

ЗАДАЧА РАВНОМЕРНОГО РАСПРЕДЕЛЕНИЯ ТРЕБОВАНИЙ ПО УЗЛАМ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ

Е.В. Ляпунцова¹, М.С. Шихалиев²

¹ Московский государственный технический университет им. В.Э. Баумана, г. Москва, Россия,

² Российский университет транспорта (МИИТ), г. Москва, Россия

Задачи организаций вычислений в вычислительных сетях стали весьма популярными за последнее время. Это и составление расписаний решения многих задач на единой вычислительной сети, и различного рода задачи загрузки узлов вычислительной сети. В статье рассматривается задача равномерной загрузки узлов вычислительной сети. В качестве критерия берется максимальная относительная загрузка узлов по объему занимаемой памяти. Задача обобщена на случай двух критериев. Предложены эвристические алгоритмы решения задач, которые выгодно отличаются от используемого ранее алгоритма ветвей и границ с получением оценок на основе метода сетевого программирования, так как при большом числе требований этот метод получения оценок приводит к большому объему вычислений, поскольку получение оценок требует на каждом шаге решения систем линейных неравенств.

Ключевые слова: узлы вычислительной сети, равномерная загрузка, объем занимаемой памяти, эвристические алгоритмы

1. Постановка задачи

Имеются m узлов вычислительной сети и n требований (запросов). Необходимо распределить требования по узлам по критерию равномерной загрузки узлов (по объему занимаемой памяти). Обозначим b_i – объем памяти, занимаемый i -м требованием, $x_{ij} = 1$, если требование i размещено в узле j , $x_{ij} = 0$, в противном случае.

Задача. Определить $\{x_{ij}, i = \overline{1, n}, j = \overline{1, m}\}$, минимизирующие

$$Y(x) = \max_j \sum_i b_i x_{ij} \quad (1)$$

при ограничениях

$$\sum_i x_{ij} = 1, i = \overline{1, n}. \quad (2)$$

Поставленная задача известна как «задача о камнях» [1]: необходимо разделить n камней разного веса на m максимально близких по суммарному весу групп. Для ее решения в [1] предложен алгоритм ветвей и границ с получением оценок на основе метода сетевого программирования.

Однако при больших n этот метод получения оценок приводит к большому объему вычислений, поскольку получение оценок требует на каждом шаге решения систем линейных неравенств. Рассмотрим эвристический алгоритм решения задачи.

Без ограничения общности будем считать все b_i целыми положительными числами.

Описание алгоритма

Предварительный шаг. Пусть x_{ij} могут принимать любые значения на отрезке $[0; 1]$. В этом случае задача легко решается. Определяем

$$M = \left\lceil \frac{1}{m} \sum_i b_i \right\rceil,$$

где $\lceil \cdot \rceil$ ближайшее сверху целое число.

1 этап. Решаем задачу максимизации

$$\sum_i x_{i1} b_i \quad (3)$$

при ограничении

$$\sum_i b_i x_{ij} \leq M, j = \overline{1, m} \text{ для } j = 1. \quad (4)$$

Это частный случай задачи о ранце, для которого при целочисленных значениях параметров существует эффективный метод дихотомического программирования [2]. Обозначим Q_1 множество требований, загруженных в первый узел. Далее решаем задачу (3), (4) для второго узла, исключив из множества требований множество Q_1 , и т. д. После решения m задач для всех узлов остается множество Q_0 требований, не вошедших в решения предыдущих задач. Имеет место теорема.

Теорема. Число требований множества Q_0 не превышает $(m - 1)$.

Доказательство

Обозначим

$$\Delta_j = M - \sum_i x_{ij}^0 b_i, j = \overline{1, m}, \quad (5)$$

где $\{x_{ij}^0\}$ – оптимальное решение j -й задачи. Заметим, что для любого $i \in Q_0$ имеет место

$$b_i > \max_j \Delta_j, i \in Q_0, \quad (6)$$

в противном случае запрос i был бы включен в решение одной из задач. С другой стороны,

$$\sum \Delta_j \geq \frac{1}{m} \sum b_i - \sum_{i,j} x_{ij}^0 b_i = \sum_{i \in Q_0} b_i. \quad (7)$$

Условия (6), (7) могут выполняться, только если число элементов множества Q_0 не превышает $(m - 1)$.

Если $Q_0 = \emptyset$, то задача решена. Теорема доказана.

Если $Q_0 \neq \emptyset$, то распределяем требования $i \in Q_0$ по узлам так, чтобы минимизировать максимальную загрузку. Эта задача легко решается на основе следующего правила.

Пусть узлы упорядочены по убыванию Δ_j , а требования упорядочены по убыванию $b_i, i \in Q_0$,

то есть

$$\Delta_{j1} \geq \Delta_{j2} \geq \dots \geq \Delta_{jm},$$

$$b_{i1} \geq b_{i2} \geq \dots \geq b_{ip},$$

где p – число элементов множества Q_0 . Размещаем требование i_1 на узел j_1 , требование i_2 на узел j_2 и т. д.

2 этап. Определяем

$$M'_2 = \max_k (M - \Delta_{ik} + b_{ik}),$$

где $b_{ik} = 0$ для $k > p$. Берем $M_2 = \lceil \frac{1}{2}(M_1 + M'_2) \rceil$ и повторяем шаги первого этапа. Возможны два варианта.

1 вариант. $Q_0 = \emptyset$.

В этом случае определяем $M_3 = \lceil \frac{1}{2}(M_1 + M_2) \rceil$ и повторяем шаги первого этапа.

2 вариант. $Q_0 \neq \emptyset$. В этом случае определяем $M_3 = \lceil \frac{1}{2}(M_2 + M'_2) \rceil$ и повторяем шаги 1 этапа алгоритма.

За конечное число этапов получаем ситуацию, когда на очередном этапе s имеет место $M_s = M_{s-1}$.

Алгоритм закончен.

Пример 1. Имеются 8 требований и 5 узлов, данные о которых приведены в табл. 1.

Таблица 1

i	1	2	3	4	5	6	7	8
b_i	15	13	11	10	9	8	5	3

Краткие сообщения

1 этап. Вычисляем $M_1 = \left[\frac{\sum b_i}{m} \right] = 15$.

1 шаг. Решаем задачу максимизации

$$15x_1 + 13x_2 + 11x_3 + 10x_4 + 9x_5 + 8x_6 + 5x_7 + 3x_8$$

при ограничении

$$15x_1 + 13x_2 + 11x_3 + 10x_4 + 9x_5 + 8x_6 + 5x_7 + 3x_8 \leq 15.$$

Задачу решаем для $R = 18$, чтобы использовать результаты для следующих этапов. Делаем это методом дихотомического программирования.

Рассматриваем требования 1 и 2. Решение приведено в табл. 2.

Рассматриваем требования 3 и 4 (табл. 3).

Рассматриваем требования 5 и 6 (табл. 4).

Рассматриваем требования 7 и 8 (табл. 5).

Таблица 2
Объединенное
требование I

1	13	–
0	0	15
2 1	0	1

Таблица 3
Объединенное
требование II

1	10	–
0	0	11
4 3	0	1

Таблица 4
Объединенное
требование III

1	8	17
0	0	9
6 5	0	1

Таблица 5
Объединенное
требование IV

1	3	8
0	0	5
8 7	0	1

Рассматриваем объединенные требования I и II (табл. 6).

Рассматриваем объединенные проекты III и IV (табл. 7).

Таблица 6
Объединенное требование V

2	11	–	–
1	10	–	–
0	0	13	15
II I	0	1	2

Таблица 7
Объединенный проект VI

3	8	16	17	–
2	5	13	14	–
1	0	11	12	–
0	0	8	9	17
IV III	0	1	2	3

Рассматриваем объединенные проекты V и VI (табл. 8).

Объединенный проект VII

Таблица 8

4	15	18	–	–	–	–	–	–	–	–	–	–
3	13	16	–	–	–	–	–	–	–	–	–	–
2	11	14	16	–	–	–	–	–	–	–	–	–
1	10	13	15	–	–	–	–	–	–	–	–	–
0	0	3	5	8	9	11	12	13	14	16	17	–
V VI	0	1	2	3	4	5	6	7	8	–	–	–

1 шаг. Определяем решение для $R = 15$.

Имеются два решения для 1-го узла. Берем одно из них. Это требование 1.

2 шаг. Определяем решение для 2-го узла. Единственное решение без учета требования [1] это требование (4, 7).

3 шаг. Определяем решение для 3-го узла. Имеется одно решение. Это требование (3, 8).

4 шаг. Определяем решение для 4-го узла. Это требование 2.

5 шаг. Определяем решение для 5-го узла. Это требование 5. Множество $Q_0 = (6)$. Загружаем требование 6 в узел 5. Получаем максимальную загрузку $M'_2 = 17$.

2 этап. Вычисляем $M_2 = \lceil \frac{1}{2}(M_1 + M'_2) \rceil = 16$.

Решаем задачу при $R = 16$. Определяем решение для 1-го узла.

1 шаг. Имеются три решения: (2, 8), (3, 7) и (6, 7, 8).

Выбираем, например, требование (2, 8).

2 шаг. Определяем решение для 2-го узла. Берем требование (3, 7).

3 шаг. Определяем решение для третьего узла. Берем требование 1.

4 шаг. Определяем решение для четвертого узла. Берем требование 4.

5 шаг. Определяем решение для пятого узла. Берем требование 5. Множество $Q_0 = (6)$. Загружаем требование 6 в узел 5. Получаем максимальную загрузку $M'_3 = 17$. Имеем $M_3 = \lceil \frac{1}{2}(M'_3 + M_2) \rceil = 17$. Алгоритм закончен. Нетрудно показать, что в данном случае получено оптимальное решение.

2. Двухкритериальная задача

Рассмотрим задачу с двумя критериями:

$$F_1 = \max_j z_j = \max_j \sum_i x_{ij} c_i; \quad (8)$$

$$F_2 = \max_j u_j = \max_j \sum_i x_{ij} a_i.$$

В качестве интегрального критерия возьмем

$$F = \max(qF_1; F_2) \rightarrow \min,$$

где q – коэффициент значимости критерия F_1 по сравнению с F_2 .

Положим $b_i = qc_i$, $i = \overline{1, n}$. В этом случае интегральный критерий принимает вид

$$F = \max(F_1; F_2), \text{ где } F_1 = \max_j \sum_i b_i x_{ij}. \quad (9)$$

3. Описание алгоритма

1 этап. Вычисляем

$$M_1 = \lceil \frac{1}{m} \max(\sum_i b_i; \sum_i a_i) \rceil.$$

Решаем задачу о ранце для первого узла: определить $x_i = (0, 1)$, $i = \overline{1, n}$, максимизирующие

$$\sum_i b_i x_i \quad (10)$$

при ограничениях

$$\sum_i b_i x_i \leq M_1; \quad (11)$$

$$\sum_i a_i x_i \leq M_1. \quad (12)$$

Обозначим Q_1 множество требований, загруженных в первый узел.

Решаем задачу (10), (11), (12) для второго узла, исключив множество требований Q_1 , и т. д.

После решения m задач возможны два варианта.

1. Множество нераспределенных требований $Q_0 = \emptyset$. В этом случае алгоритм закончен.

2. Множество $Q_0 \neq \emptyset$. В этом случае загружаем требования множества Q_0 следующим образом.

Краткие сообщения

Рассматриваем их в любом порядке. Для рассматриваемого требования i определяем узел q , такой что величина

$$Y_q = \max_j \left[\sum_{i \in Q_j} a_i + a_q \right]$$

принимает минимальное значение. Загружаем требование i в узел q . Обозначим M'_2 – значение $M(x)$ в решении задачи.

2 этап. Вычисляем

$$M_2 = \left[\frac{1}{2}(M_1 + M'_2) \right]$$

и повторяем алгоритм 1 этапа.

3 этап. Обозначим M'_3 – значение $M(x)$ в решении задачи на втором этапе.

Если $M'_3 = M_2$, то задача решена.

Если $M'_3 < M_2$, то определяем $M_3 = \left[\frac{1}{2}(M_2 + M'_3) \right]$ и повторяем алгоритм этапа 1.

Если $M'_3 > M_2$, то определяем $M_3 = \left[\frac{1}{2}(M'_2 + M_2) \right]$ и повторяем алгоритм этапа 1.

За конечное число этапов s получим ситуацию, когда $M'_s = M_{s-1}$. Алгоритм окончен.

Пример 2. Имеются 8 требований и 5 узлов, данные о которых приведены в табл. 9

Таблица 9

i	1	2	3	4	5	6	7	8
a_i	18	16	15	13	10	9	6	3
b_i	5	6	7	10	14	15	16	17

1 этап. Вычисляем

$$M_1 = \left[\frac{1}{5} \max(90; 90) \right] = 18.$$

1 шаг. Решаем следующую задачу о ранце:

максимизировать

$$18x_1 + 16x_2 + 15x_3 + 13x_4 + 10x_5 + 9x_6 + 6x_7 + 3x_8$$

при ограничениях

$$18x_1 + 16x_2 + 15x_3 + 13x_4 + 10x_5 + 9x_6 + 6x_7 + 3x_8 \leq 18$$

$$5x_1 + 6x_2 + 7x_3 + 10x_4 + 14x_5 + 15x_6 + 16x_7 + 17x_8 \leq 18.$$

Одно из решений $x_1 = 1$. Загружаем требование 1 в узел 1.

2 шаг. Решаем задачу о ранце без требования 1. Решение $x_2 = 1$. Загружаем требование 2 в узел 2.

3 шаг. Решаем задачу без требований 1 и 2. Решение $x_3 = 1$. Загружаем требование 3 в узел 3.

4 шаг. Решаем задачу без требований 1, 2, 3. Решение $x_4 = 1$. Загружаем требование 4 в узел 4.

5 шаг. Решаем задачу без требований 1, 2, 3, 4. Решение $x_5 = 1$. Загружаем требование 5 в узел 5.

Имеем $Q_0 = (6, 7, 8)$. Берем требование 6. Вычисляем:

$$Y_1 = \max(18 + 9; 5 + 15) = 27,$$

$$Y_2 = \max(16 + 9; 6 + 15) = 25,$$

$$Y_3 = \max(15 + 9; 7 + 15) = 24,$$

$$Y_4 = \max(13 + 9; 10 + 15) = 25,$$

$$Y_5 = \max(10 + 9; 14 + 15) = 29.$$

Загружаем требование 6 в узел 3.

Берем требование 7. Вычисляем

$$Y_1 = \max(18 + 6; 5 + 16) = 24,$$

$$Y_2 = \max(16 + 6; 6 + 16) = 22,$$

$$Y_3 = \max(24 + 6; 22 + 16) = 38,$$

$$Y_4 = \max(13 + 6; 10 + 16) = 26,$$

$$Y_5 = \max(10 + 6; 14 + 16) = 30.$$

Загружаем требование 7 в узел 2.

Берем требование 8. Вычисляем

$$Y_1 = \max(18 + 3; 5 + 17) = 22$$

$$Y_2 = \max(22 + 3; 22 + 17) = 39$$

$$Y_3 = \max(24 + 3; 22 + 17) = 39$$

$$Y_4 = \max(13 + 3; 10 + 17) = 27$$

$$Y_5 = \max(10 + 3; 14 + 17) = 31$$

Загружаем требование 8 в узел 1.

Имеем

$$M'_2 = \max(22; 22; 24; 13; 14) = 24.$$

2 этап. Берем

$$M_2 = \frac{1}{2}(18 + 24) = 21.$$

1 шаг. Повторяем алгоритм этапа 1 при $M_2 = 21$. Нетрудно проверить, что то же самое решение с $M'_3 = 24$.

3 этап. Берем

$$M_3 = \left[\frac{1}{2}(21 + 24) \right] = 23.$$

Повторяем алгоритм этапа 1 при $M_3 = 23$.

1 шаг. Решение $x_2 = 1, x_7 = 1$. Загружаем требования 2 и 7 в узел 1.

2 шаг. Решение $x_1 = 1, x_8 = 1$. Загружаем требования 1 и 8 в узел 2.

3 шаг. Решение $x_3 = 1$. Загружаем требование 3 в узел 3.

4 шаг. Решение $x_4 = 1$. Загружаем требование 4 в узел 4.

5 шаг. Решение $x_5 = 1$. Загружаем требование 5 в узел 5. Имеем $Q_0 = 6$.

Вычисляем

$$Y_1 = \max(22 + 9; 22 + 15) = 37,$$

$$Y_2 = \max(21 + 9; 22 + 15) = 37,$$

$$Y_3 = \max(15 + 9; 7 + 15) = 24,$$

$$Y_4 = \max(13 + 9; 10 + 15) = 25,$$

$$Y_5 = \max(10 + 9; 14 + 15) = 29.$$

Загружаем требование 6 в узел 3.

Имеем $M'_4 = 24$. Поскольку $M_4 = \left[\frac{1}{2}(M_3 + M'_4) \right] = 24 = M'_4$, то алгоритм закончен.

Заключение

Рассмотренные алгоритмы требуют решения задач о ранце. Существует эффективный приближенный алгоритм решения задач о ранце, когда предметы (требования) упорядочиваются по убыванию эффективностей $q_i = a_i/b_i$ и отбираются в этой очередности. Этот алгоритм дает небольшую погрешность при большой величине отношения n/m [3].

В частности, в задаче с одним критерием это означает, что требования могут отбираться в любом порядке. Поскольку эффективности равны 1. Результаты экспериментального исследования предложенных алгоритмов будут представлены в следующих публикациях.

Литература

1. Буркова, И.В. Метод сетевого программирования в задачах управления проектами: дис. ... д-ра техн. наук / И.В. Буркова. – М.: ИПУ РАН, 2012.
2. Буркова, И.В. Метод сетевого программирования в задачах нелинейной оптимизации / И.В. Буркова // Автоматика и телемеханика. – 2009. – № 10. – С. 15–21.
3. Сигал, И.Х. Введение в прикладное дискретное программирование: модели и вычислительные алгоритмы: учеб. пособие / И.Х. Сигал, А.П. Иванова. – 2-е изд., испр. и доп. – М.: Физматлит, 2007. – 304 с.

Ляпунцова Елена Вячеславовна, д-р техн. наук, профессор, кафедра ИБМ 7 «Техническое предпринимательство», Московский государственный технический университет им. В.Э. Баумана, г. Москва; lev77@me.com.

Шихалиев Марат Сираджединович, аспирант, Российский университет транспорта (МИИТ), г. Москва; maratsh@live.ru.

Поступила в редакцию 6 декабря 2017 г.

DOI: 10.14529/ctcr180118

THE PROBLEM OF UNIFORM DISTRIBUTION REQUIREMENTS FOR NODES OF THE COMPUTER NETWORK

E.V. Lyapunтова¹, lev77@me.com,
M.S. Shikhaliyev², maratsh@live.ru

¹ Bauman Moscow State Technical University, Moscow, Russian Federation,

² Russian University of Transport, Moscow, Russian Federation

The tasks of computing organizations in computer networks have become very popular lately. This is the scheduling of solving many problems on a single computer network and various tasks of loading nodes of the computer network. The article deals with the problem of uniform loading of nodes of the computer network. As a criterion, the maximum relative loading of nodes by the amount of memory is taken. The problem is generalized to the case of two criteria. Heuristic algorithms for solving problems are offered, which favorably differ from the algorithm of branches and boundaries used earlier to obtain estimates based on the method of network programming. With a large number of requirements, this method of obtaining estimates leads to a large amount of computation, since obtaining estimates requires, at each step, solutions of systems of linear inequalities.

Keywords: knots of the computer network, uniform loading, volume of the occupied memory, heuristic algorithms.

References

1. Burkova I.V. *Metod setevogo programmirovaniya v zadachakh upravleniya proektami: dis. dokt. tekhn. nauk* [Method of Network Programming in Project Management Tasks. Doct. diss.]. Moscow, IPC RAS, 2012.
2. Burkova I.V. [Method of Network Programming in Tasks of Non-Linear Optimization]. *Automatic Equipment and Telemechanics*, 2009, no. 10, pp. 15–21. (in Russ.)
3. Sigal I.Kh., Ivanova A.P. *Vvedeniye v prikladnoye diskretnoye programmirovaniye: modeli i vychislitelnyye algoritmy: ucheb. posobiye* [Introduction to Applied Discrete Programming: Models and Computing Algorithms: Manual]. Moscow, Fizmatlit Publ., 2007. 304 p.

Received 6 December 2017

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Ляпунцова, Е.В. Задача равномерного распределения требований по узлам вычислительной сети / Е.В. Ляпунцова, М.С. Шихалиев // Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника». – 2018. – Т. 18, № 1. – С. 148–154. DOI: 10.14529/ctcr180118

FOR CITATION

Lyapunтова E.V., Shikhaliyev M.S. The Problem of Uniform Distribution Requirements for Nodes of the Computer Network. *Bulletin of the South Ural State University. Ser. Computer Technologies, Automatic Control, Radio Electronics*, 2018, vol. 18, no. 1, pp. 148–154. (in Russ.) DOI: 10.14529/ctcr180118