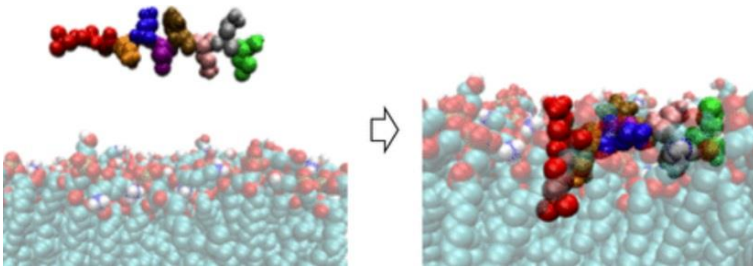


Рис. 37. Результати моделювання ММД процесу вбудовування молекули пептиду до мембрани

2) Моделювання молекулярних двигунів.

Відкриттям останніх років стали синтетичні молекулярні двигуни, які складаються з трьох молекул у нанопорі (рис. 38а), а також природні молекулярні двигуни, які переносять наночастинки в клітинах (рис. 38б).

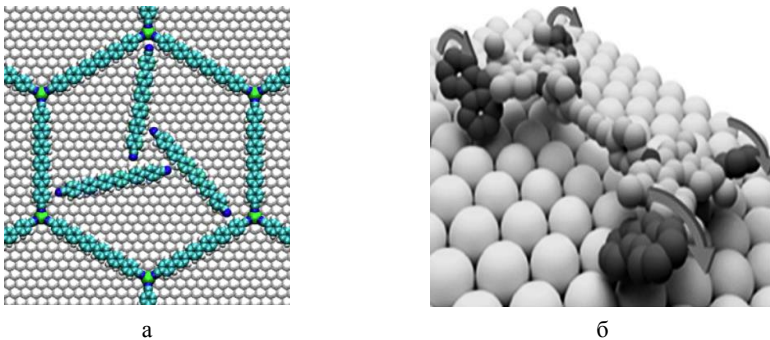


Рис. 38. Молекулярна coarse-grain модель молекулярного двигуна на поверхні нанотрубки (а) і синтетичний молекулярний двигун в нанопорі (б)

3) Моделювання вірусів.

У 2006 р. було проведене ММД-моделювання сателітного вірусу тютюнової мозаїки (СВТМ) з використанням 10^6 атомів в пакеті NAMD. Цей СВТМ вірус є невеликим ікосаедричним вірусом рослин, який погіршує симптоми інфекції звичайним вірусом тютюнової мозаїки (ВТМ). Віруси являють собою білкову оболонку (капсид), в якому знаходиться РНК вірусу, яка при контакті передається здоровій клітині, після чого «хвора» клітина виробляє не свої білки, а білок вірусу, який закодований в цій РНК. Ланцюг СВТМ складається з 60 ідентичних копій одного білка (які утворюють капсид) і 1063-нуклеотидного ланцюга РНК. Якщо б моделювання проводилося на одному ПК з потужністю 2006 року, розрахунки займали б ~ 35 років. Реальні швидкі розрахунки були виконані на великій системі паралельних процесорів з безперервним зв'язком між ними. Одним із отриманих ключових висновків з результатів моделювання є те, що капсид дуже нестабільний, коли всередині немає РНК. Були виявлені патерни і швидкості згортання для стабільних капсидів.

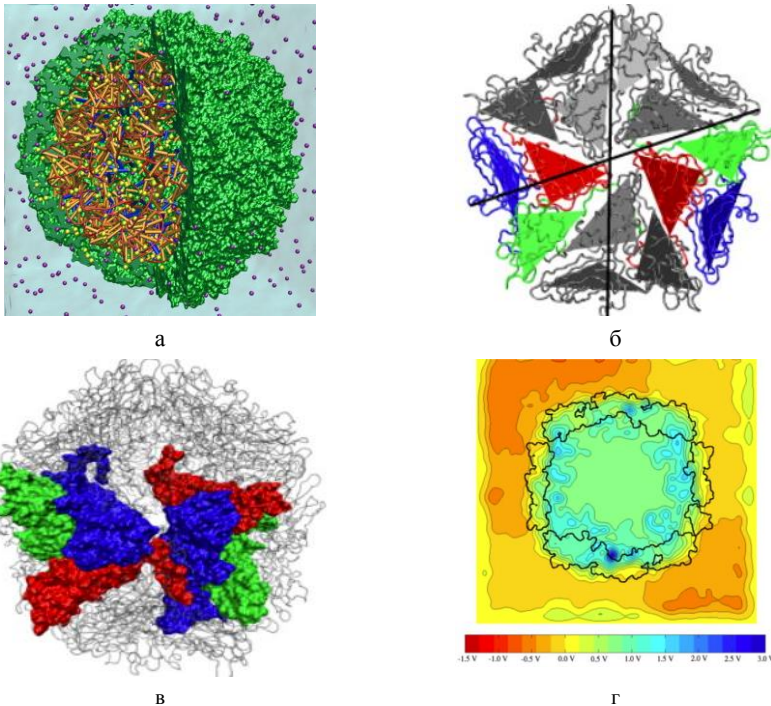


Рис. 39. Результати моделювання ММД однієї частинки (а), повної капсиди у стабільному стані (б), колапс нестабільної капсиди (в), розподіл електричного потенціалу навколо вірусу (г)

4) Моделювання структури і рухів біомолекул.

На сьогодні найпоширенішим застосуванням ММД є вивчення, аналіз та імітація гнучкості, рухів і взаємодій між різними білками. Структури, які визначені експериментальними дослідженнями за допомогою рентгенівської кристалографії або ядерного магнітного резонансу (ЯМР), показують лише середнє наближення того, що може бути реальним, але за допомогою методів обчислювального моделювання можна зробити ще точніше наближення типів структурних флуктуацій, які зазнають молекули. Просто досліджуючи симуляцію цих структур, можна кількісно визначити рухи різних ділянок молекули в стані рівноваги та типи структурних флуктуацій, які відбуваються. Таке моделювання також має здатність демонструвати динамічні поведінкові властивості молекул води та іонів солі, вплив яких часто є критичним для належного функціонування білка, а також для зв'язування ліганду. Як приклад використання ММД на рис. 40 наведені результати моделювання процесу переносу молекули глюкози із внутрішнього середовища клітини в її зовнішнє середовище крізь мембранний білок-канал. Динамічні зміни конфорації (геометричної структури) білка, які забезпечують перенос глюкози або іншої молекули – це канал, відкритий усередину до клітини (а), по центру каналу (б) і назовні до навколишнього середовища (в). На рис. 41 наведені механічні коливання ланцюгів короткої стрічки подвійної спіралі ДНК.

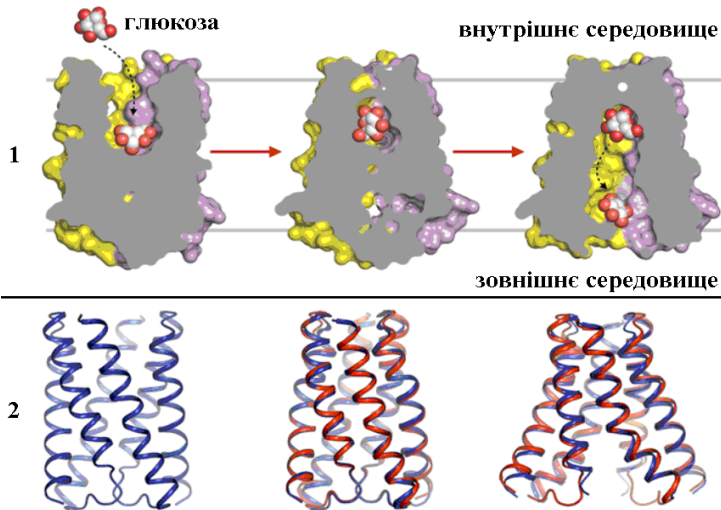


Рис. 40. Результати симуляції ММД динаміки мембранного каналу-переносника під час перенесення молекули глюкози з клітини (а) крізь канал (б) до навколишнього середовища; зображення наведені у форматах прорізна поверхня (1) і пластична стрічка (2), рис. 32

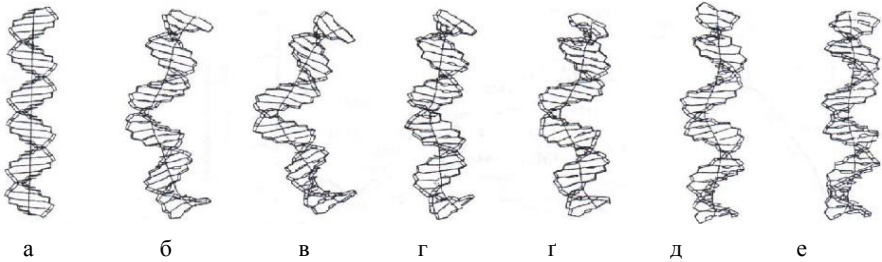


Рис. 41. ММД симуляції коливань стрічки ДНК в моменти часу $t=0$ (а), 500 (б), 1000 (в), 1500 (г), 2000 (г), 2500 (д) і 3000 (е) пікосекунд

5) Оцінка точності моделей структур.

ММД використовується для оцінки точності вже змодельованих структур або для вдосконалення структур, які вже були побудовані в лабораторії. Наприклад, часто експериментально визначені рентгенівські кристалічні структури уточнюються за допомогою обчислювальної моделі ММД. Однією з переваг цього підходу є те, що цей підхід контролює помилки моделі, які присутні. Білкові структури в атомістичних деталях є важливими для повного розуміння біологічних механізмів, але структурна інформація високої роздільної здатності залишається недоступною для багатьох білків. Прогнозування обчислювальної структури має на меті заповнити цю прогалину, але не завжди забезпечує моделі, які досягають експериментальної роздільної здатності. ММД успішно використовується для подолання цієї проблеми. На рис. 42 показаний приклад невідповідності (різних локацій частин молекули) експериментальних даних і структурних моделей для деяких молекул. Цих невідповідностей уникають шляхом уточнення структури за допомогою ММД.

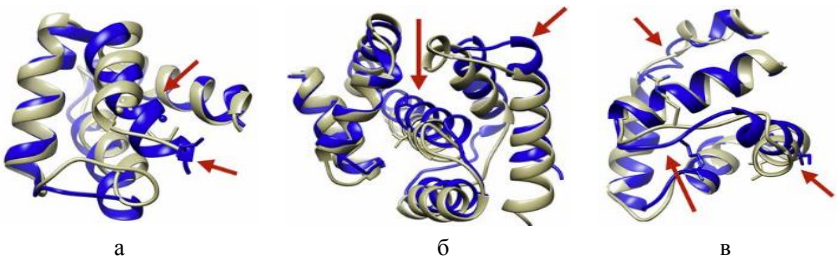


Рис. 42. Приклади невідповідності експериментально визначених структур (сірий колір) і гомологічних структурних моделей (синій колір); локації найбільш значущих невідповідностей показані стрілками

б) Дослідження гнучкості біологічних молекул.

Гнучкість молекул впливає на їх асоціацію з сусідніми атомами, молекулами та іонами, і, таким чином, відіграє активну роль у функціонуванні клітини і клітинних систем. ММД дає чітке уявлення про динамічну еволюцію

будь-якої системи, для якої відомі сили взаємодії між частинками. Цю еволюцію також можна розглядати як певну міру відображення її гнучкості (рис. 43). Нещодавно були розроблені такі методи, як модель анізотропної мережі (ANM), модель пружної мережі (ENM), аналіз головних компонентів (PCA), які дають змогу екстраполювати основні рухи молекулярної системи, що вивчається.

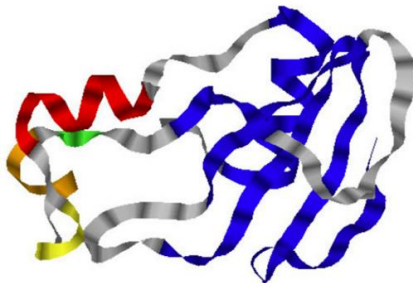


Рис. 43. Стрічкова модель, що показує 5 найбільших жорстких кластерів, які пофарбовані в порядку зменшення жорсткості у синій, червоний, оранжевий, жовтий і зелений колір відповідно; кластери меншого розміру зібрані разом і показані сірим кольором

7) Дослідження стійкості молекулярних систем.

Інше важливе застосування ММД полягає у з'ясуванні механізму, за яким молекулярна система реагує на збурення. Як відомо, збурення у будь-якій фізичній системі поширюються зі швидкостями, які називаються швидкостями звуку (малих коливань). Відомо, що для частинки з масою m , переміщення якої u описується рівнянням руху

$$m\ddot{u} = -ku, \quad (45)$$

де k позначає величину сили, яка повертає частинку до початкового стану (гармонічний осцилятор), частота і період коливань (власні коливання) обчислюються як

$$\omega_0 = \sqrt{\frac{k}{m}}, \quad T_0 = 2\pi\sqrt{\frac{m}{k}}. \quad (46)$$

Для двохатомної молекули з масою $2m$, атоми якої взаємодіють відповідно до потенціалу Ленарда-Джонса, ці величини є

$$\omega_2 = \sqrt{2}\omega_0, \quad T_2 = \frac{T_0}{\sqrt{2}}, \quad (47)$$

а для нескінченної ланцюгової молекули, яка складається s атомів m з тим самим потенціалом,

$$\omega_\infty = 2\omega_0, \quad T_\infty = \frac{T_0}{2}. \quad (48)$$

Для складніших молекулярних систем, періоди і частоти коливань можна облічити тільки за допомогою чисельних методів, у тому числі ММД. Збурення молекулярної системи можуть бути викликані зовнішніми силами, тепловими флуктуаціями, змінами концентрації солі (рН), ліпідного складу, заміною зв'язаного ліганду іншим, зміною амінокислотних залишків білка або їх мутацією, та ін. У всіх таких випадках моделювання ММД допомагає отримати повне розуміння коливального стану і стійкості досліджуваної системи. При такому моделюванні слід виконувати розрахунки кілька разів, використовуючи як збурені, так і незбурені системи, щоб отримати чітке уявлення про постійні відмінності в результатах і, таким чином, підтвердити або спростити свої попередні гіпотези. Як приклад на рис. 44 зображені результати моделювання ММД згинальних коливань вуглецевої нанотрубки.

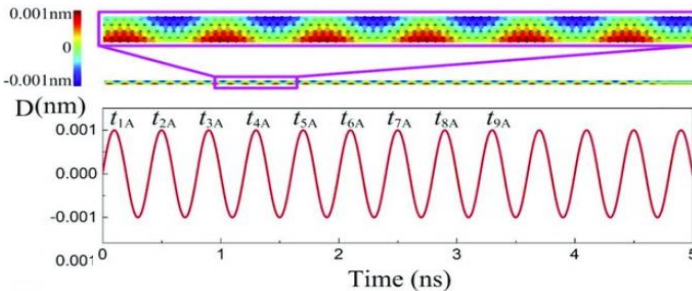


Рис. 44. Поперечні коливання $D(t)$ вуглецевої нанотрубки

8) Дослідження статичної і динамічної рідини і багатофазних систем.

На Рис.45 наведені значення контактних кутів на трифазній границі газ – рідина – тверде тіло. Залежно від здатності твердої поверхні до фізичної взаємодії з молекулами води, можна виділити гідрофільні поверхні, для яких цей контактний кут $\theta < 90^\circ$ (рис. 45а), гідрофобні поверхні $\theta > 90^\circ$ (рис. 45б) і супергідрофобні $\theta > 150^\circ$ (рис. 45с). Останнім часом підвищилась зацікавленість до виготовлення складних за структурою смарт-поверхонь, які мають унікальні фізико-хімічні властивості, такі як відштовхування води, льоду, бруду, біологічних клітин, підвищена прозорість / непрозорість, і т.д. Такі поверхні виготовляють за допомогою сучасних нанотехнологій. Важливою прикладною задачею є комп'ютерний експеримент з обчислення контактного кута в статистиці і динаміці за допомогою ММД. На рис. 46 наведені результати падіння краплі оди на наноструктуровану поверхню з гідрофобного матеріалу.

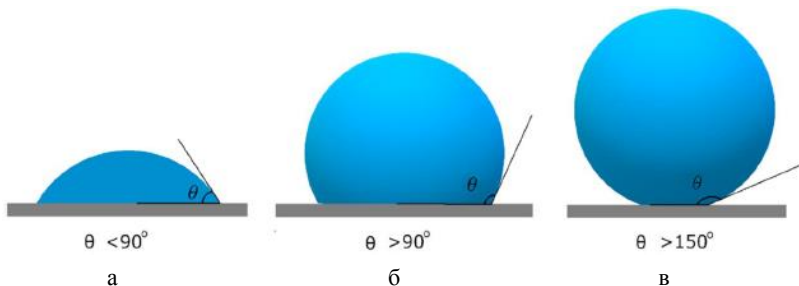


Рис. 45. Крапля води на гідрофільній (а), гідрофобній (б) і супергідрофобній (в) поверхнях

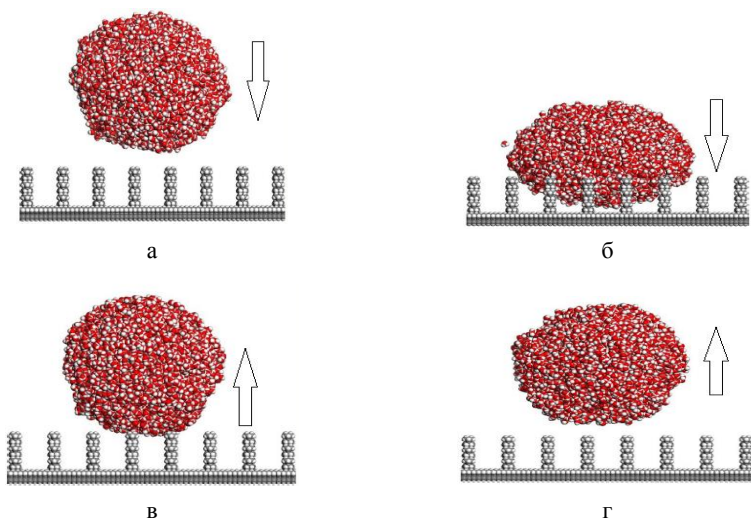


Рис. 46. Результати ММД розрахунків падіння краплі на шорстку поверхню з відштовхуванням: послідовні стадії падіння (а, б) і відштовхування (в, г)

Після обчислення траєкторій руху атомів і утворення стабільної структури, осциляцій частинок і розповсюдження малих збурень, важливим є обчислення геометричних характеристик молекули у просторі – її повного об'єму і площі поверхні, об'єму внутрішніх порожнин і пористості (відношення об'єму порожнин до повного об'єму). Для розрахунку поверхонь молекул були введені різні методи. Загалом молекулярну поверхню можна розглядати як набір сфер, що перекриваються, кожна з яких має радіус Ван-дер-Ваальса свого атома, яка є так званою **поверхнею Ван-дер-Ваальса** (рис. 47а).

Є ще так звана **поверхня молекули, яка доступна для розчинника** (рис. 47б). Вона визначається як ван-дер-ваальсова оболонка молекули, яка

розширена радіусом сфери розчинника навколо кожного центру атома. Тоді, так звана **молекулярна поверхня** (рис. 47в), яка визначається як контур, накреслений сферою радіуса r_p , що представляє молекулу розчинника, що котиться по набору ван-дер-ваальсових кульок, розташованих у центрі. Гладку інваріантну молекулярну поверхню визначають за допомогою згладжувальної сфери з радіусом r_s , що котиться по згенерованій молекулярній поверхні (рис. 28). Молекулярна поверхня містить опуклі сферичні ділянки, сідоподібні тороїдальні ділянки та увігнуті ділянки, які потім поділяються на сітку - набір трикутників (рис. 47г).

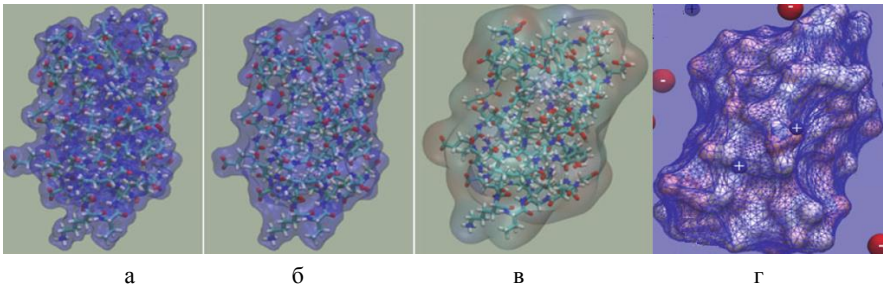


Рис. 47. Різні типи поверхні молекули: Ван-дер-Ваальсова поверхня (а), поверхня, яка доступна для розчинника (б), молекулярна поверхня (в) і триангуляційна сітка поверхні (г)

Таким чином, поверхня, яка доступна для розчинників, не має повторних секцій і містить лише опуклі сферичні плями. Це може дати результат втрати інформації, оскільки відношення контактної поверхні до поверхні повторного входу може бути мірою шорсткості молекулярної поверхні (наявності згладжених опуклих і увігнутих ділянок, які були в початковій моделі). Крім того, вибір радіусу сфери r_s також впливатиме на значення площі молекулярної поверхні в кожному випадку. Використання малого радіусу дозволяє виявити велику кількість особливостей поверхні молекули, а для $r_p = 0$ поверхня, яка доступна для розчинника, дорівнює молекулярній поверхні. Оскільки згладжувальна сфера імітує молекулу розчинника, тобто молекулу води, найменший фізично допустимий радіус становить $1,4 \text{ \AA}$ (радіус молекули води). Зі збільшенням радіуса зонда молекулярна поверхня та поверхня, яка доступна розчиннику, стають більш гладкими, і для $r_p = \infty$ поверхня молекули є опуклою оболонкою набору атомних сфер. З геометричної точки зору молекулярна поверхня прагне до кінцевого граничного значення зі збільшенням радіуса зонда, тоді як поверхня, доступна розчиннику, прагне до нескінченного. Таким чином, молекулярна поверхня може бути кращим зображенням поверхні молекули.

Розрахунок поверхні молекули є геометричною проблемою, і, отже, точність цього обчислення буде залежати від геометричного зображення

поверхні молекули. Для точного представлення поверхні молекули необхідно дотримання кількох умов: максимальна однорідність розподілу точок, згладжена поверхня поблизу особливих точок, мала зміна щільності точок, незалежність від обертання молекули та стабільність при зміні молекулярної маси.

9) **Дослідження динаміки поверхні клітин методом beads&springs.**

Біологічні клітини являють собою певний об'єм простору, який заповнений специфічними для клітини компонентами і органелами і обмежений тонкою багат шаровою гнучкою плівкою (мембраною). Найпоширеніша механічна модель мембрани базується на сукупності сферичних частинок (молекул мембрани) у вигляді твердих кульок (beads) і сил зв'язку між ними у вигляді пружин (springs). У випадку еритроцита загальна поверхня є двояко-увігнутою (рис. 48а), так що у перерізі периметр мембрани $2\pi r$ обмежує поверхню площиною $3\pi r^2/5$. Для моделювання динамічної поведінки мембрани у відповідь на механічні сили, які діють на еритроцити в течії крові, периметр замінюється на n-сторонній багатокутник із частинками у його вершинах, з рівними відстанями l_0 між частинками і рівними кутами α_0 (рис. 48а). Сторони багатокутника є пружинами, які відповідають на зовнішні сили відповідно до закону Гука для пружного тіла. За наявності поверхневих (тангенціальних) сил вздовж поверхні клітини виникає подовження мембрани і сили пружності (рис. 48б), які повертають її до попереднього стану.

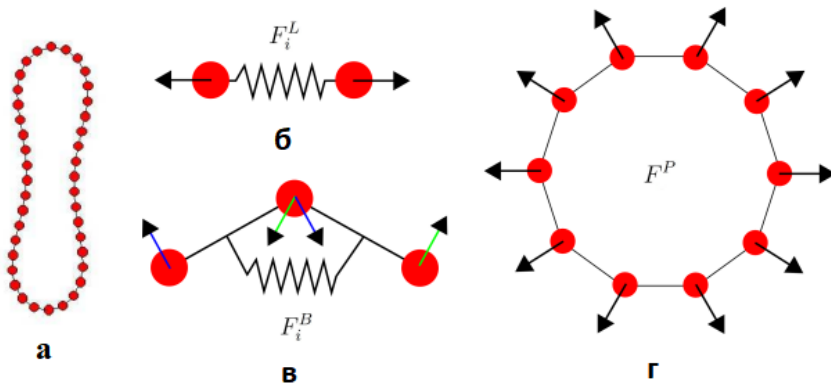


Рис. 48. Модель мембрани еритроцита (а) і сил розтягування (б), згину (в) і внутрішнього тиску (г)

$$F_{j,j+1}^L = k_{Lj} \left(1 - \frac{l_{j,j+1}}{l_0} \right), \quad (49)$$

де $l_{j,j+1}$ – відстань між частинками j і $j+1$, k_{Lj} – коефіцієнт пружності пружини між частинками j і $j+1$.

Якщо виникає локальний вигін мембрани (рис. 48в), з'являються відповідні пружні сили

$$F_j^B = k_{Bj} \left(1 - \frac{\alpha_j}{\alpha_0} \right), \quad (50)$$

де α_j – кут між частинками $j-1$, j і $j+1$, k_{Bj} – коефіцієнт згинальної пружності навколо частинки j .

Коли біологічна клітина поглинає або виділяє воду, наприклад, за рахунок осмотичних процесів, в клітині відповідно зростає або спадає гідростатичний тиск і з'являються нормальні до поверхні сили, які розтягують мембрану (рис. 48 г)

$$F_j^P = k_{pj} \left(1 - \frac{A_j}{A_0} \right), \quad (51)$$

де A_j і A_0 – нова і початкова локальні площі поверхні навколо частинки j , k_{pj} – коефіцієнт пружності поверхні навколо частинки j .

Чисельні розрахунки ММД [39] з використанням (49)-(50) дозволяють описати деформації еритроциту у потоці рідини з початкової рівноважної (рис. 48а) до парашутоподібної (рис. 49).

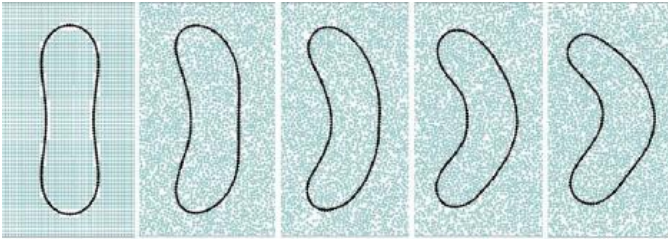


Рис. 49. Поступові зміни форми еритроциту за рахунок гідродинамічних сил

6. ПРИКЛАДИ ЗАДАЧ ДЛЯ САМОСТІЙНОГО РОЗВ'ЯЗАННЯ ЗА ДОПОМОГОЮ ММД

Задача 1. Розглядається одноатомна рідина Ленарда–Джонса, в якій діє тільки потенціал LJ у вигляді (26) або (28). Задаємо систему із 500–1000 сферичних частинок з діаметрами d_i (рис. 50). За допомогою генератора випадкових чисел задаємо координати центрів частинок з перевіркою відсутності пересічень об'ємів частинок. Залежно від заданої середньої відстані L_{ij} між парами частинок $\{i,j\}$ можна згенерувати газ ($L_{ij} \gg d_i$), рідину ($L_{ij} \geq d_i$), аморфне тверде тіло ($L_{ij} \sim d_i$, значення L_{ij} нерегулярні) або кристал (регулярні значення координат центрів частинок d_i в просторі).

Задача 2. Моделювання тертя на поверхнях розділу твердих тіл.

Розглядається система атомів з потенціалом Ленарда–Джонса (26) або (28). Задаємо систему із 500–1000 сферичних частинок з діаметрами d_i , які утворюють дві підсистеми щільних упакувань з різними коефіцієнтами моделі (26)–(28) (зображені на рис. 51 сірим і чорним кольором, відповідно). На граничних поверхнях задаємо «зубчики» у вигляді випадково відсутніх частинок або фрактальних структур. До лівого краю верхньої і до правого краю нижньої підсистеми прикладаємо зсувні сили τ , які направлені в протилежні напрямки, що імітує тертя поверхонь. Оцінити сили взаємодії на границі і динаміку пошкодження поверхонь при різних значеннях параметрів моделі і значеннях τ .

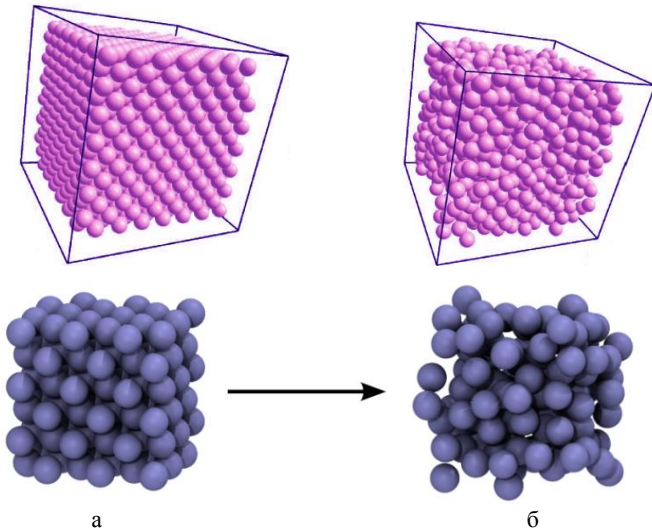


Рис. 50. Початкова кристалічна структура (а) одноатомної речовини змінюється на аморфну за рахунок топлення під час нагріву середовища (б)

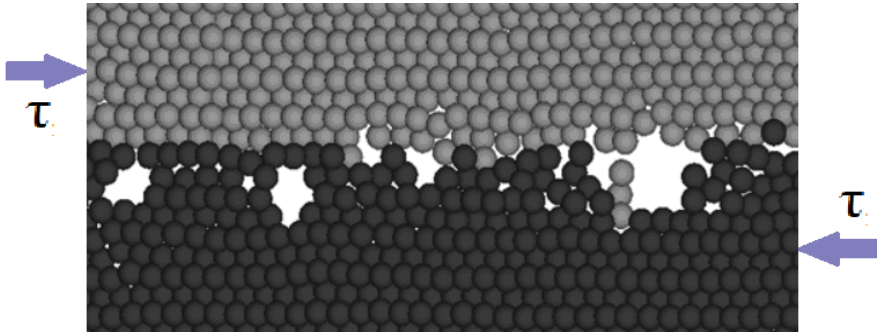


Рис. 51. Схема початкового розподілу частинок до задачі 2

Задача 3. Імпульсна (ударна) взаємодія твердих тіл.

Вихідні дані як в попередній задачі, але частинки утворюють дві паралельні поверхні, нижня з яких лежить на твердій поверхні (вертикальні переміщення частинок нижнього шару дорівнюють нулю), а верхня вільно падає в полі тяжіння із заданої висоти h (рис. 52).

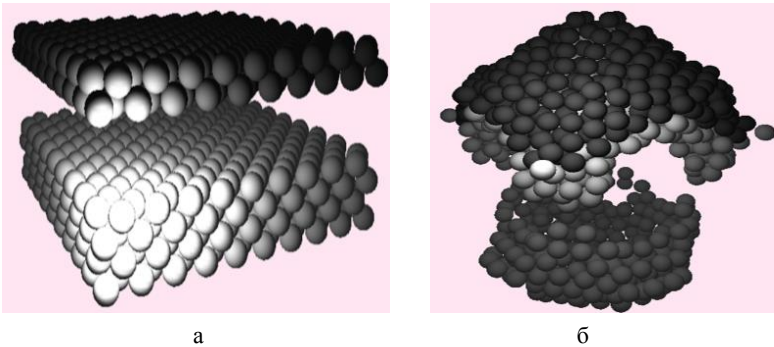


Рис. 52. Початкова геометрія до задачі 3 (а); деформація і руйнування матеріалу (б)

Задача 4. Візуалізація мікроструктур.

Дослідити щільне пакування системи сферичних частинок (500–1000 шт.) кількох типів (1–3) з різними властивостями і діаметрами (рис. 53а,б). Оцінити швидкість поширення малих збурень положення частинок і, таким чином, відносну стабільність пакування.

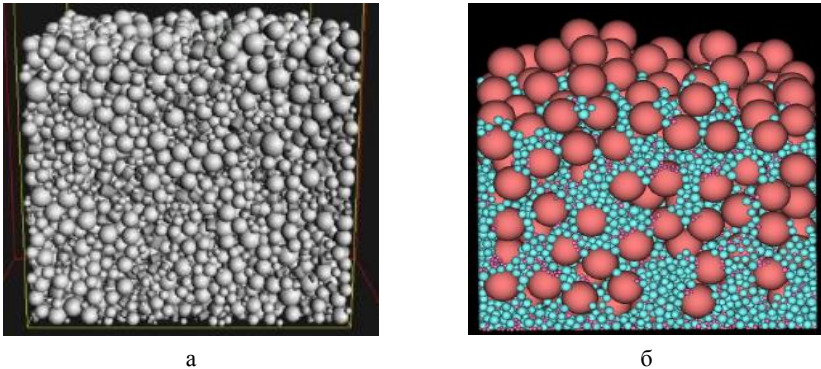


Рис. 53. Візуалізація сукупності частинок з подібними (а) і різними (б) діаметрами в полі сили тяжіння

Задача 6. Моделювання дифузії молекулярного газу в заданому об'ємі.

Задача 7. Математичне моделювання рівноважного стану ідеального газу.

Задача 8. Дослідження перепаду тиску температури газу до і після проходження через мембрану залежно від структурних характеристик мембрани (рис. 54).

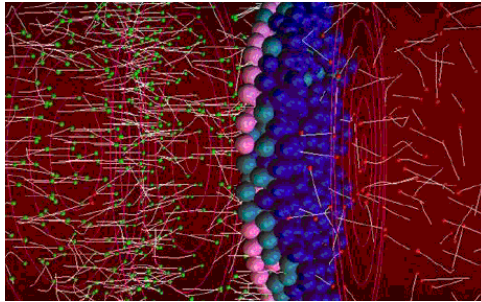
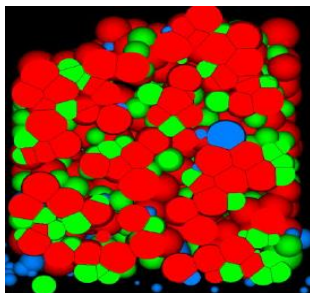


Рис. 54. Приклад ММД-моделювання дифузії частинок крізь пористу мембрану

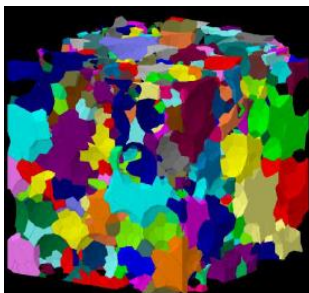
Задача 9. Моделювання процесу самоскладання ансамблів частинок у краплі води.

Задача 10. Моделювання адсорбції та дифузії газу в мікропористих структурах. Візуалізація і обчислення параметрів пористого простору.

Задача 11. Моделювання спікання системи сферичних частинок у композитний матеріал за рахунок фазового переходу при заданому підвищенні температури (рис. 55).



а



б

Рис. 55. Система сферичних частинок 3-х типів (а)
і результат їх спікання після підвищення температури (б)

ПЕРЕЛІК ПОСИЛАНЬ

1. Кізілова Н. М. Метод скінченних елементів у розв'язанні задач механіки рідини і газу : методичні рекомендації до практичних занять та самостійної роботи з курсу «Чисельні методи механіки суцільних середовищ» для студентів спеціальності «Прикладна математика». Харків: ХНУ імені В. Н. Каразіна, 2020. 64 с.
2. Кізілова Н. М. Метод скінченних елементів в розв'язанні прикладних задач механіки рідини з AnSys Rluent 2021R1. Харків: ХНУ імені В. Н. Каразіна, 2021. 80 с.
3. Кізілова Н. М. Прикладні задачі сучасної мікрофлюїдики і нанофлюїдики : методичні рекомендації до практичних занять з курсу «Наномеханіка та сучасні нанотехнології» для студентів спеціальності «Прикладна математика». Харків: ХНУ імені В. Н. Каразіна, 2020. 36 с.
4. Elzaabalawy A., Meguid S.A. Advances in the development of superhydrophobic and icephobic surfaces. *International Journal of Mechanics and Materials in Design*, 2022. Vol. 18. P. 509–547. URL: <https://doi.org/10.1007/s10999-022-09593-x>
5. Critical Assessment of protein Structure Prediction URL: https://en.wikipedia.org/wiki/critical_assessment_of_protein_structure_prediction
6. Kinoshita H., Yoshida H., Nakai H. Symplectic integrators and their application to dynamical astronomy. *Celestial Mechanics and Dynamical Astronomy*, 1990. Vol. 50. P. 59–71. URL: <https://doi.org/10.1007/BF00048986>
7. Кізілова Н. М. Основи біомеханіки. Методичні рекомендації для студентів. Харків: ХНУ імені В. Н. Каразіна, 2022. 160 с.
8. Israelachvili J. N. *Intermolecular and Surface Forces*. 3-rd edition. Elsevier. 2011. URL: <https://doi.org/10.1016/C2009-0-21560-1>
9. Erginson C., Vineyard G.H., Englert A. Dynamics of Radiation Damage in a Body-Centered Cubic Lattice. *Rhysical Review*, 1964. Vol. 133, N2A. P. 595 – 606. URL: <https://doi.org/10.1103/PhysRev.133.A595>
10. Kamberaj H. *Molecular Dynamics Simulations in Statistical Physics*. Springer, 2020.
11. Sushko G.B., Solov'yov I.A., Solov'yov A.V. Molecular dynamics for irradiation driven chemistry: application to the FEBID process. *Eur. Phys. J. Ser.D*. 2016. Vol. 70. P. 217. URL: <https://doi.org/10.1140/epjd/e2016-70283-5>
12. Chemistry at HARvard Macromolecular Mechanics URL: <https://www.charmm.org>
13. GROMACS URL: <https://www.gromacs.org>
14. GROMOS URL: <https://www.gromos.net>
15. URL: <https://www.igc.ethz.ch/gromos.html>
16. URL: <https://ambermd.org>
17. URL: <https://www.lammps.org>

18. URL: <http://www.ks.uiuc.edu/namd>
19. URL: <http://molecularsystemsengineering.org/saft.html>
20. URL: <http://www.sirahff.com>
21. Ribeiro J.V., Bernard R.C., Rudack T. QwikMD – integrative molecular dynamics toolkit for novices and experts. Scientific Reports, 2016. Vol. 6(1). 26536. URL: <https://doi.org/10.1038/srep26536>
22. URL: <http://www.biomolecular-modeling.com/Abalone>
23. URL: <https://avogadro.cc>
24. URL: <https://ball-project.org>
25. URL: <https://ball-project.org/ballview>
26. URL: <https://www.rbvi.ucsf.edu/chimera>
27. URL: <https://gabedit.sourceforge.net>
28. URL: <https://jmol.sourceforge.net>
29. URL: <http://www.fos.su.se/~sasha/mdynamix>
30. URL: <https://www.theochem.ru.nl/molten>
31. URL: <https://ugovaretto.github.io/molekel>
32. URL: <https://www.ms-2.de/home.html>
33. URL: <https://sites.google.com/a/opensource.lk/ninithi/?title=Ninithi>
34. URL: <https://nwchemgit.github.io>
35. URL: <https://qtemol.sourceforge.net>
36. URL: <http://www.rasmol.org>
37. URL: <http://pymol.org/2>
38. URL: <https://pymol.org/dokuwiki/doku.php?id=tutorials>
39. Tosenberger A., Salnikov V., Bessonov N., et al. Particle dynamics methods of blood flow simulations. Mathematical Modelling of Natural Phenomena, 2011. Vol. 6. P. 320-332. URL: <https://doi.org/10.1051/mmnp/20116512>