Физико-математические науки

УДК 004.02.942

Эффективный алгоритм построения множества оптимальных расписаний

Елена Викторовна Скакалина

Полтавский национальный технический университет, Украина
36011, г. Полтава, пр.Первомайский, 24
Кандидат технических наук, доцент
E-mail: elena.skakalina@bk.ru

Аннотация. В статье рассмотрен алгоритм решения задачи построения оптимальных расписаний. Рассмотрены модели последовательного прохождения работ и параллельного их выполнения. Предлагается эффективный алгоритм построения всех расписаний с минимальным суммарным временем завершения n работ, выполняемых на m неидентичных машинах.

Ключевые слова: задача оптимизации; построение расписаний; оптимальное распределение ресурсов; модели и методы оптимального распределения ресурсов; ERP-системы; логистика.

Введение. Несмотря на то, что задачи составления расписаний достаточно глубоко рассматривались учеными стран СНГ и зарубежными учеными, и результаты их исследований достаточно полно изложены в литературе, общего решения таких задач описано не было. Отсутствие комплексного подхода связано с тем, что эффективность составления расписания зависит от большого количества факторов, а известные методы предлагают решение лишь частных задач.

Материалы и методы. Задачи оптимального распределения ресурсов является обобщением экстремальных задач упорядочения. Проблема упорядочения по сути заключается в нахождении такой очередности выполнения операций на каждом объекте технологической или производственной систем, которая исключает его неоправданные простои.

Обсуждение проблемы. Исходя из проведенного анализа многообразия задач расписания и методов их решения, общая оценка сегодняшнего состояния теории расписаний сводится к следующему [1]:

- в настоящее время изучен широкий круг задач расписания, начиная с простейшей задачи выбора очередности выполнения одноэтапных работ одним исполнителем и кончая так называемой общей задачей, связанной с анализом многошаговых технологических процессов в системах конвейерного типа;
- несмотря на простоту постановок, лишь немногие задачи решены точно;
- разнообразие ограничений, встречающихся в конкретных ситуациях, приводит к неизбежной идеализации исследуемых систем, вследствие чего возникают трудности получения и практического использования научных результатов;
- допустимы различные критерии оценки качества расписаний, но почти все они относятся к затратам времени на производство работ, что вполне согласуется со смыслом и структурой большинства практических задач;
- существуют разные формы представления расписаний, и довольно широк выбор терминов, определяющих одни и те же понятия;
- нет единой методологии составления расписаний, и к основным группам методов, применяемых при поиске решений, относятся методы алгебры, комбинаторного анализа, математического программирования, статистических испытаний и т.п.

Однако анализ теоретических моделей расписаний из теории расписаний показал невозможность описать вышеприведенные расписания классическими средствами. Кроме того, классические отношения предшествования не отражают сложности внутренних связей между операциями в рамках одной работы[2].

Рассмотрим систему из m параллельных неидентичных машин, предназначенных для выполнения n одноэтапных работ, n > m. Пусть $[\beta_{ij}]_{n\times m}$ обозначает матрицу, в которой есть β_{ij} время выполнения работы j на машине i, $i = \overline{1,m}, j = \overline{1,n}$.Определим расписание выполнения n работ на m машинах, как последовательность $\Pi = (\pi_1, \pi_2, \dots \pi_i, \dots \pi_m)$,в которой перестановка π_iзадаёт порядок работ, назначенных на машину i. Пусть в расписании П работа j, выполняемая машиной i, завершается в момент времени $f_{ij}(\Pi)$. Для расписания П найдём суммарное время выполнения n работ на m параллельных машинах, равное $mwft(\Pi) = \sum_{j=I}^{n}\sum_{i=I}^{m} f_{ij}(\Pi)$. Требуется построить расписание П* с минимальным суммарным временем завершения работ. В [3] показано, что искомое расписание П* находится в результате выполнения процедуры определения потока минимальной стоимости в транспортной сети N, которая строится для матрицы с m строками и n столбцами,

$$C = \begin{bmatrix} [\beta_{ij}] \\ [2\beta_{ij}] \\ \dots\dots \\ [n\beta_{ij}] \end{bmatrix} (1)$$

где $[r\beta_{ij}]$ обозначает матрицу, образованную из $[\beta_{ij}]_{m\times n}$ умножением каждого её элемента на r.Рассмотрим, как, применяя алгоритм решения задачи построения всех расписаний с минимальным суммарным временем завершения работ, выполняемых на неидентичных машинах, можно построить за полиномиальное время все расписания П* с минимальным суммарным временем завершения работ. Определим матрицуи обозначаем её элементы β_{ij}^0 , $i = \overline{1,n}, j = \overline{1,m'}, m' = nm$.

$$B = CT = \left[[\beta_{ij}]^T, [2\beta_{ij}]^T, \dots, [r\beta_{ij}]^T, \dots, [n\beta_{ij}]^T\right] (2)$$

Пусть матрица B является входом алгоритма решения задачи, а перестановка $\pi^* = (\beta_{\pi^*(1)}, \beta_{\pi^*(2)}, \beta_{\pi^*(m')})$есть одно из её оптимальных решений. Тогда значение $\beta_{\pi^*(1)} = \beta_{pq}^0$, $p \leq n$, в задаче построения расписания П* вносит вклад mwft(П*) при условии, что работа p выполняется последней на соответствующей машине. Значение $\beta_{\pi^*(k)} = \beta_{pq}^0 = 2\beta_{rs}$, $r \leq n$, представляет собой слагаемое в mwft(П*) в предложении, что работа r завершается предпоследней на соответствующей машине, и так далее. Таким образом, решению π^* соответствуют решение П*, и алгоритм решения задачи корректно находит все решения задачи минимизации суммарного времени выполнения работ.

В общем случае допустимое расписание $\Pi = (\pi_1, \pi_2, \dots \pi_i, \dots \pi_m)$задачи , в котором блок π_iпредоставляет собой последовательность элементов i-й строки матрицы $[\beta_{ij}]_{m\times n}$, строится из допустимого решения πзадачи с помощью следующей процедуры:

1. Входные данные задачи:$[\beta_{ij}]_{m\times n}$– матрица, в которой элемент β_{ij} равен времени выполнения работы j на машине i; π– допустимое решение задачи , представленное перестановкой n строк матрицы $B = C^t[\beta_{rl}^0]_{n\times m'}$, $m' = nm$; если $\beta_{rl}^0 \in \pi$ и $\beta_{pq}^0 \in \pi$, то $r \neq p, l \neq q, k = 1$.

2. Пока $k \leq n$, определить индекс i блока π_i, которому принадлежит элемент β_{ij}, соответствующий элементу $\beta_{kl}^0 \in \pi$; индекс i находится с помощью соотношений$\beta_{kl}^0 = r\beta_{ij}, k = j, r = \overline{1,n} l = rm + i', 0 \leq i' \leq m$;при этом $i = m$, $\beta_{ij} \in \pi_m$, если $i' = 0$, и $i = i'$, $\beta_{ij} \in \pi_i$в противном случае;$k = k + 1$.

3. Упорядочить элементы каждого набора, полученного на шаге 2, по убыванию значения коэффициентов r, связывающих параметры β_{kl}^0и β_{ij}; в результате получим все m блоков допустимого решения $\Pi = (\pi_1, \pi_2, \dots \pi_i, \dots \pi_m)$ задачи . Следует отметить, что в решении $\Pi = (\pi_1, \pi_2, \dots \pi_i, \dots \pi_m)$допустимы пустые блоки.

Обратимся к примеру, иллюстрирующему связь задачи построения расписаний в постановке с задачей нахождения всех оптимальных решений .

Пусть $[\beta_{ij}]_{2\times 5} = \begin{bmatrix} 4 & 6 & 5 & 2 & 3 \\ 2 & 3 & 5 & 1 & 2 \end{bmatrix}$. Расписание П является последовательностью (π_1, π_2), содержащей две перестановки: перестановка π_1 задаёт порядок выполнения работ, назначенных на первую машину, а π_2 указывает, какие работы и в какой очерёдности выполняются на второй машине. Чтобы применить алгоритм решения задачи для нахождения всех оптимальных расписаний Π^* задачи , образуем из $[\beta_{ij}]_{2\times 6}$ матрицу

$$[\beta_{ij}^0]_{5\times 10} = \begin{bmatrix} 4 & 2 & 8 & 4 & 12 & 6 & 16 & 8 & 20 & 10 \\ 6 & 3 & 12 & 6 & 18 & 9 & 24 & 12 & 30 & 15 \\ 5 & 5 & 10 & 10 & 15 & 15 & 20 & 20 & 25 & 25 \\ 2 & 1 & 4 & 2 & 6 & 3 & 8 & 4 & 10 & 5 \\ 3 & 2 & 6 & 4 & 9 & 6 & 12 & 8 & 15 & 10 \end{bmatrix} \quad (3)$$

Рассмотрим допустимое решение: $\pi = (\beta_{11}^0, \beta_{24}^0, \beta_{32}^0, \beta_{46}^0, \beta_{58}^0), \beta_{11}^0 = 4, \beta_{24}^0 = 6, \beta_{32}^0 = 5, \beta_{46=3}^0, \beta_{58}^0 = 8$. Решению π соответствует расписание $\Pi = (\pi_1, \pi_2)$ где $\pi_1 = (\beta_{11})$, а π_2определяется перестановкой (β_{25}, β_{24}, β_{22}, β_{23}). Таким образом для расписания П, представленного на рис. 1, сумма моментов окончания работ, длительности выполнения которых определяются из матрицы $[\beta_{ij}]_{2\times 5}$, оказывается равной

$$\mathrm{mwft}(\Pi) = (\beta_{25} + \beta_{25} + \beta_{24} + \beta_{25} + \beta_{24} + \beta_{22} + \beta_{25} + \beta_{24} + \beta_{22} + \beta_{23} + \beta_{11}) = 15$$

$\beta_{11} = 4$	$\beta_{25}=2$	$\beta_{24}=1$	$\beta_{22}=3$	$\beta_{23}=5$

Рис. Расписание П

Приведенные рассуждения являются обоснованием корректного применения схемы минимизации функции $F_o(\pi)$ для решения обобщения задачи минимизации суммарного времени выполнения работ на неидентичных машинах, состоящего в построении множества всех расписаний с наименьшей суммой моментов завершения заданной совокупности работ. Следующий алгоритм находит все расписания Π^* за полиномиальное время [4]:

p. 1. $[\beta_{ij}]_{m\times n}$– входные данные задачи, где β_{ij} – время выполнения работы j на машине i.

p. 2. Из $[\beta_{ij}]_{m\times n}$ образовать матрицу $B = \left[[\beta_{ij}]^T, [2\beta_{ij}]^T, \dots, [r\beta_{ij}]^T, \dots, [n\beta_{ij}]^T\right]$cn строками и $m' = mn$ столбцами; $B = [\beta_{ij}^0]_{m\times n}$, – исходная матрица задачи минимизации функции $F_o(\pi)$.

p. 3. Построить множество всех решений π^* из n компонент, доставляющих минимум функции $F_o(\pi)$.

p.4. Выполнить шаги p1 – p3 алгоритма построения допустимого решения для исходных данных, представленных матрицей $B = [\beta_{ij}^0]_{m\times n}$.

p. 5. Для каждого полученного решения π^* выполнить процедуру построения соответствующего оптимального расписания $\Pi^* = (\pi_1^*, \pi_2^*, \dots, \pi_m^*)$.

Результаты. Предложенныйалгоритм обеспечивает построение всех оптимальных решений за время, зависящее от порядка входной матрицы и количества оптимальных подпоследовательностей, полученных в результате выполнения процесса вычислений.

Заключение. Вариации и видоизменения вычислительной схемы, расширяющие область ее применения, и достаточно широкий спектр задач, поддающихся решению, позволяют обозначить совокупность разработанных алгоритмов как метод построения локальных оптимальных последовательностей, ориентированный на изучение моделей распараллеливания операций на неидентичных машинах и моделей оптимального назначения работ. Алгоритмы оптимального упорядочения и назначения транспортных операций показали эффективность их использования для решения задач составления эффективных транспортных перевозок, что доказывает их востребованность при реализации логистических контуров ERP- и MES - систем.

Примечания:

1. Добрынин В.Н., Мороз В.В., Миловидова А.А. Унифицированная методика решения задачи расписания на основе задачи упорядочения // «Системный анализ в науке и образовании». М., 2010. №3.

2. Янков И.А. Параллельная обработка событий при построении и динамическом управлении сильносвязанными расписаниями / И. А. Янков, Б. Д. Шашков, С. В. Шибанов // Научный сервис в сети Интернет: масштабируемость, параллельность, эффективность: Труды Всероссийской суперкомпьютерной конференции (21-26 сентября 2009 г., г. Новороссийск). М.: Изд-во МГУ, 2009. С. 33-35. ISBN 978-5-211-05697-8

3. Скакалина Е.В. Подход к решению задачи оптимизации логістики агрохолдинга // Вестник Полтавской государственной аграрной академии (Научно-производственный отраслевой журнал). Полтава, 2013. № 4 (71). С. 139-145. КВ №17244-6014 ПР.

4. Скакалина Е.В. Эффективное построение множества расписаний с минимальным суммарным временем завершения работ // Радиоэлектроника и информатика. 2001. №3(16). С.44-46.

UDC 004.02.942

Effective Algorithm to Creating Multiple Optimal Schedules

Elena V. Skakalina

Poltava National Technical Yuri Kondratyuk University, Ukraine
24 PervomayskyProspekt, Poltava 36011
PhD, Associate Professor

Abstract. The article examines an algorithm to solve the issue of creating optimal schedules. The present research explores models of successive sequencing of work and its parallel execution. The article proposes an effective algorithm for creating all schedules with a minimal total time for completing “n” amount of work, which are being executed on “m” number of identical machines.

Keywords: the goal of optimising, creating schedules, optimal allocation of resources, models and methods of optimally allocating resources, ERP-systems, logistics.