федеральное государственное бюджетное образовательное
учреждение высшего образования
«Национальный исследовательский университет «МЭИ»

Институт информационных и вычислительных технологий
Кафедра управления и интеллектуальных технологий

Отчёт по лабораторной работе №1
По дисциплине «Управление в больших системах»
«Синтез больших систем управления. Распределение задач по узлам управления»

Выполнил студент: Михайловский М. Ю.
Группа: А-03-21
Вариант: 5
Проверили: Новиков В. Н, Обычайко Д. С.

Москва 2024

Содержание

1 Постановка задачи

Имеется I задач, которые должны быть решены последовательно друг за другом. Для их решения имеется J узлов. Затраты и время решения i-ой задачи на j-ом узле заданы соответственно матрицами стоимости затрат $C = [c_{ij}]$ и временных затрат $T = [t_{ij}]$.

$$C = \begin{bmatrix} 4{,}5 & 7 & 2 & 2 \\ 5 & 8 & 1 & 3 \\ 5{,}5 & 9 & 6 & 2 \\ 6 & 10 & 7 & 1 \\ 6{,}5 & 7 & 3 & 1 \end{bmatrix}, \; T = \begin{bmatrix} 4{,}5 & 3 & 2 & 9 \\ 5 & 6 & 5 & 10 \\ 5{,}5 & 7 & 6 & 11 \\ 6 & 8 & 7 & 12 \\ 6{,}5 & 9 & 8 & 5 \end{bmatrix} \tag{1}$$

Требуется минимизировать суммарную стоимость затрат при заданном ограничении на суммарные временные затраты $T_з = 25$. Для записи оптимизационной задачи используем x_{ij}:

$$x_{ij} = \begin{cases} 1, & i\text{-ая задача решается на } j\text{-ом узле} \\ 0, & \text{иначе} \end{cases}.$$

Тогда задача оптимизации примет вид:

$$\begin{cases} \sum\limits_{i=1}^{I} \sum\limits_{j=1}^{J} c_{ij} x_{ij} \to \min \\ \sum\limits_{i=1}^{I} \sum\limits_{j=1}^{J} t_{ij} x_{ij} \le T_з \\ \sum\limits_{j=1}^{J} x_{ij} = 1, \forall i : i = \overline{1, I} \end{cases} \tag{2}$$

Для проведения оптимизации будет написана программа с пользовательским интерфейсом для ввода исходных данных.

2 Алгоритмы решения задачи

Решать задачу мы будем в несколько этапов. Сначала над матрицами затрат будут проведены две процедуры оптимизации, которые исключат заведомо неоптимальные узлы для использования.

2.1 Первая оптимизация

Оптимизация проводится построчно, пусть алгоритм находится на i-ой строке. Фиксируется номер узла s, на котором стоимость затрат наименьшая:

$$s = \arg\min_j c_{ij}$$

Затем, исключаются те элементы строки, которые удовлетворяют условию (3). При чём ровно одно из этих неравенств может выполняться нестрого.

$$\begin{aligned} c_{ij} &> c_{is} \\ t_{ij} &> t_{is} \end{aligned} \tag{3}$$

Так мы исключаем узлы, которые не лучше зафиксированного s-ого узла ни по стоимости, ни по времени затрат. В результате такой процедуры данные матрицы (1) преобразуются к следующему виду:

$$C^{(0)} = \begin{bmatrix} - & - & 2 & - \\ - & - & 1 & - \\ 5{,}5 & 9 & 6 & 2 \\ 6 & 10 & 7 & 1 \\ - & - & - & 1 \end{bmatrix},\ T^{(0)} = \begin{bmatrix} - & - & 2 & - \\ - & - & 5 & - \\ 5{,}5 & 7 & 6 & 11 \\ 6 & 8 & 7 & 12 \\ - & - & - & 5 \end{bmatrix} \tag{4}$$

2.2 Вторая оптимизация

Вторая оптимизация так же как и первая проводится построчно. Будем рассматривать её для фиксированной i-ой строки. Эта оптимизация исключает те узлы, использование которых напрямую приводит к нарушению временного ограничения $T_з$.

Для этого используется вектор наименьших временных затрат на узлах системы:

$$T_{\min} = \begin{pmatrix} \min\limits_j t_{1j} & \min\limits_j t_{2j} & \dots & \min\limits_j t_{Ij} \end{pmatrix}^{\mathsf{T}}$$

То есть теоретически наименьшее время решения всех I задач будет равно:

$$T_{\text{теор.мин}} = \|T_{\min}\|_1 = \sum_{i=1}^{I} T_{\min\ i}$$

Тогда для данной i-ой строки j-ый элемент исключается если выполняется условие (5). Это означает, что при решении i-ой задачи на j-ом узле, в любом случае будет нарушено временное ограничение $T_з$.

$$T_{\text{теор.мин}} - T_{\min\ i} + t_{ij} > T_з \tag{5}$$

В результате такой оптимизации матрицы (4) приобретают следующий вид:

$$C^{(1)} = \begin{bmatrix} - & - & 2 & - \\ - & - & 1 & - \\ 5{,}5 & 9 & 6 & - \\ 6 & - & 7 & - \\ - & - & - & 1 \end{bmatrix},\ T^{(1)} = \begin{bmatrix} - & - & 2 & - \\ - & - & 5 & - \\ 5{,}5 & 7 & 6 & - \\ 6 & - & 7 & - \\ - & - & - & 5 \end{bmatrix} \tag{6}$$

В результате в каждой i-ой строке остаётся некоторое допустимое множество узлов для использования $J_{\text{доп}\,i} \subseteq J$.

2.3 Оптимизация методом ветвления

В самом начале фиксируются два различных решения задачи, доставляющие минимумы суммарным стоимостям затрат $C_{\min}$ и временным затратам $T_{\min}$.

$$C_{\min} = \begin{pmatrix} \min\limits_j c_{1j} & \min\limits_j c_{2j} & \dots & \min\limits_j c_{Ij} \end{pmatrix}^{\mathsf{T}}$$
$$T_{\min} = \begin{pmatrix} \min\limits_j t_{1j} & \min\limits_j t_{2j} & \dots & \min\limits_j t_{Ij} \end{pmatrix}^{\mathsf{T}}$$

Для них рассчитываем теоретические минимумы. Этими двумя метриками и будем определять показатели каждой вершины дерева в методе ветвления.

$$C_{\text{теор мин}} = \|C_{\min}\|_1 = 15{,}5,\ T_{\text{теор мин}} = \|T_{\min}\|_1 = 23{,}5.$$

Соотнесём им корень дерева v^0. Пусть корень дерева будет на нулевом уровне $l = 0$. Тогда ему будут смежны вершины v_j^1, $j \in J_{\text{доп}\,i}$, которые будут на первом уровне $l = 1$.

И введём вектор номеров узлов соответствующих вершинам, которые составляют путь до вершины $v^l_{j_l}$:

$$J(v^l_{j_l}) = (j_1\ j_2\ \ldots\ j_l)^\intercal\ ,\ l \le J$$

Тогда на каждой строке матрицы i или уровне дерева l ($i = l$) будем проводить описанную далее процедуру.

1. Рассчитываем для вершин v^l_j, $j \in J_{\text{доп }1}$ метрики $\left\|C_{\text{метр j}}\right\|_1$, $\left\|T_{\text{метр }j}\right\|_1$. Здесь $C_{\text{метр j}}$ вектор, где первые l элементов берутся $J(v^l_j)$, все последующие элементы берутся $C_{\min}$. Метрика временных затрат $T_{\text{метр }j}$ определяется аналогично.

2. Сравниваем эти вершины по их метрикам. Исключаем из рассмотрения те вершины, для которых метрика временных затрат $T_{\text{метр }j} > T_{\text{з}}$. Из оставшихся вершин выбираем вершину с наименьшей метрикой затрат стоимости $C_{\text{метр j}}$.

3. Вершины следующего уровня v^{l+1}_j, $j \in J_{\text{доп }1}$ делаем смежными выбранной вершине v^l_j.

В результате полученное дерево будет иметь вид представленный на рис. 2.1. В результате данного алгоритма на каждом уровне выбирается единственная вершина.

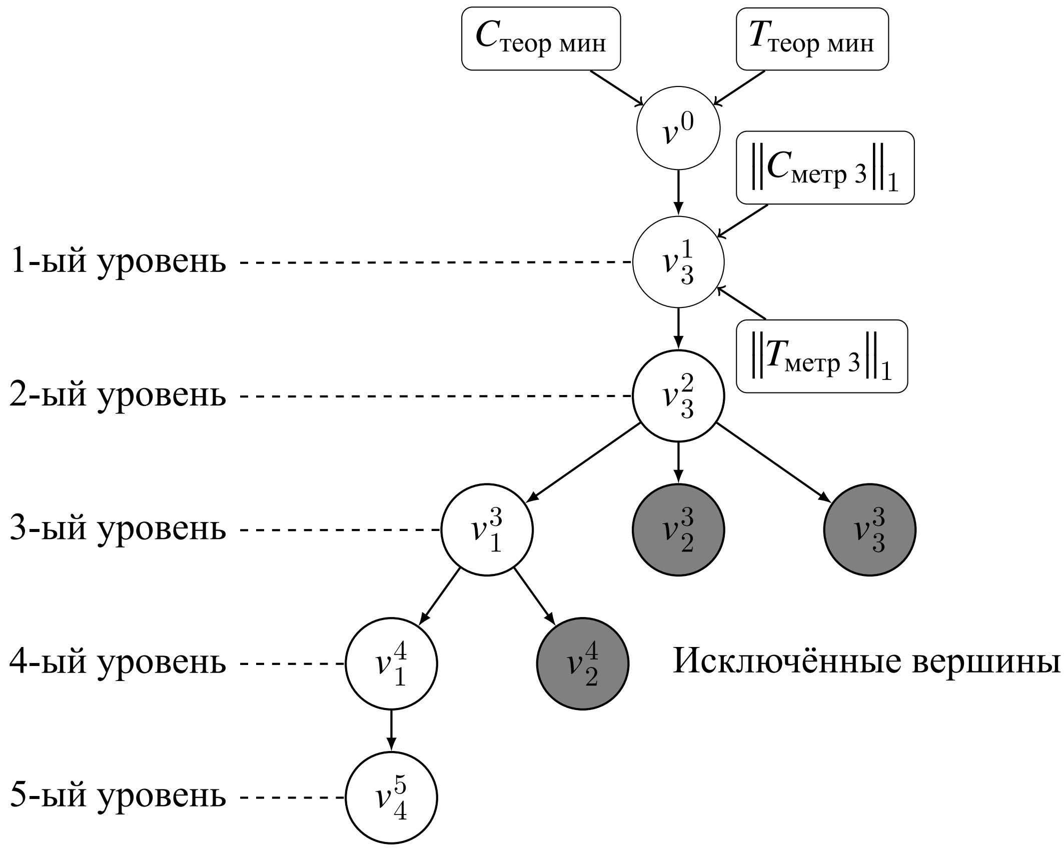

Рис. 2.1. Пример дерева ветвления

3 Реализация программы

Программа написана на языке python 3.11.3. Использованы следующие библиотеки: *numpy=2.1.1, PyQt6=6.7.1, PyQt6-Qt6=6.7.3, PyQt6_sip=13.8.0*.