Алгоритм выбора оптимального решения векторной задачи теории расписания

Кутхашвили К.В.

Факультет математики и информационных технологий

Университет Грузии

г. Тбилиси, республика Грузия

e-mail: kkutkhashvili@yahoo.com

***Аннотация* — На основании методов общей теории расписаний построена математическая модель для таких задач теории расписаний, для которых выполнение заданий возможно одноступенчатой многопроцессорной системой, в которой процессоры наполовину взаимозаменяемы, а дополнительные ресурсы и множество частичного порядка пустые. Кроме того, построен алгоритм для нахождения парето-оптимального решения.**

Ключевые слова: расписание; модель; алгоритм; многокритериальная

Среди задач дискретной оптимизации значительное место занимают задачи прикладного характера. Идеи дискретности широко используются в технической кибернетике для планирования целого ряда производств народного хозяйства. Многие задачи управления или планирования фиксированного ресурса при условии конечного множества заданий требуют их упорядочения во времени. Целью работы является исследование одной конкретной задачи теории расписания при наличии двух критериев и построение алгоритма для нахождения оптимального решения многокритериальной задачи.

Рассмотрена задача теории расписания: Система заданий задана в виде << X, "<", $[\tau_{ij}]i$=1,...,m, j=1,...,n, $\{\Re_j\}j$=1,...,n, $\{\omega_j\}j$=1,...,n >>.

В систему входят X=$\{\xi_1,...,\xi_n\}$ заданий, которые должны быть непосредственно выполнены с помощью P=$\{P_1,...,P_m\}$ процессоров. Процессоры идентичны по быстродействию, но по функциональным возможностям различны. Подразумевается, что m<n. При этом , "<" – нерефлексивное множество частичного порядка, определённое на множестве X, пустое. Матрица $[\tau_{ij}]i$=1,...,m; j=1,...,n представляет собой вектор. Его компоненты τ_j (j=1,...,n) соответствуют той длительности времени, которое необходимо для выполнения j-ого задания на произвольном процесоре; $\Re_j$=(R1(ξ_j),...,Rs(ξ_j)), j=1,...,n, множество, j-ый компонент которого показывает количество ресурсов типа R_j, которое необходимо для выполнения ξ_j-ого задания, пусто. $\{\omega_j\}j$=1,...,n – вектор весов функций, j-ый компонент которого выражает цену существования j-ого задания в системе, а компоненты – величины постоянные.

Наша цель сотавить непрерывное расписание, то есть построить такое непрерывное отображение S (в этом случае образ представляет собой график последовательного выполнения заданий, который задан в виде таблицы), которое каждому заданию сопоставляет последовательность одного или нескольких взаимопересекающихся временных интервалов, расположенных на промежутке [0,∞). Построенное расписание должно удовлетворять следующим условиям: в течение каждого интервала назначается одно устройство; невозможно выполнение одного задания одновременно на двух или нескольких процессорах; сумма длин временных интервалов совпадает с длительностью выполнения задания; временные интервалы разных заданий, назначенных на одно устройство, не должны пересекать друг друга. Должны быть предусмотрены ограничения на последовательность выполнения заданий и на использование дополнительных видов ресурсов. Если через f_j-ое обозначим момент окончания задания ξ_j , тогда на промежутке [0, max f_j] не может существовать такой интервал, в течение которого не одно устройство не было бы сопоставлено с каким-либо заданием.

Для оптимальности построенного графика нужно, чтобы он удовлетворял одному из следующих критериев:

$$\min_S \rho(S) = \min_S[\max_{i\le i\le n}\{f_i(S)\}]$$

$$\rho(S^*) = \min_S \rho(S) = \min_S \frac{1}{n}\sum_{i=1}^{n}\omega_i f_i(S)$$

Указанные выше числовые функции образуют векторный критерий.

Методом ветвей и границ находим оптимальные решения для каждого критерия следующим образом: для каждого k -ого ответвления составляем $t1_{\xi_i}^{(k)}$ вектор и для каждого ξ_i выбираем

$$t1_{\xi_i}^{(k)} = \min_{j\in Q_i} \overline{t_{p_j}}, \quad i=1,\underline{2,...,}n,$$

$$t_{\xi_i}^{(k)} = \max(d_i,\ \min_{j\in Q_i} \overline{t_{p_j}}) \quad i=1,2,...,n.$$

Для того чтобы перейти на следующий уровень, следует выбрать то задание, которое готово к выполнению. Для этого выделим из множества X подмножество заданий $X_1^{(k)} \subset X$ следующим образом: подберем то ξ_i, для которого значение выражения $(t_{\xi_i}^{(k)} + \tau_i)$ минимально. Пусть таким ξ_i будет $\xi_{(k)}^*$, номер его соотвествующего процессора - $j_{(k)}^*$, а минимальным значением будет η , тогда $\eta = \min_{\xi_i\in X_1^k}(t_{\xi_i}^{(k)} + \tau_i)$. Задание $\xi_{(k)}^*$ и все те задания ξ_i , для которых выполняются условия $t_{\xi_i}^{(k)} < \eta$, $j^{*(k)} \in Q_i$, $t1_{\xi_i}^{(k)} - t_{\xi_i}^{(k)} \ge 0$, составляют множество $X_1^{(k)}$.

На каждом шаге обратного хода находятся наибольшие общие подмножества множества $X_1^{(k)}$ для всех критериев, и из найденных подмножеств выбирается парето-оптимальное решение вышеописанным алгоритмом.

[1] Теория расписаний и вычислительные машины (под редакцией Э.Г.Коффмана). - М.: Наука. - 1984.А. Левитин «Алгоритмы. Введение в разработку и анализ. Из. "Вильямс", 2006г.

[2] И.П. Норенков "Комбинированные и генетические алгоритмы составления расписаний в задачах проектирования", вестник МГТУ, сер. "Приборостроение", №2, 1995.

[3] [4] Танаев В.С., Ковалев М.Я., Шафранский Я.М. Теория расписаний. Групповые технологии. Минск, 1998.

[4] [5] Кравцов М.М., Полиэдральные аспекты транспортных задач и сложность многокритериальных задач дискретной оптимизации: Автореф. дис. на соиск. учен. степ. д.ф.-м.н.: (05.13.16) / Ин-т техн. кибернетики.-Минск,1994.- 29с.