Методы и программная инфраструктура для решения задачи молекулярной конформации

Власов Т.Е.

Аннотация **- Рассматривается задача отыскания геометрической структуры кластера одинаковых атомов, взаимодействие которых описывается парными потенциалами. Применяемые для расчета методы стохастической оптимизации характеризуются высокой ресурсоемкостью, которая затрудняет их применение для анализа соединений с большим количеством атомов. Предлагается программная инфраструктура для реализации стохастических методов. Приводятся результаты вычислительных экспериментов, подтверждающие эффективность предложенных подходов при решении задачи поиска конформации молекулярного кластера с минимальной энергией. Исследуется влияние различных параметров на скорость и точность расчетов.**

I. Введение

Кластерами принято называть группы близко расположенных, тесно связанных друг с другом атомов, молекул, ионов. Изучение кластеров имеет важнейшее значение для понимания процессов конденсации, расчета электронных и динамических характеристик наноматериалов, создания новых источников света и многих других областей[1]. Одной из фундаментальных задач данного направления является определение геометрической структуры, или, как иногда говорят, конформации кластера, соответствующей минимальной энергии взаимодействия входящих в него частиц. Такие конформации наиболее часто наблюдаются в веществе при определенных условиях, например во время перехода из одного состояния в другое.

Широко используется модель, при учитываются только парное взаимодействия частиц, входящих в кластер. Энергия взаимодействия двух частиц задается парным потенциалом: функцией одного аргумента, которая определяет зависимость энергии взаимодействия от расстояния между частицами. При этом энергия взаимодействия всех частиц кластера определяется как сумма энергии парных взаимодействий входящих в него частиц:

$$E(x) = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} v(r_{ij})$$

где r_{ij} - расстояние между частицами i и j, а вектор x содержит декартовых координат этих частиц. Так как эта модель обычно применяется для моделирования химических веществ, состоящих из одинаковых атомов, то частицы кластера будем также называть атомами. При исследовании различных веществ и процессов применяются различные потенциалы. В данной работе ограничимся рассмотрением трех распространенных потенциалов: Леннарда-Джонса, Морзе и Дзугутова, характеристики и графики которых приведены в таблице 1. Эти потенциалы применяются для моделирования структуры различных химических веществ. Кластеры Леннарда-Джонса встречаются в инертных газах, Дзугутова – в структурах металлов и стекол. Параметрρ в потенциале Морзе позволяет моделировать различные вещества.

$$v_m(r) = e^{\rho(1-r)}\left(e^{\rho(1-r)} - 2\right)$$

Нахождению глобального минимума функции посвящено значительное число работ. Существует база данных[2], в которой перечислены наименьшие найденные значения энергии и соответствующие конформации для различного числа атомов и разных потенциалов взаимодействия. Для нахождения минимума функции применялись различные методы, которые можно условно разделить на две категории. К первой относятся подходы, не использующие специфичные для данной задачи свойства целевой функции, т.е. неспециализированные методы оптимизации. Во вторую категорию входят методы, использующие специфику задачи. Начнем с рассмотрения второй категории. Эти методы основаны на общих геометрических закономерностях, наблюдаемых для конформаций с минимальной энергией.

Несмотря на высокую эффективность геометрически-обоснованных методов, они имеют ряд существенных недостатков. Во-первых, данные методы ориентированы на достаточно узкий класс задач, поэтому для каждой рассматриваемой модели взаимодействия требуется разрабатывать свой метод. Во-вторых, априорные предположения о геометрической природе кластера могут стать причиной ошибок в нахождении минимумов, что было продемонстрировано в работе[3]. Поэтому, большое внимание уделяется также решению данной задачи неспециализированными оптимизационными методами. Считается, что такие методы обязательно должны применяться для верификации результатов проблемно-ориентированной оптимизации, основанной на геометрических соображениях.

К сожалению, попытки решения рассматриваемой задачи методами, гарантирующими глобальную оптимальность, не увенчались успехом для кластеров из 8 и более атомов. Поэтому в качестве наиболее перспективных подходов к решению данной задачи рассматриваются различные эвристические алгоритмы. Одним из наиболее известных алгоритмов, хорошо зарекомендовавших себя при решении задачи поиска

Власов Т.Е. Московский государственный университет, Москва (zasalamel@ya.ru)

конформации с минимальной энергией взаимодействия, является локально-стохастический метод Monotonic Sequence Basin-Hoping (MSBH). Суть подхода состоит в комбинации незначительных сдвигов в пространстве допустимых решения и локального поиска. Более подробно этот алгоритм рассматривается далее.

II. Постановка задачи и метод решения

Задача поиска структуры кластера с минимальной энергией взаимодействия формулируется как задача безусловной оптимизации: требуется найти минимум функции (1) в евклидовом пространстве размерности $n = 3N$, где N - число атомов в кластере. Аналитические выражения для градиента и гессиана целевой функции могут быть легко получены.

Рассматриваемый в данной работе алгоритм MSBH работает по следующей схеме. В качестве начальной выбирается точка x_0 локального минимума функции. На каждом шаге алгоритма текущая точка локального минимума x подвергается возмущению Φ , к результату которого применяется локальный алгоритм $L_f : \tilde{x} = L_f(\Phi(x))$. Если, $f(\tilde{x}) < f(x)$ то точка хзаменяется на $\tilde{x}$. Далее процесс повторяется. Алгоритм останавливается, когда после заданного числа N_{max} итераций не удается улучшить найденный локальный минимум. Оператор возмущения изменяет координаты точки в пределах некоторой окрестности: $\Phi(x) = x + \xi, \quad \xi_i \in [-r, r], \quad i = 1, \cdots, n.$ Величины ξ_i генерировались как равномерно-распределенные случайные числа в диапазоне $[-r, r]$. Возможно применение других законов распределения. Основными параметрами алгоритма MBH являются радиус окрестности просмотра r и количество N_{max} итераций при генерации случайных точек. Радиус окрестности просмотра r должен быть достаточно небольшим, чтобы допускать эффективный перебор с помощью оператора возмущения в окрестности радиуса r . Эти параметры подбираются экспериментально.

На практике алгоритм MSBH чаще всего применяется вместе с каким-либо алгоритмом, генерирующим начальные приближения – например, методом Монте-Карло. Мы применяли вариант, предложенный в работе[4], в котором некоторое количество начальных приближений генерируется случайно в кубе $[-1,1]^n$. Далее сгенерированные точки используются в качестве начальных для MSBH. Такой алгоритм идеально подходит для реализации в среде параллельных и распределенных вычислений, так как различные начальные приближения могут обрабатываться независимым образом.

III.Основы генетических алгоритмов

Для задачи в статье был использован ряд генетических алгоритмов. В этом разделе будут даны общие сведения о них.

Генетический алгоритм — это эвристический алгоритм поиска, используемый для решения задач оптимизации и моделирования путём случайного подбора, комбинирования и вариации искомых параметров с использованием механизмов, напоминающих биологическую эволюцию. Является разновидностью эволюционных вычислений, с помощью которых решаются оптимизационные задачи с использованием методов естественной эволюции, таких как наследование, мутации, отбор и кроссинговер[7].
Отличительной особенностью генетического алгоритма является акцент на использование оператора «скрещивания», который производит операцию рекомбинации решений-кандидатов, роль которой аналогична роли скрещивания в живой природе.

Задача формализуется таким образом, чтобы её решение могло быть закодировано в виде вектора («генотипа») генов, где каждый ген может быть битом, числом или неким другим объектом. В классических реализациях ГА предполагается, что генотип имеет фиксированную длину. Однако существуют вариации ГА, свободные от этого ограничения.

Некоторым, обычно случайным, образом создаётся множество генотипов начальной популяции. Они оцениваются с использованием «функции приспособленности», в результате чего с каждым генотипом ассоциируется определённое значение («приспособленность»), которое определяет насколько хорошо фенотип, им описываемый, решает поставленную задачу.

При выборе «функции приспособленности» (или fitness function в англоязычной литературе) важно следить, чтобы её «рельеф» был «гладким».

Из полученного множества решений («поколения») с учётом значения «приспособленности» выбираются решения (обычно лучшие особи имеют большую вероятность быть выбранными), к которым применяются «генетические операторы» (в большинстве случаев «скрещивание» — crossover и «мутация» — mutation), результатом чего является получение новых решений. Для них также вычисляется значение приспособленности, и затем производится отбор («селекция») лучших решений в следующее поколение.

Этот набор действий повторяется итеративно, так моделируется «эволюционный процесс», продолжающийся несколько жизненных циклов (*поколений*), пока не будет выполнен критерий остановки алгоритма. Таким критерием может быть:

- нахождение глобального, либо субоптимального решения;
- исчерпание числа поколений, отпущенных на эволюцию;
- исчерпание времени, отпущенного на эволюцию.

Генетические алгоритмы служат, главным образом, для поиска решений в многомерных пространствах поиска.

Таким образом, можно выделить следующие этапы генетического алгоритма:

1. Задать целевую функцию (приспособленности) для особей популяции
2. Создать начальную популяцию

- (Началоцикла)

1. Размножение (скрещивание)
2. Мутирование

3. Вычислить значение целевой функции для всех особей
4. Формирование нового поколения (селекция)
5. Если выполняются условия остановки, то (конец цикла), иначе (начало цикла).

Создание начальной популяции.

Перед первым шагом нужно случайным образом создать начальную популяцию; даже если она окажется совершенно неконкурентоспособной, вероятно, что генетический алгоритм все равно достаточно быстро переведет её в жизнеспособную популяцию. Таким образом, на первом шаге можно особенно не стараться сделать слишком уж приспособленных особей, достаточно, чтобы они соответствовали формату особей популяции, и на них можно было подсчитать функцию приспособленности (Fitness). Итогом первого шага является популяция H, состоящая из N особей.

Размножение (Скрещивание).

Размножение в генетических алгоритмах обычно половое — чтобы произвести потомка, нужны несколько родителей, обычно два.

Размножение в разных алгоритмах определяется по-разному — оно, конечно, зависит от представления данных. Главное требование к размножению — чтобы потомок или потомки имели возможность унаследовать черты обоих родителей, «смешав» их каким-либо способом.

Почему особи для размножения обычно выбираются из всей популяции H, а не из выживших на первом шаге элементов H_0 (хотя последний вариант тоже имеет право на существование)? Дело в том, что главный бич многих генетических алгоритмов — недостаток разнообразия (diversity) в особях. Достаточно быстро выделяется один-единственный генотип, который представляет собой локальный максимум, а затем все элементы популяции проигрывают ему отбор, и вся популяция «забивается» копиями этой особи. Есть разные способы борьбы с таким нежелательным эффектом; один из них — выбор для размножения не самых приспособленных, но вообще всех особей.

Мутации.

К мутациям относится все то же самое, что и к размножению: есть некоторая доля мутантов m, являющаяся параметром генетического алгоритма, и на шаге мутаций нужно выбрать mN особей, а затем изменить их в соответствии с заранее определёнными операциями мутации.

Отбор.

На этапе отбора нужно из всей популяции выбрать определённую её долю, которая останется «в живых» на этом этапе эволюции. Есть разные способы проводить отбор. Вероятность выживания особи h должна зависеть от значения функции приспособленности Fitness(h). Сама доля выживших s обычно является параметром генетического алгоритма, и её просто задают заранее. По итогам отбора из N особей популяции H должны остаться sN особей, которые войдут в итоговую популяцию H'. Остальные особи погибают.

IV. Построение распределённой системы.

В ходе написания статьи был реализован прототип распределённой системы. Прототип был реализован на языке программирования Perl. Система состоит из двух основных компонент:

1. Генератор заданий.
2. Система запуска заданий.

Генератор заданий служит для анализа уже полученных результатов, и генерации заданий на их основе уже полученных для их дальнейшей обработки. Основной скрипт принимает в качестве параметра json - файл в котором указан, алгоритм генерации, а также конфигурация плагина.

На момент написания статьи были реализованных 4 плагина:

1. Случайной генерации.
2. Генерации с выбором наилучших представителей.
3. Генерации на основе генетического подхода 1.
4. Генерации на основе генетического подхода 2

Случайный генератор наиболее простой из плагинов, результат его работы является некоторое количество случайно сгенерированных случайно в кубе $[-1,1]^n$.

Генератор с выбором наилучших на первой итерации генерирует некоторое количество заданий. Далее генерация заданий производился в несколько итераций по одинаковому числу заданий в каждой итераций.

Работа генератора на основе первого генетического подхода на первой итерации повторяет работу предыдущего генератора: генерирует некоторое количество случайных заданий, затем ожидает результата их обработки. На каждой следующей итерации генератор выбирает множество наилучших представителей с предыдущей итерации, затем из этого множества выбирает несколько случайных пар молекул Для каждой пары производится слияние молекул. Во первых перед слиянием молекулы нормализуются. Процесс нормализации состоит из двух этапов: центрирование и вращения. Центрирование - это сдвиг относительно центра массы молекул. Вращение - это направление с самым дальним из центра масс молекулы относительно оси OX , и вторым по дальности вектором относительно OY . Ясно, что после этих двух преобразований энергия молекулы не изменится. После нормализации происходит процесс слияния. Выбирается граница слияния n и из первой молекулы выбирается n самых близких атомов, к центру массы молекулы, из второй молекулы - n атомов, наиболее далёких от центра массы молекулы.

Работа генератора на основе второго генетического подхода во много повторяет работу предыдущего генератора. Разница начинает проявляться в момент слияния. Как и в предыдущем случае, лучшие молекулы проходят процедуру нормализации. Вторым этапом процедуры нормализации является не вращение, а упорядочивание относительно плоскости, относительно которой, в последствии, будет происходить разделение (в данной работе взята плоскость OX). При слиянии же получается молекула, одна часть которой берётся от первой молекулы, с одной стороны плоскости

разделения, другая от второй молекулы с другой стороны разделения стороны плоскости

Система запуска заданий служит для обработки ранее сгенерированных заданий. Конфигурация алгоритма хранится в json формате. Основным параметром служит алгоритм запуска (настраивается в качестве плагина). Под алгоритмом запуска понимается, некоторый менеджер, который решает на какой машине и в какой момент времени будут запущены задания.

V. Результаты вычислительных экспериментов

Вычислительные эксперименты проводились для потенциалов Морзе (таблица 1). В таблице 1 приведено сравнение четырёх версий распределённого алгоритма MSBH. Вычисления производились на стационарном компьютере с процессором IntelCore 2 Duo 2.53 Hzx 2

№	Генератор 1	Генератор 2	Генератор 3	Генератор 4
1	-194.962	-197.907	-195.462	-196.456
2	-150.668	-152.334	-150.573	-150.774
3	-106.863	-106.863	-106.863	-106.863
4	-241.824	-243.824	-241.956	-242.519
5	-138.69	-138.69	-138.69	-138.69

Таблица 1. Значения глобальных минимумов для потенциала Морзе.

Генератор1 - Случайной генерации.
Генерации2 - Генератор с выбором наилучших представителей.
Генератор3 - Генерации на основе первого генетического подхода.
Генератор4 - Генерация на основе второго генетического подхода.

Эксперимент 1 - N = 50, eps = 0.8, 1000 итераций
Эксперимент 2 - N = 40, eps = 0.8, 1000 итераций
Эксперимент 3 - N = 30, eps = 0.8, 1000 итераций
Эксперимент 4 - N = 60, eps = 0.8, 500 итераций
Эксперимент 5 - N = 37, eps = 0.8, 1000 итераций

Из вышеприведенных данных видно, что распределённый алгоритм MSBH, в основе которого лежит принцип выбора наилучшего элемента работает лучше относительно обоих генетических подходов, предложенных в статье. С другой стороны генетический алгоритм оправдал себя в сравнении со случайным алгоритмом.

VI. Заключение

В статье рассмотрена распределенная программная инфраструктура для решения задач конечномерной оптимизации, которая позволяет объединять разрозненные разнородные суперкомпьютерные ресурсы в единое вычислительное пространство.

Входе написания статьи был реализован прототип распределённой системы на языке программирования Perl. С помощью этой системы был поставлен ряд экспериментов, направленный на улучшение текущей реализации распределенной версии алгоритма MSBH. В основе улучшения MSBH лежат принципы генетических алгоритмов.

Численные эксперименты показали, что улучшение алгоритма MSBH, основанного на основе генетического подхода, занимает промежуточное положение между ранее существующими алгоритмами, основанными на выборе наилучшего элемента и случайного выбора.

Библиография

[1] Елецкий А.В. «Экзотические» объекты атомной физики // Соросовский образовательный журнал, № 4, 1995, С. 86-95.

[2] The Cambridge Cluster Database, D. J. Wales, J. P. K. Doye, A. Dullweber, M. P. Hodges, F. Y. Naumkin F. Calvo, J. Hernández-Rojas and T. F. Middleton, [http://www-wales.ch.cam.ac.uk/CCD.html]

[3]Leary, R. H. 1997. Global Optima of Lennard-Jones Clusters. // Journal of Global Optimization 11, 1 (Jul. 1997), P. 35-53.

[4] Leary, R. H. 1997. Global Optima of Lennard-Jones Clusters. // Journal of Global Optimization 11, 1 (Jul. 1997), P. 35-53.

[5] А.П. Афанасьев, М.А.Посыпкин, И.Х. Сигал, Проект BNB-Grid: решение задач глобальной оптимизации в распределенной среде // Труды второй международной конференции "Системный анализ и информационные технологии" САИТ-2007, том. 2, с. 177-181.

[6] Посыпкин М.А. Архитектура и программная организация библиотеки для решения задач оптимизации методом ветвей и границ на многопроцессорных вычислительных комплексах // Проблемы вычислений в распределен-ной среде: распределенные приложения, коммуникационные системы, ма-тематические модели и оптимизация. Труды ИСА РАН. - М.: КомКнига, 2006,С. 18-25.

[7] Генетический алгоритм
[http://en.wikipedia.org/wiki/Genetic_algorithm]