

О сочетании венгерского алгоритма и алгоритма Кабша решения задач вычислительной геометрии

И.А. Блатов¹, Е.В. Китаева²

¹Поволжский государственный университет телекоммуникаций и информатики, Льва Толстого 23, Самара, Россия, 443010

²Самарский национальный исследовательский университет им. академика С.П. Королева, Московское шоссе 34А, Самара, Россия, 443086

Аннотация. Рассматривается задача определения минимума суммы квадратов расстояний между двумя системами точек в пространстве. Необходимость решения такой задачи возникает в компьютерной химии и при распознавании образов. Грубый алгоритм имеет факториальную сложность. Предлагается алгоритм, позволяющий найти решение с заданной точностью ε за $O(n^3 / \varepsilon^{3/2})$ арифметических операций.

1. Введение

Пусть в пространстве заданы две системы точек M и N , каждая из которых содержит n точек. Рассматривается задача отыскания такого положения этих систем в пространстве и паросочетания их вершин, для которых сумма квадратов попарных расстояний $\rho(M, N)$ между точками M и N была бы минимальной.

В случае фиксированного паросочетания данная задача решается за линейное ($O(n)$) количество операций алгоритмом Кабша [1]–[2]. Грубый поиск оптимального паросочетания требует перебора $n!$ различных положений. В случае фиксированного положения в пространстве систем M и N известен венгерский алгоритм [3] отыскания оптимального паросочетания за $O(n^3)$ операций. Однако неизвестен [4] алгоритм общего назначения, позволяющий за полиномиальное время найти оптимальное паросочетание и соответствующее $\rho(M, N)$. Различные варианты алгоритмов, сочетающих метод Кабша и венгерский метод, рассматривались в [5]–[7] (см. также библиографию в [6]–[7]). Однако это либо эвристические алгоритмы, не гарантирующие отыскание глобального минимума, либо алгоритмы, использующие взаимозависимость и специальные свойства сравниваемых структур, либо экспоненциально или факториально сложные алгоритмы. Авторы [6] утверждают, что предложенный ими алгоритм Go-PERMDIST имеет полиномиальную сложность, но не приводят точных оценок вычислительной сложности.

В настоящей статье для решения данной задачи рассматриваются три алгоритма общего назначения, основанные на алгоритмах Кабша и венгерском алгоритме, позволяющие найти решение поставленной задачи с заданной точностью ε за $O(n^3 / \varepsilon^{3/2})$ арифметических операций. Проведен сравнительный экспериментальный анализ этих алгоритмов. Первый (грубый) алгоритм представляет собой сочетание венгерского алгоритма с перебором по узлам

\mathcal{E} –сети в пространстве параметров изометрий в R^3 . Второй алгоритм (комбинированный) отличается от первого последовательным выполнением венгерского алгоритма и алгоритма Кабша вместо одного лишь венгерского алгоритма. Третий алгоритм (итерационный) сочетает в себе венгерский алгоритм, алгоритм Кабша и итерационный метод покоординатного спуска в пространстве параметров изометрий в R^3 . Проведен сравнительный экспериментальный анализ этих алгоритмов. Показана высокая эффективность второго алгоритма по сравнению с первым и третьим.

2. Постановка задачи и алгоритмы

Пусть $M = \{\vec{M}_i(x_i, y_i, z_i), 1 \leq i \leq n\}$, $N = \{\vec{N}_i(x_i, y_i, z_i), 1 \leq i \leq n\}$ – две совокупности точек (радиус-векторов) в R^3 , заданных координатами относительно прямоугольной декартовой системы координат $OXYZ$. В дальнейшем знак вектора писать не будем. Через I обозначим единичную матрицу третьего порядка. Если $w = (w_1, w_2, w_3)$ вектор в R^3 , то $\|w\| = \sqrt{w_1^2 + w_2^2 + w_3^2}$ обозначает его евклидову норму. Через C, C_1, C_2, \dots будем обозначать положительные константы, не зависящие от n . Если это не вызывает недоразумений, то одним и тем же символом C могут обозначаться разные константы. Для некоторой величины f , зависящей от n , запись $f = O(n)$ означает, что $|f| \leq Cn$ при всех n .

Рассмотрим задачу отыскания вектора $v = (v_1, v_2, v_3)$, ортогональной матрицы третьего порядка $U = \{u_{ij}\}$ ($UU^T = I$) и перестановки $P = (p_1, \dots, p_n)$ первых n натуральных чисел, для которых величина $E = \frac{1}{n} \sum_{i=1}^n \|UM_{p_i} - N_i - v\|^2$ минимальна. Систему M будем называть подвижной системой, а N – неподвижной системой. Сделаем следующие предположения.

A1. Центры тяжести $\frac{1}{n} \sum_{i=1}^n M_i$ и $\frac{1}{n} \sum_{i=1}^n N_i$ систем M и N совмещены сдвигом в начале координат $OXYZ$.

В предположении **A1** рассматриваемая задача имеет вид

$$E = \frac{1}{n} \sum_{i=1}^n \|UM_{p_i} - N_i\|^2 \rightarrow \min_{P,U} \tag{1}$$

при условиях

$$UU^T = I, P = (p_1, \dots, p_n), p_i \in [1, n], p_i \neq p_j \text{ при } i \neq j. \tag{2}$$

A2. Оси тензоров инерции систем M и N совмещены ортогональным преобразованием с осями декартовой системы $OXYZ$.

Замечание 1. В случае ограничения $p_i = i, i = 1, \dots, n$, задача (1)–(2) решается известным алгоритмом Кабша [1]–[2], результат действия которого обозначим $AK(M, N)$. Положение UM подвижной системы, соответствующее результату действия алгоритма Кабша, обозначим $argmin AK(M, N)$.

Замечание 2. В случае, когда положение подвижной системы в пространстве фиксировано ($U = I$), задача (1)–(2) решается венгерским алгоритмом [3], результат действия которого обозначим $AV(M, N)$. Саму перенумерованную венгерским алгоритмом систему $\{M_{p_i}, i = 1, \dots, n\}$ обозначим $argmin AV(M, N)$.

Перейдем к определению алгоритмов. В соответствии с [8] определим матрицу U в виде

$$U(\varphi, \psi, \theta) = U_i(\varphi, \psi, \theta) = P_i A(\varphi, \psi, \theta), i = 1, 2, \tag{3}$$

где

$$A(\varphi, \psi, \theta) =$$

$$= \begin{pmatrix} \cos\theta + (1 - \cos\theta)x^2 & (1 - \cos\theta)xy - z \sin\theta & (1 - \cos\theta)xz + y \sin\theta \\ (1 - \cos\theta)yx + z \sin\theta & \cos\theta + (1 - \cos\theta)y^2 & (1 - \cos\theta)yz - x \sin\theta \\ (1 - \cos\theta)zx - y \sin\theta & (1 - \cos\theta)zy + x \sin\theta & \cos\theta + (1 - \cos\theta)z^2 \end{pmatrix}, \quad (4)$$

$x = \cos\varphi \cos\psi, y = \sin\varphi \cos\psi, z = \sin\psi, w = (x, y, z)$ – направляющий вектор оси поворота. Здесь $\varphi \in [-\pi/2, \pi/2], \psi \in [-\pi/2, \pi/2]$ – углы, определяющие направляющий вектор w прямой l , проходящей через начало координат, $\theta \in [0, 2\pi)$ – угол поворота точки системы M вокруг прямой l против часовой стрелки, $P_1 = I, P_2$ – матрица отражения [8] относительно прямой l .

Зафиксируем натуральное $m, h = \pi/m$.

Алгоритм 1 (примитивный).

A1.1. $E := AV(M, N)$;

A1.2. Для $i := 1 \dots 2$ выполнить

 Для $j := 0 \dots m$ выполнить

 Для $k := 0 \dots m$ выполнить

 Для $l := 0 \dots m$ выполнить

начало

$$\varphi := -\frac{\pi}{2} + jh, \psi := -\frac{\pi}{2} + kh, \theta := 2lh;$$

$$E1 := AV(U_i(\varphi, \psi, \theta)M, N);$$

 если ($E1 < E$) то ($E := E1$);

конец.

Алгоритм 2 (комбинированный).

A2.1. $E := AK(\operatorname{argmin} AV(M, N), N)$;

A2.2. Для $i := 1 \dots 2$ выполнить

 Для $j := 0 \dots m$ выполнить

 Для $k := 0 \dots m$ выполнить

 Для $l := 0 \dots m$ выполнить

начало

$$\varphi := -\frac{\pi}{2} + jh, \psi := -\frac{\pi}{2} + kh, \theta := 2lh;$$

$$E1 := AK(\operatorname{argmin} AV(U_i(\varphi, \psi, \theta)M, N), N);$$

 если ($E1 < E$) то ($E := E1$);

конец.

Третий алгоритм представляет собой комбинацию алгоритма 2 и итерационного метода покоординатного спуска. Определим сначала алгоритм покоординатного спуска. Обозначим

$$E_1(\varphi, \psi, i) = \min_{\theta \in [0, 2\pi]} AV(U_i(\varphi, \psi, \theta)M, N), \quad E_2(\varphi, \theta, i) = \min_{\psi \in [-\frac{\pi}{2}, \frac{\pi}{2}]} AV(U_i(\varphi, \psi, \theta)M, N),$$

$$E_3(\psi, \theta, i) = \min_{\varphi \in [-\frac{\pi}{2}, \frac{\pi}{2}]} AV(U_i(\varphi, \psi, \theta)M, N).$$

Через $\theta = \operatorname{argmin} E_1(\varphi, \psi, i), \psi = \operatorname{argmin} E_2(\varphi, \psi, i), \varphi = \operatorname{argmin} E_3(\varphi, \psi, i)$ обозначим соответствующие значения аргументов, при которых достигается минимум. Зафиксируем $\epsilon \in (0, 1], \varphi \in [-\pi/2, \pi/2], \psi \in [-\pi/2, \pi/2]$. Рассмотрим следующий алгоритм

начало

$\theta := 0;$
 $F := AV(U_1(\varphi, \psi, \theta)M, N);$
 1: Для $i := 1 \dots 2$ выполнить
начало
 $F_1(i) := E_1(\varphi, \psi, i); \theta_1 := \operatorname{argmin} E_1(\varphi, \psi, i);$
 $F_2(i) := E_2(\varphi, \theta_1, i); \psi_1 := \operatorname{argmin} E_2(\varphi, \theta_1, i);$
 $F_3(i) := E_3(\psi_1, \theta_1, i); \varphi_1 := \operatorname{argmin} E_3(\psi_1, \theta_1, i);$
конец;
 $F_4 := \min_{i \in \{1, 2\}} F_3(i);$
 если $(|F_4 - F| > \epsilon ps)$ то $(\varphi := \varphi_1; \psi := \psi_1; \theta := \theta_1; F := F_4; \text{идти на } 1);$
 $F := F_4;$

конец.

Результат действия данного алгоритма (число F) обозначим $F := DCM(\varphi, \psi, \epsilon ps).$

Алгоритм 3 (итерационный).

A3.1. $E := AV(M, N);$
 A3.2. Для $j := 0 \dots m$ выполнить
 Для $k := 0 \dots m$ выполнить
начало
 $\varphi := -\frac{\pi}{2} + jh, \psi := -\frac{\pi}{2} + kh;$
 $E1 := DCM(\varphi, \psi, \epsilon ps);$
 если $(E1 < E)$ то $(E := E1);$
конец.

3. Оценка скорости сходимости алгоритмов

В данном разделе мы докажем следующее утверждение.

Теорема. Для отыскания значения E с точностью \mathcal{E} алгоритмами 1-3 достаточно совершить $O(n^3 / \mathcal{E}^{3/2})$ арифметических операций.

Доказательство проведем для алгоритма 1. Очевидно, что значение E для алгоритмов 2 и 3 не больше, чем для алгоритма 1, поэтому и для них теорема будет справедлива. Обозначим $AV(U_i(\varphi, \psi, \theta)M, N) = f_i(\varphi, \psi, \theta), i = 1, 2.$ Пусть минимум $E1$ функции $f_i(\varphi, \psi, \theta)$ достигается при некотором паросочетании вершин P и значениях параметров $\varphi = \tilde{\varphi}, \psi = \tilde{\psi}, \theta = \tilde{\theta}, i = \tilde{i}$ Обозначим через $\tilde{f}(\varphi, \psi, \theta)$ функцию, значение которой равно $AV(U_{\tilde{i}}(\varphi, \psi, \theta)M, N)$ для этого фиксированного паросочетания P при всех $\varphi, \psi, \theta.$ Тогда, очевидно, $\min \tilde{f}(\varphi, \psi, \theta) = \min_{i \in \{1, 2\}} f_i(\varphi, \psi, \theta).$ Но в силу своего определения (см. (3)-(4)) \tilde{f} есть дважды дифференцируемая функция с ограниченными вторыми частными производными. Поэтому в точке минимума ее первый дифференциал $d\tilde{f}(\tilde{\varphi}, \tilde{\psi}, \tilde{\theta}) = 0.$ Тогда согласно формуле Тейлора

$$\tilde{f}(\varphi, \psi, \theta) = E + \frac{1}{2} \sum_{i=1}^3 \sum_{j=1}^3 \frac{\partial^2 f}{\partial x^i \partial x^j}(\tilde{\varphi}, \tilde{\psi}, \tilde{\theta})(x_i - \tilde{x}_i)(x_j - \tilde{x}_j), \tag{5}$$

где обозначено $x_1 = \varphi, x_2 = \psi, x_3 = \theta.$ При этом

$$\frac{\partial^2 f}{\partial x_i \partial x_j}(\varphi, \psi, \theta) = \sum_{k=1}^n \frac{\partial^2 f_k}{\partial x_i \partial x_j}(\varphi, \psi, \theta), \tag{6}$$

где $f_k(\varphi, \psi, \theta) = \frac{1}{n} \|U_{\tilde{\tau}}(\varphi, \psi, \theta)M_{p_k} - N_k\|^2$. Поскольку $\left| \frac{\partial^2 \tilde{f}}{\partial x_i \partial x_j} \right| \leq C/n$, то из (6) следует, что

$$\frac{\partial^2 f}{\partial x_i \partial x_j}(\tilde{\varphi}, \tilde{\psi}, \tilde{\theta}) \leq C. \tag{7}$$

Поскольку в $O(1/m)$ -окрестности точки $(\tilde{\varphi}, \tilde{\psi}, \tilde{\theta})$ всегда найдется набор узлов $(\varphi_j, \psi_k, \theta_l)$, по которым идет перебор в алгоритме 1, то из (5)-(7) имеем $\min_{j,k,l} \tilde{f}(\varphi_j, \psi_k, \theta_l) - E \leq \frac{C}{m^2}$. Учитывая, что для любых $(\varphi, \psi, \theta) \min_{i,j,k,l} f_i(\varphi, \psi, \theta) \leq \min_{j,k,l} \tilde{f}(\varphi, \psi, \theta)$, получаем, что

$$\min_{i,j,k,l} f(\varphi_j, \psi_k, \theta_l) - E \leq \frac{C}{m^2}. \tag{8}$$

Поскольку каждое значение $f_i(\varphi, \psi, \theta)$ находится за $O(n^3)$ арифметических операций, а нахождение минимума с точностью ε требует в силу (8) значения $m = O(\varepsilon^{-1/2})$, то общее число операций составит $O(n^3 / \varepsilon^{3/2})$. Теорема доказана.

4. Результаты численных экспериментов.

Рассматривались три семейства S_1, S_2, S_3 атомных решеток, интерпретируемых как совокупности точек в R^3 . Первое семейство S_1 состоит из 19 структур, каждая из которых содержит 9 точек, второе S_2 – из 12 структур, содержащих 18 точек каждая, третье семейство S_3 – из 7 структур, содержащих 57 точек каждая. Вычислительный эксперимент для каждой из структур состоял в последовательном отыскании величины (1) для всевозможных пар структур семейств. Приближенное значение этой величины, найденное одним из алгоритмов 1,2,3, обозначим E_{appr} . В алгоритме 3 минимум на каждой итерации покоординатного спуска определялся методом золотого сечения.

Результаты счета представлены в трех таблицах. В каждой из таблиц для соответствующего семейства представлена зависимость абсолютной погрешности $\Delta E = \max_{S_i} |E_{appr} - E_{exact}|$ для алгоритмов 1,2,3 соответственно, а также время счета на компьютере INTEL(R) Core(TM) i5-2500 CPU @ 3,30GHz 3,60 GHz. При этом для S_1 E_{exact} вычислялось с помощью алгоритма Кабша и полного перебора 9! возможных паросочетаний, а для S_2 и S_3 в качестве E_{exact} бралось значение алгоритма 2 для значения $m = 128$.

Таблица 1. Семейство S_1 .

m	4	8	16	32	64
Алгоритм 1	3.15E-1 2 сек.	1.07E-1 12 сек.	5.59E-2 1 мин. 20 сек.	2.79E-2 9 мин. 37 сек.	6.07E-3 1 ч. 12 мин. 42 сек.
Алгоритм 2	5.04.E-2 5 сек.	1.45E-5 25 сек.	1.45 E-5 2 мин. 54 сек.	1.34E-5 21 мин.30 сек.	1.27E-5 2 ч. 44 мин. 13 сек.
Алгоритм 3	7.18.E-2 1 мин.45 сек.	5.86E-2 5 мин. 20сек.	5.86E-2 18 мин. 37 сек.	5.64E-2 1 ч. 9 мин. 41 сек.	5.64E-2 4 ч. 36 мин. 2 сек.

Таблица 2. Семейство S_2 .

m	4	8	16	32	64
-----	---	---	----	----	----

Алгоритм 1	3.56E-1 2 сек.	2.40E-1 14 сек.	1.86E-1 1 мин. 38 сек.	2.45E-2 11 мин. 45 сек.	1.38E-2 1 ч. 29 мин. 22 сек.
Алгоритм 2	8.26E-2 3 сек.	3.11E-2 21 сек.	3.71E-6 2 мин. 20 сек.	3.38E-6 17 мин. 43 сек.	4.22E-7 2 ч. 14 мин. 49 сек.
Алгоритм 3	8.55E-2 1 мин.10 сек.	8.55E-2 3 мин. 35 сек.	8.55E-2 12 мин. 27 сек.	8.55E-2 46 мин. 4 сек.	8.55E-2 2 ч. 57 мин. 38 сек.

Таблица 3. Семейство S_3 .

m	4	8	16	32	64
Алгоритм 1	7.20E-1 5 сек.	5.02E-1 1 мин. 31.сек.	1.60E-1 3 мин.35 сек.	1.48E-1 26 мин. 40 сек.	1.83E-2 3 ч. 33 мин. 51 сек.
Алгоритм 2	3.80E-1 6 сек.	1.53E-1 35 сек.	8.10E-2 4 мин. 2 сек.	1.74E-5 30 мин. 8 сек.	1.39E-6 3 ч. 46 мин.49 сек.
Алгоритм 3	1.28E-1 2 мин. 28 сек.	1.28E-1 7 мин. 22 сек.	1.15E-1 25 мин. 25 сек.	1.15E-1 1 ч. 24 мин. 40 сек.	1.15E-1 6 ч. 1 мин. 19 сек.

5. Выводы

1. Из результатов экспериментов видно, что алгоритм 2 имеет существенные преимущества перед алгоритмами 1 и 3. При сопоставимом с алгоритмом 1 времени счета, алгоритм 2, начиная с некоторого значения m , фактически, дает точное значение E , достигая на всех парах оптимального паросочетания перед выполнением алгоритма Кабша. Остаточная погрешность связана с погрешностями численных методов алгебры, используемых в алгоритме Кабша.

2. Алгоритм 3 демонстрирует отсутствие сходимости. Это связано с тем, что метод золотого сечения находит минимум только для унимодальных функций, а условие унимодальности не всегда выполняется. Однако применение более надежных методов отыскания одномерного минимума приведет к существенному возрастанию времени счета. Применение метода золотого сечения в итерационных алгоритмах показало хорошие результаты для отыскания максимума объема пересечения выпуклых многогранников в работе [9]. Однако для задач из настоящей статьи он оказался менее эффективен.

6. Литература

- [1] Kabsch, W. A solution of the best rotation to relate two sets of vectors // Acta Crystallographica. – 1976. – Vol. 32. – P. 922-923.
- [2] Kabsch, W. A discussion of the solution for the best rotation to relate two sets of vectors // Acta Crystallographica. – 1978. – Vol. 34. – P. 827-828.
- [3] Kuhn, H.W. The Hungarian Method for the Assignment Problem // Naval Research Logistics. – 2005. – Vol. 52(1). – P. 7-21.
- [4] Sadeghi, A. Metrics for measuring distances in configuration spaces / A. Sadeghi, S. Alireza Ghasemi, B. Schaefer, S. Mohr, M.A. Lill, S. Goedecker // The Journal of chemical physics. – 2013. – Vol. 139. – P. 184118.
- [5] Allen, W.J. Implementation of the Hungarian Algorithm to Account for Ligand Symmetry and Similarity in Structure-Based Design / W.J. Allen, R.C. Rizzo // J. Chem. Inf. Model. – 2014. – Vol. 54. – P. 518-529.
- [6] Griffiths, M. Optimal Alignment of Structures for Finite and Periodic Systems / M. Griffiths, S.P. Niblett, D.J. Wales // J. Chem. Theory Comput. – 2017. – Vol. 13. – P. 4914-4931.

- [7] Temelso, B. ArbAlign: A Tool for Optimal Alignment of Arbitrarily Ordered Isomers Using the Kuhn-Munkres Algorithm / B. Temelso, J.M. Mabey, T. Kubota, N. Appiah-padi, G.C. Shields // J. Chem. Info. Model. – 2017. – Vol. 57(5). – P. 1045-1054.
- [8] Szymanski, J.E. Basic Mathematics for Electronic Engineers: Models and Applications. – Taylor & Francis, 1989. – P. 154.
- [9] Shevchenko, A.H. Local coordination versus overall topology in crystal structures: deriving knowledge from crystallographic databases / A.H. Shevchenko, I.A. Blatov, E.V. Kitaeva, V.A. Blatov // Crystal Growth and Design. – 2017. – Vol.17(2). – P. 774-785.

On the combination of the Hungarian algorithm and the Kabsch algorithm for solving problems of computational geometry

I.A. Blatov¹, E.V. Kitaeva²

¹Povolzhskiy State University of Telecommunications and Informatics, L. Tolstoy street 23, Samara, Russia, 443010

²Samara National Research University, Moskovskoe Shosse 34A, Samara, Russia, 443086

Abstract. The problem of determining the minimum sum of squares of distances between two systems of points in space is considered. The need to solve this problem arises in computer chemistry and in pattern recognition. A rough algorithm has factorial complexity. An algorithm is proposed that allows you to find a solution with a given accuracy \mathcal{E} for $O(n^3/\mathcal{E}^{3/2})$ arithmetic operations