

УДК 519.857

## УПРАВЛЯЕМЫЙ ФРОНТАЛЬНЫЙ АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ РАСПРЕДЕЛЕНИЯ РЕСУРСОВ В СЕТЕВЫХ КАНОНИЧЕСКИХ СТРУКТУРАХ

© 2008 г.

*М.Х. Прилуцкий, Е.А. Кумагина*

Нижегородский госуниверситет им. Н.И. Лобачевского

pril@iuni.unn.ru

Поступила в редакцию 09.09.2008

Рассматривается задача распределения ресурсов в сетевых канонических структурах. Предлагается фронтальный алгоритм ее решения, основанный на принципе оптимальности динамического программирования.

*Ключевые слова:* сетевые канонические структуры, распределение ресурсов, динамическое программирование, фронтальный алгоритм.

### Задача распределения ресурсов в канонических структурах

Изготовление сложных изделий требует выполнения совокупности взаимозависимых работ, связи между которыми описываются с помощью канонических сетевых моделей – ориентированных взвешенных графов без петель и контуров, элементам которых поставлены в соответствие некоторые характеристики. Каноничность сетевой модели означает, что никакая работа не может начать выполняться до тех пор, пока не завершатся все ей предшествующие по технологии изготовления работы.

При описании предметной области мы будем пользоваться понятиями: изделие – продукт трудовой деятельности, работа – составная часть изделия, ресурс – станок для выполнения работы.

Работы сетевых моделей мы будем характеризовать технологическими и организационными условиями. К технологическим условиям относятся: условия взаимозависимости выполнения работ, условия, связанные с длительностями выполнения работ, условия выполнения работ без перерывов и условия, связанные с тем, что на станке одновременно может выполняться только одна работа. К организационным условиям относятся условия, связанные с моментами начала и окончания выполнения работ.

Общая проблема управления процессом изготовления сложных изделий заключается в определении сроков начала и окончания выполнения работ на станках таким образом, чтобы, не нарушая требования технологического и, возможно, организационного характера, обеспечить изготовление всей заданной совокупности изделий.

### Математическая модель

*Исходные параметры математической модели*

$T = \{1, 2, \dots, T_0\}$  – множество тактов планирования.

$P = \{1, 2, \dots, k\}$  – множество изделий.

$J(p)$  – множество работ изделия  $p$ ,  $p \in P$ .

$J = \bigcup_{p \in P} J(p) = \{1, 2, \dots, n\}$  – множество всех работ.

$I = \{1, 2, \dots, m\}$  – множество станков.

*Технологические параметры*

$K(j)$  – множество работ, непосредственно предшествующих работе с номером  $j$ ,  $j \in J$ .

$R = \| r_{ij} \|_{m \times n}$  – матрица ресурсоемкостей, элемент которой  $r_{ij}$  определяет время выполнения работы  $j$  на станке  $i$ .

*Организационные параметры*

$q_j$  – ранний срок возможного начала выполнения работы  $j$ ,  $j \in J$ .

$D_j$  – директивный срок окончания выполнения работы  $j$ ,  $j \in J^D$ , где  $J^D$  – множество работ, имеющих директивные сроки,  $J^D \subseteq J$ .

*Варируемые параметры математической модели*

$x = (x_1, \dots, x_n)$  – вектор времен начала выполнения работ.

$y = (y_1, \dots, y_n)$  – вектор времен окончания выполнения работ.

$Z = \| z_{ijt} \|_{m \times n \times T_0}$ , где

$$z_{ijt} = \begin{cases} 1, & \text{если } i \text{ } \text{подает} \text{ } i \text{ } \text{в} \text{ } \text{время} \text{ } t \text{ } \text{и} \text{ } \text{работ} \text{ } j \text{ } \text{в} \text{ } \text{время} \text{ } t, \\ 0, & \text{иначе} \end{cases}$$

*Ограничения математической модели*

*Технологические ограничения*

Естественные ограничения на введенные переменные:

$$\begin{aligned} x_j \in T, y_j \in T, j \in J, \\ z_{ijt} \in \{0, 1\}, i \in I, j \in J, t \in T. \end{aligned} \quad (1)$$

Взаимозависимость работ, определяющая каноничность сетевой модели, задается ограничениями:

$$y_j \geq \max_{l \in K(j)} y_l, j \in J. \quad (2)$$

Требования выполнения работ на станке без перерывов:

$$y_j = x_j + r_{ij}, i \in I, j \in J. \quad (3)$$

Каждый станок одновременно выполняет одну работу:

$$\sum_{j \in J} z_{ijt} = 1, i \in I, t \in T. \quad (4)$$

Каждая работа выполняется на одном станке:

$$\sum_{i \in I} z_{ijt} = 1, j \in J, t \in T. \quad (5)$$

#### Организационные ограничения

Ограничения на сроки начала работ:

$$x_j \geq q_j, j \in J. \quad (6)$$

Ограничения на сроки окончания работ:

$$y_j \leq D_j, j \in J^D. \quad (7)$$

Исходные параметры, варьируемые параметры и ограничения (1)–(7) представляют собой общую математическую модель сетевого планирования.

### Постановка задачи

В рамках построенной общей математической модели ставятся различные оптимизационные задачи. Постановка задачи определяется двумя факторами: множеством допустимых решений и критерием (или группой критериев) оптимальности. Множество допустимых решений задается системой ограничений, которая выбирается из вышеописанных условий (1)–(7). Среди ограничений задачи обязательно должны присутствовать технологические ограничения и могут присутствовать ограничения организационного типа. Вместо ограничений организационного типа могут вводиться соответствующие критерии оптимальности.

Формально задача изготовления сложных изделий ставится следующим образом. Требуется найти совокупность варьируемых параметров (1), удовлетворяющих системе ограничений (2)–(6), для которых достигается минимального значения обобщенный критерий оптимальности (8), связанный со штрафными санкциями, налагаемыми на систему за нарушения директивных сроков выполнения работ:

$$F(x, y, z) = \sum_{j \in J^D} a_j \times \max(0, y_j - D_j), \quad (8)$$

где  $a_j$  – штрафные санкции за нарушение на один такт планирования работой  $j$  определенного для неё директивного срока,  $j \in J^D$ .

В дальнейшем решение задачи (1)–(8) мы будем называть расписанием выполнения работ или просто расписанием.

Поставленная задача относится к классу NP-трудных задач, т.е. для ее решения не существует эффективных алгоритмов определения оптимального решения [1].

### Управляемый фронтальный алгоритм

Для решения поставленной задачи предлагается фронтальный подход, основанный на идеологии «жадных алгоритмов».

При таком подходе в определенные такты времени строится «фронт работ» [2, 3]. Это множество работ, которые могут быть выполнены начиная с определенного такта. Возникает задача упорядочения работ из фронта работ. После установления порядка выполнения выбранная из фронта работ очередная работа включается в строящееся расписание, причем для ее выполнения используются все доступные к данному моменту времени ресурсы в максимально возможном объеме. Для установления порядка могут быть применены специальные перестановочные процедуры (порядок выполнения работ определяется путем решения специальных задач о назначениях).

В данной работе предлагается развить идею фронтального подхода и принимать решение о выборе очередной работы основываясь не на одном конкретном алгоритме, а на совокупности различных алгоритмов.

Будем моделировать процесс построения решения поставленной задачи системой без обратных связей. Пусть  $\hat{O}(t) = \{j_1, j_2, \dots, j_k\}$  – «фронт работ», определенный к такту планирования  $t$ , – это множество работ, любая из которых может начать выполняться с указанного момента времени,  $j_l \in J, l = \overline{1, k}, t \in T$ . Это те работы, для которых все им предшествующие работы уже выполнены, и ранние сроки начала этих работ не превышают указанного момента времени. Множество различных фронтов работ обозначим через  $W$ .

Фронт работ на заданный момент времени будем отождествлять с состоянием системы. Основная идея фронтального алгоритма заключается в преобразовании множества  $\hat{O}(t)$  в вектор, который определяет порядок последовательного включения очередной работы в строящееся решение. Для этого на множестве

работ из  $\hat{O}(t)$  необходимо установить линейный порядок. На основании линейного порядка, используя «строитель» расписания, будем применять процедуру, которая «жадным» образом включает работы в строящееся расписание.

Пусть  $Q$  – множество различных стратегий преобразования множества  $\hat{O}(t)$  в вектор. В качестве таких стратегий могут быть, например, стратегия, основанная на решении специальной задачи о назначениях [1], стратегия упорядочения работ по их резервам времени [4], стратегия, основанная на поиске перестановки заданной глубины [4], стратегия, основанная на алгоритме Метрополиса [4], и др.

Выбор стратегии будем отождествлять с управлением, применяемым к системе, результатом которого будет часть расписания, построенного «строителем» расписаний. Предполагается, что применение той или иной стратегии требует использования различных временных ресурсов. Пусть  $\tau_q$  – время, необходимое для реализации стратегии  $q$ ,  $q \in Q$ .

Обозначим через  $g(\hat{O}(t), q)$  суммарное время простоя станков для части расписания, построенного после применения стратегии  $q$  к фронту работ  $\hat{O}(t)$ .

Рассматриваемая система функционирует следующим образом. В состоянии  $\hat{O}(t)$  к системе применяется управление, задаваемое стратегией  $q$ , которое приносит системе «доход»  $\tau_q$ , связанный с временем на реализацию стратегии  $q$ . Под воздействием управления  $q$  множество работ из фронта преобразуется в вектор, по которому, используя «строитель» расписания, в уже построенное к этому моменту времени расписание добавляются работы.

При этом система перейдет в новое состояние  $\hat{O}(t')$ , где  $t'$  – время завершения выполнения самой первой работы из части построенного расписания. Такой переход приносит системе «доход», задаваемый величиной  $g(\hat{O}(t), q)$ . При этом предполагается, что величины  $\tau_q$  и  $g(\hat{O}(t), q)$  измеряются в одних и тех же единицах и суммарный «доход» от перехода системы из состояния  $\hat{O}(t)$  в состояние  $\hat{O}(t')$  под воздействием управления  $q$  определяется величиной  $\tau_q + g(\hat{O}(t), q)$ .

В новом состоянии  $\hat{O}(t')$  к системе применяется управление  $q'$ ,  $q' \in Q$ , и т.д. Система функционирует конечное число тактов  $T_0$  – до мо-

мента построения «полного» расписания, включающего в себя все работы из множества  $J$ .

Относительно рассматриваемых систем обычно ставят следующую задачу [5]: при заданном состоянии системы и числе тактов, оставшихся до конца функционирования, найти оптимальную стратегию из некоторого класса стратегий по критерию максимизации суммарного «дохода». Будем задавать стратегию в виде функции  $f(\hat{O}(t), T)$ , определенной на множестве  $W \times \{1, 2, \dots, T_0\}$  со значениями из  $Q$ . Функция  $f(\hat{O}(t), T)$  в зависимости от состояния (фронта работ) и числа тактов, оставшихся до конца функционирования (в общем случае – от числа работ, которые еще не включены в строящееся расписание), определяет, с помощью какой стратегии необходимо установить линейный порядок на множестве работ из фронта работ.

Пусть  $S$  – множество различных стратегий. Обозначим через  $\mu(\hat{O}(t), T, f(\hat{O}(t), T))$  суммарный «доход», который получит система, если она начинает функционировать в состоянии  $\hat{O}(t)$ , до конца функционирования осталось  $T$  тактов, и к системе применяются управления, задаваемые функцией  $f(\hat{O}(t), T)$ . Тогда задача выбора оптимальной стратегии будет заключаться в определении такой функции  $f_0(\hat{O}(t), T)$ , для которой выполняется

$$\mu(\Phi(t), T, f_0(\Phi(t), T)) = \max_{f(\Phi(t), T) \in S} \mu(\Phi(t), T, f(\Phi(t), T)).$$

Так как поведение системы в момент времени  $t$  не зависит от предыстории процесса и общий «доход» системы складывается из «доходов», полученных на каждом шаге, то к системе применим принцип оптимальности динамического программирования.

Обозначим через  $v(\hat{O}(t), T)$  суммарный «доход», который получит система, если она начинает свое функционирование из состояния  $\hat{O}(t)$ , до конца функционирования осталось  $T$  тактов и к системе на каждом шаге применяются оптимальные управления. Тогда, применив к системе принцип оптимальности динамического программирования, построим рекуррентные соотношения:

$$v(\hat{O}(t), T) = \min_{q \in Q} \{ \tau_q + g(\hat{O}(t), q) + v(\hat{O}(t'), T - 1) \},$$

из которых, задав граничные «доходы»  $v(\hat{O}(t), 0) = 0$ ,  $\hat{O}(t) \in W$ , можно находить оптимальную стратегию.

### Заключение

В работе приводится общая математическая модель распределения ограниченных ресурсов в

сетевых канонических структурах. В рамках математической модели поставлена задача минимизации суммарного штрафа за нарушения директивных сроков работ при изготовлении сложных изделий. Для решения этой задачи предложен управляемый фронтальный алгоритм, в котором упорядочивание работ осуществляется на основе решения задачи динамического программирования.

*Список литературы*

1. Прилуцкий М.Х., Кумагина Е.А. Задачи распределения разнородных ресурсов в сетевых канонических структурах // Перспективные информационные технологии и интеллектуальные системы. 2000. № 4. С. 46–52.
2. Прилуцкий М.Х., Кумагина Е.А. Задача упорядочения работ как задача о назначениях // Вестник

Нижегородского государственного университета. Математическое моделирование и оптимальное управление. Н. Новгород: Изд-во ННГУ, 1999. Вып. 21. С. 270–275.

3. Прилуцкий М.Х., Кумагина Е.А. Перестановочные процедуры решения задач распределения ресурсов // Межвузовский сборник научных трудов «Прикладные задачи моделирования и оптимизации». Воронеж, 2000. Ч. II. С. 81–90.
4. Прилуцкий М.Х., Власов С.Е. Многостадийные задачи теории расписаний с альтернативными вариантами выполнения работ // Системы управления и информационные технологии. 2005. № 2. С. 44–48.
5. Казанцев Э.Н., Прилуцкий М.Х. Об одном классе управляемых марковских цепей // Успехи математических наук. 1978. Том 33. Вып. 6 (204). М., С. 213–214.

**CONTROLLED FRONTAL ALGORITHM FOR RESOURCE ALLOCATION PROBLEM IN NETWORK CANONICAL STRUCTURES**

*M.Kh. Prilutsky, E.A. Kumagina*

The problem of limited-resource allocation in network canonical structures is considered. The frontal algorithm based on dynamic programming optimality principle is proposed.