

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ И ОПТИМАЛЬНОЕ УПРАВЛЕНИЕ

УДК 519.8

ОБЩАЯ ЗАДАЧА ОДНОПРОЦЕССОРНОГО ОБСЛУЖИВАНИЯ ПРОСТРАНСТВЕННО РАССРЕДОТОЧЕННОЙ ГРУППЫ СТАЦИОНАРНЫХ ОБЪЕКТОВ

© 2007 г.

Ю.С. Федосенко, А.Ю. Шлюгаев

Волжская государственная академия водного транспорта

fds@aqua.sci-nnov.ru*Поступила в редакцию 16.04.2007*

Рассматривается дискретная математическая модель однофазного обслуживания автономно перемещающимся процессором пространственно рассредоточенной группы стационарных объектов. С каждым объектом ассоциирован индивидуальный штраф и продолжительность обслуживания, являющиеся монотонно возрастающими функциями от момента времени завершения и начала обслуживания соответственно; известны также продолжительности перемещений между объектами. Задача заключается в построении стратегии обслуживания, обеспечивающей минимум суммарного штрафа по всем объектам группы. На основе идеологии динамического программирования строится алгоритм синтеза оптимальных стратегий обслуживания и приводятся результаты вычислительных экспериментов.

Введение

Формулируется дискретная модель и решается задача синтеза оптимальной стратегии однофазного однократного обслуживания автономно перемещающимся процессором – mobile-процессором [1] группы стационарных объектов, расположенных в границах существенно протяженной рабочей зоны.

Принципиальным отличием рассматриваемой математической модели от ранее исследованных [2, 3] является двумерный характер зоны расположения объектов, представляемой односвязным неориентированным взвешенным графом. Вершины графа определяют точки расположения объектов, а веса ребер – расстояния между парами смежных точек, т.е. таких, между которыми осуществимо непосредственное перемещение процессора. Продолжительности перемещений процессора между парами смежных вершин графа известны.

На пути перемещения процессора в точку расположения очередного выбранного для обслуживания объекта могут располагаться точки расположения других объектов группы.

Такие точки проходятся процессором транзитом, т.е. соответствующие им объекты не обслуживаются; при этом количество транзитных проходов не ограничено.

Для каждого объекта известны момент времени готовности к обслуживанию и продолжительность обслуживания в зависимости от момента его начала. По прибытии в точку расположения объекта процессор ожидает наступление момента готовности объекта к обслуживанию, если этот момент еще не наступил, и затем выполняет его обслуживание без прерываний. В случае завершения обслуживания объекта позднее некоторого определенного для него момента времени начисляется штраф, величина которого зависит от величины запаздывания.

Задача заключается в построении стратегии обслуживания, обеспечивающей минимизацию суммарного штрафа по всем объектам группы.

На содержательном уровне рассматриваемая модель описывает технологию снабжения топливом группы разнотипных плавучих дизель-электрических комплексов, осуществляющих донную добычу нерудных строительных материалов в крупномасштабной акватории с разветвленной русловой

топологией. В качестве примера такого производственного полигона укажем Камский грузовой район, протяженность которого от устья р. Камы до устья р. Белой порядка 400 км. В разные периоды навигации на полигоне работает от 8 до 20 и более плавучих добывающих комплексов (ПДК) с ежедневной суммарной потребностью в дизельном топливе от 100 до 500 тонн, которое доставляется к ПДК закрепленным за районом специализированным танкером-заправщиком. В условиях высоких расходов по содержанию ПДК и цен на дизельное топливо, сокращения возможностей предприятий по созданию его технологически значимых оперативных запасов основная задача диспетчера района заключается в минимизации экономических потерь, связанных с непроизводительными простоями ПДК. При этом существенное влияние на работу диспетчера оказывают жесткие ограничения на допустимую длительность штатного регламента формирования управляющих воздействий путем решения соответствующих экстремальных задач в (on-line) – (off-line)-режимах. В силу сказанного актуальной является проблема разработки достаточно быстрых (по условиям конкретного применения) алгоритмов синтеза стратегий снабжения топливом добывающих комплексов.

Постановка задачи

Имеется n -элементная группа $O_n = \{o(1), o(2), \dots, o(n)\}$ независимых стационарных объектов, расположенных в соответствующих вершинах i ($i = \overline{1, n}$) неориентированного односвязного графа G ; граф G содержит также некоторую базисную вершину o . Все объекты группы O_n подлежат однократному обслуживанию автономно перемещающимся процессором P , для которого граф G является рабочей зоной. При этом в начальный момент времени процессор P расположен в базисной вершине o . Обслуживание объекта $o(i)$ осуществимо, начиная с момента $t(i)$ ($i = \overline{1, n}$) готовности объекта к обслуживанию.

По каждому объекту $o(i)$ заданы функции: $t(i, t)$ – продолжительность обслуживания, если оно начинается в момент времени t ($i = \overline{1, n}, t \geq t(i)$), и $j(i, \bar{t})$ – индивидуальный штраф за обслуживание, завершающееся в момент времени \bar{t} ($i = \overline{1, n}, \bar{t} \geq 0$). Обе функции

являются монотонно возрастающими (в нестрогом смысле) по второму аргументу.

Считается известной матрица $W = \{w(i, j)\}$ размерности $(n+1) \cdot (n+1)$, где $w(i, j)$ – продолжительность перемещения процессора P из вершины i в вершину j непосредственно; если такое перемещение недопустимо, то $w(i, j) = \infty$. Отметим, что продолжительность перемещения из вершины i в вершину j может оказаться меньше величины $w(i, j)$, если перемещение осуществляется не непосредственно, а через некоторые промежуточные вершины графа G .

Стратегию обслуживания процессором P объектов группы O_n определяем как перестановку $r = (i_1, i_2, \dots, i_n)$ элементов множества $\{1, 2, \dots, n\}$; объект группы O_n с номером i_k согласно стратегии r обслуживается k -м по очереди ($k = \overline{1, n}$).

Обозначим через $t_{\text{нач}}(r, i_k)$ и $t^*(r, i_k)$ соответственно моменты начала и завершения обслуживания объекта $o(i_k)$ при реализации стратегии r . Считаем, что все допустимые реализации

r компактны, т.е. имеют место соотношения:

$$t_{\text{нач}}(r, i_1) = \max\{l(0, i_1); t(i_1)\}; \quad (1)$$

$$t^*(r, i_k) = t_{\text{нач}}(r, i_k) + t(i_k, t_{\text{нач}}(r, i_k)), \\ k = \overline{1, n};$$

$$t_{\text{нач}}(r, i_k) = \max\{t^*(r, i_{k-1}) + l(i_{k-1}, i_k); t(i_k)\}, \\ k = \overline{2, n}.$$

В соотношениях (1), (2) через $l(i, j)$ обозначены элементы матрицы $L = \{l(i, j)\}$, получаемой из W путем применения алгоритма Флойда [4]. Результатом работы алгоритма Флойда также является матрица $A = \{a(i, j)\}$, необходимая для синтеза оптимальных по времени перемещения путей.

Совокупный штраф по всем объектам группы O_n , обслуживаемым согласно стратегии

$$r, \text{ есть } Q(r) = \sum_{q=1}^n j(q, t^*(r, q)).$$

Требуется найти стратегию r^* , которая доставляет минимальное значение функции $Q(r)$, т.е. найти решение следующей однокритериальной проблемы

$$\min_{\rho} Q(\rho). \quad (3)$$

Задача (3) NP-трудна, поскольку к ней может быть сведена задача коммивояжера [5]. Также, как и в случае задачи коммивояжера, для решения (3) может быть применен метод динамического программирования [6].

Вывод соотношений динамического программирования

В соответствии с общей концепцией построение рекуррентных соотношений динамического программирования направлено на формирование вычислительной схемы синтеза стратегии r^* как некоторой пошаговой процедуры. Каждый очередной шаг этой процедуры определяется множеством номеров обслуженных на данный момент времени объектов и номером последнего обслуженного объекта. При этом в прямой вычислительной схеме пошаговое продвижение реализуется в направлении от момента времени начала до момента завершения обслуживания группы объектов.

Пусть t – произвольный момент принятия решения (МПР), т.е. это либо момент времени 0, либо момент завершения обслуживания произвольного объекта группы O_n по синтезируемой стратегии. Решение, принимаемое в МПР, заключается в выборе объекта, который будет обслуживаться следующим.

Положим, что в момент времени t завершено обслуживание объекта $o(i)$ (по состоянию на момент времени 0 считаем завершенным обслуживание формально вводимого объекта $o(0)$), а S – совокупность объектов из группы O_n , которые были обслужены раньше, чем $o(i)$. Создающуюся в рассматриваемый МПР ситуацию полностью описывает трехкомпонентный кортеж (t, i, S) . Тройки вида (t, i, S) далее будем называть состояниями системы. При этом начальное состояние определяется как кортеж $A_0 = (0, 0, \emptyset)$, а финальные состояния системы имеют общий вид $(t, i, O_n \setminus \{o(i)\})$, где $o(i)$ – произвольный объект из O_n .

Кортеж $A = (t_1, j, S_1)$ относится к числу состояний, непосредственно предшествующих состоянию $B = (t_2, i, S_2)$, если: $S_2 = S_1 \cup \{o(j)\}$, объект $o(i)$ множеству S_2 не принадлежит и $t_2 = t_2^a + t(i, t_2^a)$, где $t_2^a = \max\{t_1 + l(j, i); t(i)\}$. Для того чтобы из состояния A реализовать непосредственный (одношаговый) переход в состояние B , необходимо, начиная от момента времени t_1 , совершить два последовательных действия: 1) переместить процессор из вершины j в вершину i ; 2) выполнить обслуживание объекта $o(i)$.

Затраты на перевод системы из состояния A непосредственно в состояние B оказываются равными $j(i, t_2)$. Эти затраты именуем стоимостью перехода из состояния A в B . Если состояние A для B является непосредственно предшествующим, то состояние B для A относится к числу непосредственно следующих, т.е. порождаемых состоянием A .

Минимально возможную величину суммарных затрат при переводе системы из ее начального состояния в состояние (t, i, S) обозначим через $\Sigma(t, i, S)$. Как очевидно,

$$\Sigma(0, 0, \emptyset) = 0. \quad (4)$$

Одношаговый переход в произвольное, но отличное от начального состояние (t, i, S) осуществляется из любого непосредственно предшествующего ему состояния. По функции $t(i, t)$ легко определяется используемая ниже функция $D^*(i, t)$ – требуемый момент начала обслуживания объекта $o(i)$, если его обслуживание должно завершиться в момент времени t . В случае когда $D^*(i, t) = t(i)$, т.е. обслуживание объекта $o(i)$ начинается непосредственно в момент его готовности, момент времени $D^*(i, t)$ не обязательно совпадает с моментом времени прибытия процессора после завершения обслуживания предшествующего объекта $o(j)$. Очевидно, что в такой ситуации процессор может прибыть в точку расположения объекта $o(i)$, начиная от момента времени $t_a(i, j) = t(j) + t(j, t(j)) + l(j, i)$ и заканчивая моментом времени $D^*(i, t)$. При этом $t(j) + t(j, t(j))$ – наиболее ранний из возможных момент времени завершения обслуживания предшествующего объекта.

Учитывая данное обстоятельство, согласно принципу динамического программирования имеем:

$$\Sigma(t, i, S) = \min_{o_j \in S} \{ \Sigma^P(D^*(i, t) - \quad (5)$$

$$- l(j, i), j, S \setminus \{o(j)\}) + j(i, t) \}.$$

В формуле (5)

$$\Sigma^P(t, i, S, j) = \begin{cases} \Sigma(D^*(i, t) - l(j, u), j, S \setminus \{o(j)\}), \\ \text{при } D^*(i, t) > t(i); \\ \min_{t_p \in T_p} \{ \Sigma(t_p, j, S \setminus \{o(j)\}) \}, \\ \text{при } D^*(i, t) = t(i), \end{cases}$$

где через T_p обозначен целочисленный отрезок $[t(j) + t(j, t(j)), D^*(i, t) - l(j, i)]$.

Оптимальное значение Q_{opt} критерия $Q(\rho)$, соответствующее решению задачи (1), очевидно есть

$$Q_{opt} = \min_{i \in \{1, 2, \dots, n\}} \Sigma(t, i, O_n \setminus \{o(i)\}).$$

Оценка временной вычислительной сложности процедуры синтеза оптимальной стратегии обслуживания

Обозначим через Δ алгоритм, реализующий процедуру синтеза стратегии r^* по рекуррентным соотношениям (4), (5). Его вычислительная сложность определяется количеством наборов (t, i, S) , на которых подсчитываются значения функции $\Sigma(t, i, S)$, и сложностью вычисления каждого значения. В рассматриваемых тройках (t, i, S) число значений первого аргумента ограничено максимальным значением момента времени завершения обслуживания t_n , которое можно оценить сверху следующим неравенством:

$$t_n \leq \sum_{i=0}^{n-1} \max_{\substack{j=1, n \\ j \neq i}} l(i, j) + \max_{j=1, n} t(j) + \sum_{i=1}^n t(i, \max_{j=1, n} t_j).$$

Общее количество различных пар значений второго и третьего аргументов в наборах (t, i, S) , как очевидно, не превышает $n \cdot 2^{n-1}$. Число элементарных операций, выполняемых при подсчете каждого отдельного значения функции $\Sigma(t, i, S)$, мажорируется линейной функцией параметра n .

Следовательно, временная вычислительная сложность алгоритма Δ оценивается сверху величиной $O(n^2 2^{n-1} t_n)$.

Результаты вычислительных экспериментов

Для проведения вычислительных экспериментов на языке Java была выполнена программная реализация алгоритма Δ .

Эксперименты проводились для $n = 8, 9, \dots, 12$ путем решения тестовых наборов, каждый из которых состоял из 100 задач. При этом, рассматривались функции-константы затрат времени на обслуживание $t(i, t) = dt_i$ и функции индивидуального штрафа вида

$$j(i, \bar{t}) = \begin{cases} 0, & \text{при } \bar{t} \leq t(i) + t(i, t(i)) + td_i; \\ b_i \cdot e^{a_i [t - t(i) - t(i, t(i)) - td_i]}, & \\ \text{при } \bar{t} > t(i) + t(i, t(i)) + td_i (i = \overline{1, n}). \end{cases}$$

Числовые характеристики решаемых задач генерировались по равномерному закону распределения из следующих диапазонов значений: $t(i) \in [20, 180]$, $dt_i \in [1, 10]$, $td_i \in [1, 10]$, $a_i \in [2, 10]$, $b_i \in [1, 10]$, $w_{ij} \in [4, 20]$.

В таблице приведена экспериментальная статистика по оценке временной (продолжительность синтеза t_c) и объемной (требуемый в ходе вычислений объем памяти V) вычислительной сложности алгоритма Δ . Ячейки таблицы имеют следующий формат: [минимальное значение; максимальное значение], среднее значение.

Таблица

n	t_c, c	V, Kb
8	[0,078; 0,656], 0,334	[5,273; 24,416], 13,839
9	[0,892; 5,498], 2,256	[27,685; 78,799], 48,610
10	[5,919; 82,929], 29,199	[79,664; 231,730], 139,807
11	[66,957; 1168,51], 307,59	[204,38; 858,16], 407,77
12	[993,48; 9209,08], 2867,11	[818,64; 2446,53], 1186,45

Для решения большеразмерных задач (при $n \geq 12$) целесообразна разработка модификаций метода динамического программирования, активно использующих идеи метода ветвей и границ, а также поиск эффективных алгоритмов синтеза субоптимальных стратегий обслуживания, в частности использующих концепцию d-расписаний [7] и идеи «природных» вычислений [8].

Заключение

1. Математическое описание технологии снабжения топливом группы плавучих разнотипных дизель-электрических комплексов, осуществляющих донную добычу нерудных строительных материалов в крупномасштабной акватории с разветвленной русловой топологией выполнено в работе на основе дискретизации пространственных, временных и других параметров модели. Очевидно, что степень адекватности принятого формализма реальным транспортно-технологическим процессам зависит от выбранного шага дискретности. В этом смысле повышение точности решения задачи синтеза оптимальной стратегии может быть выполнено за счет выбора единиц измерения координат и параметров рассматриваемой модели.

2. В качестве непосредственного обобщения исследованной задачи для приложений

значимыми являются модели, учитывающие модификации:

а) заданы две базисные вершины: начальная – o_n и конечная – o_k ; обслуживание всех объектов группы должно быть выполнено при перемещении процессора из точки o_n в точку o_k ;

б) заданы директивные сроки завершения обслуживания для всех или только для привилегированных объектов группы;

в) учет суммарных затрат процессора на перемещения, в том числе в бикритериальной постановке [9].

Список литературы

1. Коган, Д.И. Проблема синтеза оптимального расписания обслуживания бинарного потока объектов mobile-процессором / Д.И. Коган, Ю.С. Федосенко, А.В. Шеянов // Труды III Международной конференции «Дискретные модели в теории управляющих систем», М. 1998 г. – М.: Изд-во МГУ, 1998. – С. 43–46.

2. Коган, Д.И. Задачи оптимального обслуживания группы стационарных объектов, расположенных в одномерной зоне / Д.И. Коган, А.В. Синий, Ю.С. Федосенко // Вестник ВГАВТ. Межвузовская серия: Моделирование и оптимизация сложных систем. – Н. Новгород: Изд-во ВГАВТ, 2004. Вып. 9. – С. 27–34.

3. Синий, А.В. Математические модели и алгоритмы синтеза оптимальных расписаний бункеровки добывающих комплексов в крупномасштабных русловых районах / А.В. Синий, Ю.С. Федосенко // Технологии Microsoft в теории и практике программирования: сб. материалов конф. Н. Новгород, 21–22 марта 2006 г. – Н. Новгород: Изд-во Нижегородского госуниверситета им. Н.И. Лобачевского, 2006. – С. 276–279.

4. Асанов, М.О. Дискретная математика. Графы, матроиды, алгоритмы / М.О. Асанов, В.В. Баранский, В.В. Расин. – Ижевск: НИЦ «Регулярная и хаотическая динамика», 2001. – 288 с.

5. Гэри, М. Вычислительные машины и труднорешаемые задачи / М. Гэри, Д. Джонсон. – М.: Мир, 1982. – 416 с.

6. Беллман, Р. Динамическое программирование / Р. Беллман, С. Дрейфус. – М.: ИЛ, 1960. – 400 с.

7. Коган, Д.И. Задача диспетчеризации: анализ вычислительной сложности и полиномиально разрешимые подклассы / Д.И. Коган, Ю.С. Федосенко // Дискретная математика. 1996. Т. 8. № 3. С. 135–147.

8. Макконнелл, Дж. Основы современных алгоритмов / Дж. Макконнелл. – М.: Техносфера, 2006. – 368 с.

**GENERAL PROBLEM OF OPTIMAL SINGLE-PROCESSOR SERVICING
OF A GROUP OF SPATIALLY-DISTRIBUTED OBJECTS**

1:

Yu.S. Fedosenko, A.Yu. Shlyugaev

We consider a discrete mathematical model of single-phase servicing of a group of spatially-distributed fixed objects by an autonomously-moving processor. Every object has its associated fine and service-duration functions which increase monotonically with the service completion and beginning time, respectively. In addition, durations of the processor motions between the objects are assumed known. The problem consists in constructing an object-servicing strategy which minimizes the value of the total fine criterion. Based on the dynamic-programming approach, we develop an algorithm for synthesizing optimal servicing strategies and present the results of computational experiments.

9. Коган, Д.И. Динамическое программирование и дискретная многокритериальная оптимизация: Учебное пособие / Д.И. Коган. – Н. Новгород: Изд-во ННГУ, 2005.