УДК 004.02

ТЕХНОЛОГИЯ ОЦЕНКИ СЛОЖНОСТИ ДЛЯ ВЫБОРА МЕТОДА РЕШЕНИЯ ЗАДАЧИ СОСТАВЛЕНИЯ РАСПИСАНИЯ

Добрынин Владимир Николаевич[1], Миловидова Анна Александровна[2]

[1]Кандидат технических наук, профессор Института системного анализа и управления;
ГОУ ВПО «Международный Университет природы, общества и человека «Дубна»,
141980, Московская обл., г. Дубна, ул. Университетская, 19;
e-mail: arbatsolo@yandex.ru.

[2]Аспирант;
ГОУ ВПО Международный университет природы, общества и человека «Дубна»;
Институт системного анализа и управления;
141980, Московская обл., г. Дубна, ул. Университетская, 19;
e-mail: milanna.ya@gmail.com.

Проблема составления расписания характеризуется тем, что, несмотря на многообразие методов решения ее в различных предметных областях, отсутствует унифицированный инвариантный метод. Следует отметить, что теория расписания даёт ответ на этот проблемный вопрос. Однако не решена технологическая задача сведения разнообразных содержательных постановок к задаче упорядочения и оценки сложности для выбора эффективного метода решения данной задачи.

Ключевые слова: задача составления расписания, задача упорядочения, оценка сложности задачи составления расписания.

ASSESSMENT OF COMPLEXITY TECHNOLOGY FOR CHOICE OF SOLUTION METHOD OF SCHEDULING PROBLEM

Dobrynin Vladimir[1], Milovidova Anna[2]

[1]Candidate of Science in Engineering, professor of Institute of system analysis and management;
Dubna International University of Nature, Society and Man,
Institute of system analysis and management;
141980, Dubna, Moscow reg., Universitetskaya str., 19;
e-mail: arbatsolo@yandex.ru.

[2]Postgraduate student;
Dubna International University of Nature, Society and Man,
Institute of system analysis and management;
141980, Dubna, Moscow reg., Universitetskaya str., 19;
e-mail: milanna.ya@gmail.com.

Scheduling problem is characterized by the fact that, despite the variety of methods for it decision in various subject areas, there is no unified invariant method. It is necessary to notice that the schedule theory answers on this problem question. However the technological reduction problem of various substantial statements to a problem of streamlining and an estimation of complexity for a choice of an effective problem decision method isn't solved.

Keywords: scheduling problem, streamlining problem, an estimation of complexity of scheduling problems.

Введение

Время – важнейший жизненный ресурс. Жан де Лабрюйер писал: «Сожаление о неразумно растраченном времени, которому предаются люди, не всегда помогает им разумно употребить его остаток». Для того чтобы не сожалеть о растраченном времени, необходимо им управлять.

В настоящее время существует множество стратегий управления временем, ведутся исследования в этой области, разрабатываются методы. Управление временем (тайм-менеджмент, time management, организация времени) – это технология организации времени и повышения эффективности его использования [1].

Основное содержание управления временем составляет планирование расходов времени на решения разнообразных задач в соответствии с их срочностью и важностью. Простейшей формой такого планирования служит составление расписания, где в хронологическом порядке фиксируются все предстоящие задачи, требующие решения.

Расписание – некоторая совокупность указаний относительно того, какие именно требования какими именно ресурсами обслуживаются в каждый момент времени [2].

Хрестоматийным стал один из первых экспериментов Тейлора, героем которого стал грузчик по фамилии Шмидт. Он был одним из членов бригады, где на каждого приходилась устрашающая дневная норма – требовалось погрузить в железнодорожные вагоны по двенадцать с половиной тонн чугунных болванок. Рабочие клялись, что выполнить эту норму очень трудно, а перевыполнить – невозможно. Однако Шмидт, следуя подробным инструкциям Тейлора по выполнению трудового процесса, сумел превысить норму почти в четыре раза. Соответственно увеличился его заработок [3].

Таким образом, расписание является полезным инструментом в организации человеческой деятельности. Для многих людей составление расписания является профессиональной обязанностью. Они организовывают учебный процесс, составляют расписание движения транспорта, разрабатывают календарные планы работы предприятий и учреждений и т.д. Зачастую довольно сложная задача – увязать со временем множество действий, сопряженных с достижением заданной цели.

Расписание используется:

- в управлении проектами (для выбора той или иной очередности выполнения работ);
- в оперативно-календарном планировании (при составлении план-графиков загрузки оборудования);
- в системах технологической подготовки производства (при формировании маршрутно-операционных технологических процессов);
- в производственной логистике (при планировании объемов и мест складирования производственных заделов);
- в транспортной логистике (при составлении оптимальных схем грузоперевозок);
- при организации занятий в ВУЗе, рейсов самолетов, движения поездов, обслуживании клиентов и т.д.

Еще в середине 50-х годов XX века возникла потребность в глубоких исследованиях в данном направлении в связи с внедрением автоматизированных систем управления производством. Решение задач, связанных с составлением расписания выполнения работ в производственных системах привело к созданию теории решения задач составления расписаний [4-7].

В это время появляется термин «теория расписаний». Теория расписаний – раздел дискретной математики, занимающийся проблемами упорядочения.

В настоящее время интерес к решению задач расписания возрос в связи с оживлением промышленного производства. Но многие авторы отмечают, что появились новые задачи, которые вызывают определенные трудности. В свое время был сформулирован класс так называемых трудно решаемых задач теории расписания (*NP*-трудных), для решения которых еще не найдено эффективных алгоритмов решения. Именно к таким задачам можно отнести большинство реальных задач производственного планирования.

Вместе с тем нужно отметить, что по сравнению с серединой прошлого века рост вычислительной мощности компьютерных систем, появление современных суперкомпьютеров и кластеров рабочих станций сделали возможным решение многих задач. Постоянное повышение мощности компьютерных систем расширяет возможности разработчиков в применении более эффективных методов решения; задачи расписания, которые еще в недалеком прошлом не могли быть решены в реальном масштабе времени, успешно решаются, благодаря использованию параллельных алгоритмов, реализуемых на многопроцессорных системах, обладающих высоким быстродействием. Это делает актуальным возврат к решению задач составления расписания.

В наиболее общей формулировке задача составления расписания состоит в следующем. Среди множества работ и множества ресурсов (человеческих, технологических и др.), обеспечивающих совершение данных работ, должны быть распределены временные ресурсы таким образом, чтобы была выполнена некоторая фиксированная система работ. Цель состоит в том, чтобы при заданных свойствах работ и ресурсов и наложенных на них ограничениях найти эффективный алгоритм упорядочения работ во времени, оптимизирующий желаемую меру эффективности.

Все известные работы в этом направлении [8-20] можно условно разделить на две группы. К первой группе относятся работы, использующие классические методы: методы динамического программирования, целочисленного программирования, нелинейного программирования, методы имитационного моделирования: методы ветвей и границ [21], метод раскраски графа [10, 22], задача о назначениях [23] и др. Вторая группа работ основана на современных методах решения задач целочисленного программирования, использующих эвристические алгоритмы решения данных задач [21, 22, 24, 25, 26].

Отсутствие комплексного подхода связано с тем, что эффективность составления расписания зависит от большого количества факторов. Решение технологической задачи сведения разнообразных содержательных постановок задач расписания другими словами унификация таких задач позволит значительно сократить время, затрачиваемое на выбор метода решения задачи для фиксированной предметной области.

1. Систематизация задач «расписания»

Рассмотрим некоторую обобщенную классификацию задач составления расписания, построенную на основе анализа особенностей характеристик ресурсов (работ и машин), а также критериев (целевых функций), предложенную Юсуповой Н. И., Сметаниной О. Н., Ахтариевым А. А. в статье «Об одной классификации задач составления расписаний» [27].

В предлагаемой классификации любая задача может быть записана следующим образом: $\alpha|\beta|\gamma$, где α – характеристика машин (Таблица 1); β – характеристики работ (Таблица 2); γ – целевая функция задачи (Таблица 3). Работы состоят из операций $J_i = \{O_{i_1}, O_{i_2}, \dots, O_{i_n}\}$. Операция O_{i_j} требует p_{ij} времени и может выполняться на одной из машин множества $\mu_{ij} \subseteq \{M_1, \dots, M_m\}$. Если $|\mu_{ij}| = 1, \forall ij,$ то получаем модель с предписаниями. Если $|\mu_{ij}| = m, \forall ij,$ то получаем модель с параллельными машинами.

Поле α состоит из 2 частей $\alpha = \alpha_1\alpha_2$, где α_1 – характеристики машин.

Целевые функции обозначаются, как c_i – время окончания работы J_i. Рассматриваются два типа минимизируемых целевых функций: $f(c) = max_i\, f_i(c_i)$ и $f(c) = \sum_{i=1}^{n} f_i(c_i)$.

Особенности задач теории составления расписаний в зависимости от характеристик машин, работ, целевой функции приведены на рис. 1. В левой части рисунка представлены возможные целевые функции (C_{max} – время окончания всех работ, L_{max} – запаздывание относительно директивных сроков, $\sum_{i=1}^{n} w_i c_i$ – взвешенная сумма окончания работ). Посередине находятся характеристики машин: одна машина, параллельные машины, рабочий цех и потоковая и открытая линии. А справа характеристики работ: разбиение на группы, с разрешенными прерываниями и с условиями предшествования.

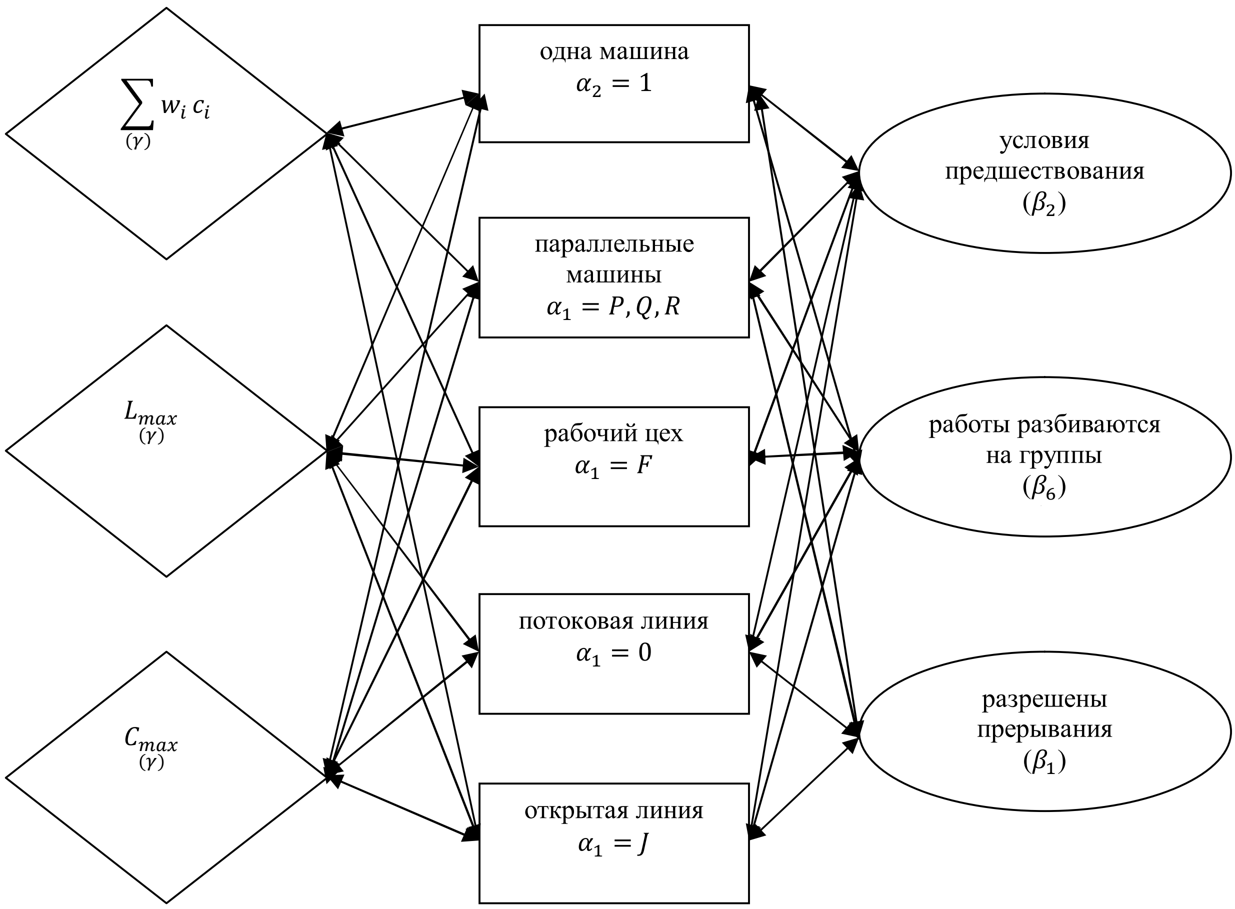

Рис. 1. Особенности составления расписания в зависимости от целевой функции, характеристики машин (ресурсов) характеристики работ

Таблица 1. Особенности задач составления расписания в зависимости от характеристик машин

Значения α_1	*Характеристики машин*
$\boldsymbol{\alpha_1 = \emptyset}$	Для каждой работы задана машина для ее выполнения
$\boldsymbol{\alpha_1 = P}$	Машины параллельны и одинаковы $p_{ij} = p_i$
$\boldsymbol{\alpha_1 = R}$	Машины параллельны, длительности выполнения работ произвольны, но $p_{ij} = p_i/s_{ij}$
Если $\boldsymbol{\alpha_1} \in \{\emptyset, \boldsymbol{P}, \boldsymbol{Q}, \boldsymbol{R}\}$,	То $n_i = 1 \forall j$, т.е. каждая работа состоит из одной операции
$\boldsymbol{\alpha_1 = F}$ **(рабочий цех)**	У каждой операции своя машина $\lvert p_{ij} \rvert = 1$ и линейный порядок выполнения операций $O_{i1} \to O_{i2} \to \ldots \to O_{in_i}$
$\boldsymbol{\alpha_1 = O}$ **(потоковая линия)**	Машины упорядочены $M_1, M_2, \ldots, M_m$ и каждая работа проходит все машины в этом порядке $n_i = m$ и $\mu_{ij} = M_j, \forall i$
$\boldsymbol{\alpha_1 = J}$ **(открытая линия)**	Каждая работа состоит из m операций ($n_i = m$), но $\mu_{ij} = \{M_1, \ldots, M_m\}$ и на множестве операций нет условия предшествования
$\boldsymbol{\alpha_1 = X}$ **(смешанный цикл)**	Смесь J и O
$\boldsymbol{\alpha_1 = G}$ **(общий случай)**	Произвольный порядок предшествования на операциях (как в календарном планировании)
Если $\boldsymbol{\alpha_1} \in \{\boldsymbol{J}, \boldsymbol{F}, \boldsymbol{O}, \boldsymbol{X}, \boldsymbol{G}\}$,	То $n_i \geq 1$, т.е. у каждой работы может быть несколько операций

Таблица 2. Особенности задач составления расписания в зависимости от характеристик работ

Значения β	***Характеристики работ***
$\boldsymbol{\beta_1}$	Разрешаются прерывания
$\boldsymbol{\beta_2}$	Условия предшествования на множестве работ (цепи деревья, сети)
$\boldsymbol{\beta_3 = r_i}$	Время поступления на обслуживание
$\boldsymbol{\beta_4 \in \{p_{ij} = 1; p_{ij} \in \{0,1\}; p_{ij} = p_{ij}(t), \dots\}}$	Уточнение для времени выполнения операций
$\boldsymbol{\beta_5 = d_i}$	Директивные сроки окончания работ
$\boldsymbol{\beta_6}$	Работы разбиваются на группы, и в каждой группе берется максимум (сумма) времен выполнения работ

Таблица 3. Особенности задач составления расписания в зависимости от целевой функции

Целевая функция	***Формулировка целевой функции***
$\boldsymbol{C_{max} = \max_{i=1,\dots,n} c_i}$	Время окончания всех работ
$\boldsymbol{L_{max} = \max_{i=1,\dots,n} (c_i - d_i)}$	Запаздывание относительно директивных сроков
$\boldsymbol{D_{max} = \max_{i=1,\dots,n} \lvert c_i - d_i \rvert}$	Отклонение от директивных сроков
$\boldsymbol{F_{max} = \max_{i=1,\dots,n} (\max\{0, d_i - c_i\})}$	Опережение директивных сроков
$\boldsymbol{\sum_{i=1}^{n} w_i c_i}$	Взвешенная сумма окончания работ

Рассмотрим следующие практические примеры задач расписания:

- Загрузка оборудования на производстве.

Примером задачи загрузки оборудования на производстве может служить задача составления плана обработки партии деталей на заводском оборудовании.

Задача заключается в том, чтобы распределить временные ресурсы, затрачиваемые на выполнение работ соответствующими машинами.

Работами являются отдельные операции обработки деталей, машинами – оборудование, на котором выполняются эти операции. Критериями оптимальности плана могут быть общее время, необходимое для обработки всей партии деталей (окончания всех работ), потребляемая оборудованием электроэнергия и т. д.

Основываясь на приведенной классификации, машины могут быть организованы следующим образом: для каждой работы задана машина для ее выполнения, машины работают параллельно, рабочий цех (у каждой операции своя машина и линейный порядок выполнения операций), потоковая линия (машины упорядочены и каждая работа проходит все машины в этом порядке), открытая линия (каждая работа состоит операций, но и на множестве операций нет условия предшествования). Операции обработки деталей могут быть организованы следующим образом: разрешаются прерывания, заданы условия предшествования на множестве работ, задано время поступления на обслуживание, заданы времена выполнения операций, заданы директивные сроки окончания работ.

- Составление плана-графика работы и отдыха летных экипажей.

Такая задача возникает в процессе планирования летной работы авиакомпании [28].

Задача заключается в том, чтобы распределить временные ресурсы, затрачиваемые на выполнение работ соответствующими машинами.

Задача заключается в составлении плана-графика работы и отдыха летных экипажей авиационной эскадрильи на месяц на основании плана полетов. Работами здесь являются авиарейсы со всеми присущими им свойствами, а машинами – летные экипажи. Критериями оптимальности могут быть равномерность распределения летного и рабочего времени по всем экипажам эскадрильи, а также равномерность занятости каждого экипажа в течение месяца. В авиакомпании также решается аналогичная предыдущей задача, в которой вместо летных экипажей ресурсом являются самолеты.

- Расписание занятий учебных заведений.

Одним из наиболее распространенных примеров задачи составления расписаний является задача составления расписания занятий в учебном заведении [29].

Задача заключается в том, чтобы распределить временные ресурсы, затрачиваемые на выполнение работ соответствующими машинами.

В данном случае работами являются отдельные занятия, а роль машин могут играть аудитории или преподаватели. Критериями оптимальности могут служить равномерность учебной нагрузки у преподавателей, количество «окон» у преподавателей и учебных групп и т. д. Составленное расписание должно удовлетворять ряду ограничений, например, по рабочему времени преподавателей в течение недели, по загруженности учащихся и т. д.

Таким образом, на основе рассмотренных практических примеров задач расписания и некоторой классификации задач можно сделать следующие выводы:

- Теория изучает широкий круг задач, начиная с простейшей задачи выбора очередности выполнения одноэтапных работ одним исполнителем и заканчивая так называемой общей задачей, связанной с анализом многошаговых технологических процессов в системах конвейерного типа.
- Существуют разные формы представления расписаний, и довольно широк выбор терминов, определяющих одни и те же понятия.

2. Пространство состояний задачи расписания

Проанализировав ряд задач расписания можно выделить две составляющие:

- общая – особенности, присущие всем задачам;
- частная – особенности, отражающие предметную направленность.

Общим можно считать некоторое множество атрибутов расписания, из которых генерируется множество всевозможных перестановок (Таблица 4).

Таблица 4. Общая составляющая задач составления расписания

	Пространство состояния расписания	*Теоретико-множественное описание*
Учебное расписание	Состояние Преподавателей Состояние Студентов (групп) **Состояние Аудиторно-временного фонда** Состояние Предметов	Множество «Преподаватели» Множество «Студенты» Множество «Аудиторно-временной фонд» Множество «Предметы»
Расписание движения поездов	Состояние Станций отправления Состояние Станций прибытия Состояния Поездов Состояние Временных ресурсов Состояние Маршрутов из пунктов отправления в пункты прибытия	Множество «Станции отправления» Множество «Станции прибытия» Множество «Поезда» Множество «Временные ресурсы» Множество «Маршруты из пунктов отправления в пункты прибытия»
Расписание движения самолетов	Состояние Пунктов отправления Состояние Пунктов прибытия Состояние Самолетов **Состояние Временных ресурсов** Состояние Маршрутов (коридоров)	Множество «Пункты отправления» Множество «Пункты прибытия» Множество «Самолеты» Множество «Временные ресурсы» Множество «Маршруты»
Расписание загрузки оборудования	Состояние Рабочих мест Состояние Рабочих Состояние Технологических маршрутов **Состояние Временных ресурсов**	Множество «Рабочие места» Множество «Рабочие» Множество «Технологические маршруты» Множество «Временные ресурсы»

В качестве частной составляющей выступают структурные и семантические особенности прикладной задачи, особенности условий и ограничений, а также требования к результату, каким требованиям должно соответствовать полученное расписание.

Рассмотрим частную составляющую задачи построения расписания в учебном заведении [30].

Опишем переменные, используемые при формализации задачи. Сначала опишем индексы, от которых зависят переменные.

Все пары в течение недели последовательно пронумерованы (индекс p).

В течение одного дня у данной подгруппы может быть не более M_p пар (см. условие $A2$).

Заметим, что M_p может зависеть от дня недели.

Каждая группа разбита на 2 подгруппы (ПГ), и все ПГ пронумерованы (индекс g).

Соединение ПГ в группы, групп в потоки отображается с помощью матриц связности G и Р соответственно.

Все аудитории пронумерованы (индекс a).

Для каждой аудитории указывается ее вместимость V_a и тип θ_a: компьютерный класс – C, аудитория для практических или традиционных лекционных занятий – D, лекционная аудитория со средствами отображения – A, специальная лаборатория – S.

Все дисциплины пронумерованы (индекс d), где под дисциплиной понимается либо лекционный курс в потоке, либо курс практических занятий в группе, либо курс лабораторных занятий в подгруппе.

Для каждой дисциплины заданы общее количество часов T_{gd} (в -й подгруппе), какой преподаватель в какой подгруппе ведет эту дисциплину, тип τ_{dd}-й дисциплины (лекция L, лабораторное занятие C в компьютерном классе или S в специальной лаборатории, практическое занятие P).

Отметим, что как список дисциплин, так и количество часов по ним могут быть различными для четных и нечетных недель.

Все преподаватели пронумерованы (индекс l).

Пусть $L(d)$ – номер преподавателя, который ведет дисциплину d.

Введем переменные, используемые при формализации задачи.

Пусть $x_{pgad} = 1$, если занятие по -й дисциплине у g-й ПГ проводится в a-й аудитории на p-й паре, и $x_{pgad} = 0$ в противном случае.

Тогда можно записать следующие ограничения, вытекающие из общих соображений относительно учебного расписания.

$O1$. Занятие не может проводиться одновременно в двух разных аудиториях; т. е. если $1 \neq a2$, то для всех $p, g, a1, a2, a1 \neq a2$,

$$x_{pga_1d_1} \cdot x_{pga_2d_2}. = 0. \tag{1}$$

Условие (1) можно заменить на следующее:

$$\sum_a x_{pga_1d_1} \leq \rho_{gd}, \tag{2}$$

где ρ_{gd}=1, если -я ПГ изучает дисциплину d, $\rho_{gd} = 0$ в противном случае.

$O2$. Одновременно не могут проводиться занятия по двум разным дисциплинам в одной и той же ПГ, т. е. если $d1 \neq d2$, то

$$x_{pga_1d_1} \cdot x_{pga_2d_2}. = 0. \tag{3}$$

Условие можно заменить на следующее:

$$\sum_t x_{pgad} \leq \rho_{gd}. \tag{4}$$

O3. Одновременно не могут две ПГ иметь занятия в одной и той же аудитории, если они – не две ПГ одной группы на практическом занятии или ПГ, входящие в один поток на лекционном занятии. Это означает, что если $g1 \neq g2$, то при $d1 \neq d2$, либо при $d1 = d2 = d$ и $\tau = 0$

$$x_{pg_1ad_1} \cdot x_{pg_2ad_2}. = 0. \tag{5}$$

Здесь

$$\tau_{g_1g_2d} = \begin{cases} P_{g_1g_2} \cdot \rho_{g_1d}, \textit{если } \tau_d = L \\ G_{g_1g_2} \cdot \rho_{g_1d}, \textit{если } \tau_d = P \\ 0, \textit{если } \tau_d = C \textit{ или } \tau_d = S \end{cases}$$

или, иначе, $\tau_{g_1g_2d} = \rho_{g_1d}(P_{g_1g_2}\chi(\tau(d) = Л) + Gg_1g_2\chi(\tau(d) = П)$ – коэффициент, указывающий на принадлежность ПГ к одной группе или одному потоку, и эта ПГ (или этот поток) должна (должны) изучить дисциплину d ($\rho_{g_1d} = 1$ в случае ПГ и $G_{g_1g_2} = 1$ в случае потока); здесь $\chi(A) = 1$, если условие A выполняется, и $\chi(A) = 0$ в противном случае, $P_{g_1g_2}$ и $G_{g_1g_2}$ – соответствующие элементы матриц P и G соответственно. Условие $O3$ можно переформулировать иначе: именно в данной аудитории могут находиться только те ПГ, у которых на данной паре проводятся занятия по данной дисциплине. Это позволяет формализовать условие $O3$ также и следующим образом: для всех p, a, d.

Данные, связанные с учебным процессом:

- g – число всех подгрупп;
- матрица связности G подгрупп и P – групп: $G_{g_1g_2} = 1$, $P_{g_1g_2} = 1$, если ПГ g_1 и g_2 – в одной группе или в одном потоке соответственно, и $G_{g_1g_2} = 0$, $P_{g_1g_2} = 0$ в противном случае;
- R_g – число студентов в -й ПГ;
- D – число всех дисциплин;
- вектор $\{L(d)\}$, где $L(d)$ – номер преподавателя, ведущего -ю дисциплину;
- T_{gd} – число пар по -й дисциплине в g-й ПГ в течение недели в соответствии с учебным расписанием ПГ;

- тип τ_d -й дисциплины (L – лекция, P – практическое занятие, C – лабораторное занятие в компьютерном классе, S – лабораторное занятие в специализированной лаборатории);
- матрица $\rho = \{\rho_{gd}\}$, где $\rho_{gd} = 1$, если -я подгруппа изучает d-ю дисциплину, и $\rho_{gd} = 0$ в противном случае (т. е. $\rho_{gd} = \chi(T_{gd} > 0)$);
- для всех ПГ значения δ_g – суммарное число пар в неделю у -й ПГ;
- для всех дней недели значения g_u – доля u-х дней недели в семестре, когда занятия не будут проводиться (в связи с праздниками или по другим причинам);
- список K_1 всех ПГ первого курса;
- tp – длительность одной пары, $t\Pi$ – длительность перемены;
- $\overline{T_{gd}^{Л}}, \overline{T_{gd}^{П}}, \overline{T_{gd}^{Лаб}}$ – общее число часов лекционных, практических и лабораторных занятий по d-й дисциплине в g-й подгруппе;
- D_n – множество дисциплин, относящихся к дипломным или курсовым работам;
- Ф – множество дисциплин по физкультуре;
- g_u – доля непраздничных дней, приходящихся в течение семестра на -й день недели.

Данные, связанные с возможностями учебного заведения:

- M_p – максимальное число учебных пар в день;
- K – общее число учебных аудиторий;
- по всем учебным аудиториям a: V_a – вместимость a-й аудитории;
- θ – тип a-й аудитории (D – лекционная аудитория с учебной доской, A – лекционная аудитория со средствами отображения, C – компьютерный класс, S – специализированная лаборатория);
- $\Delta(a_1, a_2)$ – условное расстояние между a_1-й и a_2-й аудиториями;
- L – число всех преподавателей;
- B – множество всех аудиторий вне ВУЗа;
- $T_Н^a$ и $T_З^a$ моменты начала и окончания занятий в аудитории $a \in B$ и время ω_a, необходимое, чтобы добраться до аудитории $a \in B$ из ВУЗа или возвратиться обратно в ВУЗ;
- $v_Л^к$ и $v_П^к$ (человек) – объемы зарезервированных в k-м корпусе аудиторий для лекционных и для практических занятий;
- κ_{max} – число всех учебных корпусов;
- $C_a^Л$, $C_a^П$ – число и $C_{k,Л}$, $C_{k,П}$ – списки всех лекционных аудиторий и аудиторий для практических занятий в k-м учебном корпусе вместимостью $v_П^a$, $v_Л^a$ человек соответственно.

Данные, получаемые на основе субъективных оценок и методик:

- степень приоритетности p_{ad} -й аудитории для чтения d-й дисциплины при $0 \leq\leq 1$ и степень не p_{ad} желательности проведения занятия по d-й дисциплине в a-й аудитории;
- по преподавателям: b_{lu} и c_{lu} – номер пары, с которой -й преподаватель хотел бы начинать занятия и до которой хотел бы их закончить в u-й день недели; f_{lu}– наименьшее количество пар, которые хотел бы иметь l-й преподаватель в u-й день недели;
- набор чисел $\{\mu_u\}$ степеней дифференциации нагрузки по дням недели;
- π_{pal} – уровень приоритетности при $\pi_{pal} > 0$ или неприоритетности при $\pi_{pal} < 0$ или проведения u-ого занятия l-м преподавателем в a-й аудитории;
- α – доля в процентах лекционных занятий в дневном расписании студентов (при числе пар больше двух);
- β – доля в процентах числа занятий по одной дисциплине (при числе пар больше двух);
- номер r пары, после которой у первого курса не должно быть занятий;
- $\gamma_П, \gamma_{Лаб,S}, \gamma_{Лаб,C}$ – соответственно максимально допустимые доли практических, специальных лабораторных и компьютерных занятий в течение учебного дня в случае, если число пар больше двух;

- k_u – коэффициент, учитывающий желательность того, чтобы свободным днем был именно -й день недели;
- $C(d,p)$ – степень нежелательности проведения занятий по -й дисциплине на p-й паре учебного дня;
- $\delta(p,d)$ – уровень желательности проведения -й дисциплины на p-й паре;
- J – множество ПГ, для которых предпочтительно иметь один и тот же день недели для самостоятельной работы;
- μ_u $(u=\overline{1,6})$ – коэффициенты, учитывающие желательную степень дифференциации нагрузки по дням недели.

Всю совокупность требований можно условно разбить на следующие 4 категории (A, B, C, H), имеющие разный уровень приоритетности [30].

A. Имеют абсолютный приоритет, т. е. учитываются всегда и не могут быть нарушены ни при каких обстоятельствах.

B. Имеют высокий приоритет и могут быть нарушены в единичных случаях при крайней необходимости.

C. Имеют средний приоритет, могут быть нарушены, хотя их нарушение крайне нежелательно.

H. Имеют низкий приоритет, их нарушение не влечет за собой никаких серьезных последствий, однако желательно, по возможности, соблюдать эти требования.

Отметим, что приведенная градация требований достаточно условна и, как будет ясно из дальнейшего, возможность каждого конкретного требования может быть учтена на основе весовых коэффициентов, учитывающих важность требования.

Перечислим требования каждой категории.

Ограничения категории А

$A1$. В дневном расписании каждой группы/подгруппы (ПГ) студентов не должно быть «окон».

$A2$. В течение дня в каждой ПГ не должно быть более 5-ти пар занятий (для первого курса – не более 4-х пар).

$A3$. Если в дневном расписании число пар больше 2-х, то не должны проходить занятия только по одной дисциплине.

$A4$. Если в дневном расписании число пар больше 2-х, то не должны проходить только лекционные занятия.

$A5$. Занятия первого курса должны кончаться в первой половине дня.

$A6$. Некоторые занятия могут проходить в других организациях вне ВУЗа (выездные занятия). В связи с этим необходимо учесть требования этих организаций (по месту, времени проведения, количеству студентов, участвующих в каждом занятии, и др.).

$A7$. Для каждой учебной аудитории необходимо учесть приоритетность ее выделения для каждой дисциплины (специальности, группы). Это, прежде всего, касается специальных и профилированных лабораторий.

$A8$. Необходимо учесть ограничения на учебные расписания конкретных должностных лиц, внешних совместителей и других выделенных сотрудников. В частности, члены ученого совета университета (ученых советов институтов, деканатов) не должны иметь учебных занятий в промежутки времени, когда предусмотрены заседания соответствующих советов. Ректору, проректорам и другим членам руководства ВУЗа может быть удобно (или неудобно) проведение занятий в определенные промежутки времени, в определенных аудиториях.

$A9$. В каждом учебном корпусе на каждой паре необходимо зарезервировать одну свободную аудиторию, вмещающую одну стандартную группу.

Ограничения категории B

$B1$. В течение дня число учебных пар не должно превышать 4-х.

$B2$. Занятия на первом курсе должны начинаться с первой пары.

$B3$. Лекционные занятия должны предшествовать практическим и лабораторным занятиям.

$B4$. Необходимо в максимальной степени минимизировать перемещения студентов из одних помещений в другие, в особенности из одних учебных корпусов в другие.

$B5$. В течение учебного дня в случае, когда число пар больше 2-х, не должны проходить занятия только одного типа: лабораторные занятия в компьютерном классе, либо лабораторные занятия в специальной лаборатории, либо практические занятия, за исключением случаев, когда проведение занятия связано с внешней организацией или с внешними совместителями.

$B6$. Необходимо обеспечить наличие дня самостоятельной работы студентов для тех ПГ, у которых на данный семестр запланированы дипломные и курсовые работы.

Ограничения категории C

$C1$. Не ставить два занятия по физкультуре в одной подгруппе в течение дня подряд. Желательно, чтобы в течение учебного дня было не более одной пары по физкультуре и занятия на открытом воздухе проходили не позже определенного часа, что особенно важно в зимнее время года, когда рано темнеет.

$C2$. Каждый преподаватель в течение дня должен иметь не более 5-ти пар.

$C3$. Каждый преподаватель в течение дня не должен иметь более 3-х лекций подряд.

$C4$. В расписании преподавателей суммарная длина «окон» должна быть минимальной; в идеальном варианте в расписании преподавателей «окон» не должно быть вообще.

$C5$. Необходимо назначать занятия преподавателей, по возможности, в близкие аудитории с целью минимизировать их перемещения.

Ограничения категории H

$H1$. Занятия по физкультуре должны быть на последних парах дневного расписания ПГ.

$H2$. Занятия по военной кафедре должны начинаться как можно раньше.

$H3$. Занятия по некоторым дисциплинам (например, по математике) должны проводиться на первых парах дневных расписаний ПГ.

$H4$. Желательно обеспечить наличие свободного дня для самостоятельной работы студентов для всех ПГ из заданной их совокупности (группы, специальности, потока и т. п.).

$H5$. Необходимо учесть индивидуальные пожелания отдельных преподавателей, связанные со временем начала и (или) окончания занятий у них в отдельные дни недели, числом пар в течение дня и т. п.

$H6$. Для некоторых преподавателей необходимо обеспечить максимальное отличие или совпадение их расписаний с расписаниями некоторых других преподавателей. Данное требование связано, в частности, с возможностью обеспечить временные замены одних дисциплин другими либо совместить дисциплины ввиду предполагаемого отсутствия преподавателя, ведущего одну из этих дисциплин, в течение длительного времени; например, в связи с намеченными поездками либо относительно частым пропуском занятия по уважительным причинам (болезнью, отпуском, командировками и т. п.).

$H7$. Для некоторых дисциплин желательно, чтобы они проходили одновременно; например, при наличии нескольких специальных дисциплин, читаемых по выбору студентов или выбираемых студентами.

$H8$. Необходимо минимизировать потери учебных занятий, связанные с праздничными и другими мероприятиями, в результате которых занятия не проводятся.

$H9$. Средняя дневная нагрузка по всем дням недели (за исключением, возможно, субботы) приблизительно должна быть одинаковой, т. е. колебания нагрузки по всем дням недели должны быть минимальными.

Таким образом, на основе анализа пространства состояний задач расписания из различных прикладных областей сделаны следующие выводы:

В задачах составления можно выделить две составляющие: общую, характеризующая особенности, присущие всем задачам (атрибуты или компоненты расписания), и частную – особенности, отражающие предметную направленность (ограничения, требования к результату).

Трудности получения и практического использования научных результатов в значительной степени обусловлены разнообразием ограничений, встречающихся в конкретных ситуациях, что приводит к неизбежной идеализации исследуемых систем.

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.

3. Методы решения задачи расписания

Методы решения задач расписания разделяются на три класса:

- Точные методы (в основе лежит метод полного перебора);
- Приближенные методы;
- Методы с гарантированной точностью.

Точные методы

К точным методам решения относятся методы линейного программирования, как правило, данные о задаче не так точны, как предполагается методом, что в значительной степени затрудняет его использование. Однако методы линейного программирования имеют целый ряд интересных для практического использования приложений.

Также к точным методам решения задачи расписания относятся алгоритмы целочисленного программирования Гомори, в котором предполагается, что все коэффициенты в выражениях и целевой функции являются целыми числами. Если учесть неделимость отдельных видов работ и ресурсов, комбинаторный характер операций, из которых состоит расписание, то такой метод имеет определенные возможности [31].

Метод отсечения

Алгоритмы методов отсечения разработаны для решения полностью или частично целочисленных и дискретных задач линейного программирования. Общая схема методов отсечения заключается в переходе от решения задачи целочисленного (дискретного) линейного программирования (ЦЛП) – $\Omega^{ц}F$– задачи» к решению последовательности задач линейного программирования (ЛП) – «$\Omega_k F$ – задач», $k = 0, 1, 2, ...$

$\Omega_k F$ – задача получается из $\Omega^{ц}F$ – задачи снятием требования целочисленности (дискретности).

Каждая последующая k-ая задача получается из предыдущей $(k-1)$-ой путем добавления к условиям, определяющим область допустимых решений $(k-1)$-ой задачи, еще одного ограничения, называемого **правильным отсечением**.

После решения каждой k-ой задачи ЛП ($k = 0,1,2,...$) проверятся, удовлетворяет ли полученное оптимальное решение условиям исходной $\Omega^{ц}F$ – задачи. При положительном ответе итерационный процесс решения – задач прекращается – полученное оптимальное решение задачи ЛП является и решением исходной задачи.

Решение каждой k-ой задачи ЛП называется k-ой большой итерацией. Доказывается, что при выполнении определенных условий решение исходной задачи будет получено через конечное число итераций. Если задачи ЛП не имеют решения, не имеет решения и исходная задача. Блок-схема алгоритма показана на рис. 2.

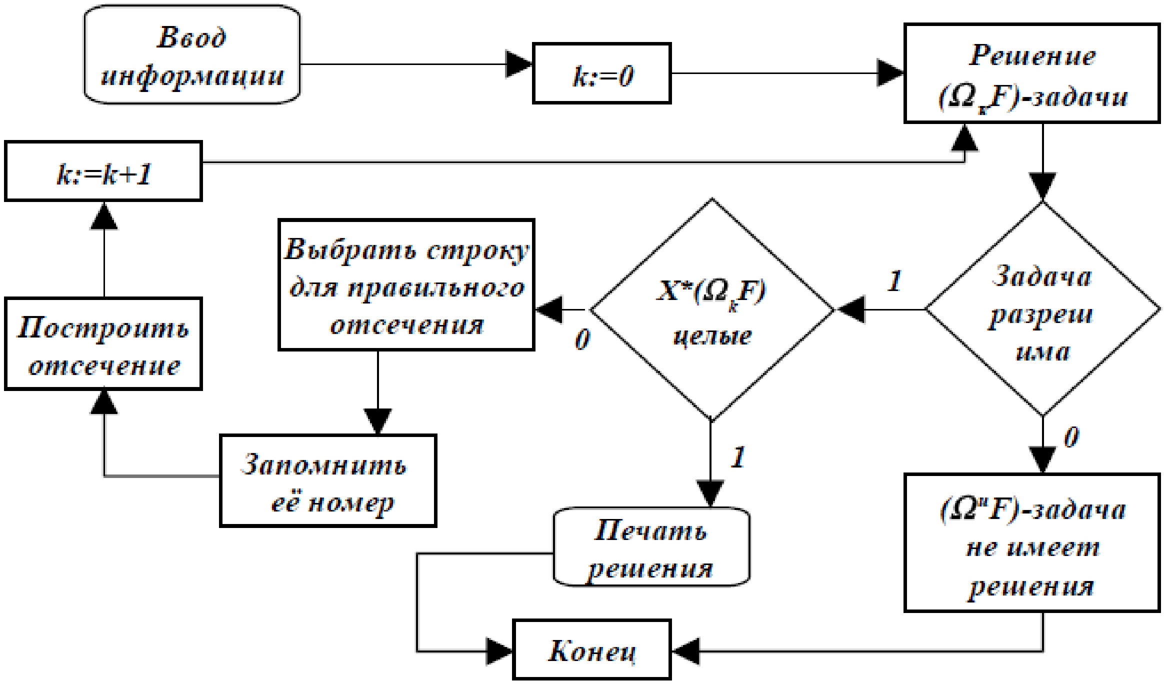

Рис. 2. Алгоритм метода отсечения

Дополнительные ограничения – правильные отсечения – должны:

- быть линейными;
- являться «отсечением», т. е. отсекать от области допустимых решений ту ее часть, которая содержит оптимальное решение задачи ЛП, не удовлетворяющее требованиям исходной задачи;
- быть «правильными», т. е. в отсекаемой области не должно содержаться ни одной точки, принадлежащей области допустимых решений исходной задачи.

Известны три алгоритма отсечения:

- 1-ый алгоритм Гомори решения целочисленной задачи ЛП.
- 2-ой алгоритм Гомори решения частично целочисленной задачи ЛП.
- Алгоритм Дальтона-Ллевелина решения дискретной задачи ЛП.

Общая формула построения правильного отсечения для всех алгоритмов запишется в следующем виде:

$$z = -\gamma_0^k - \sum_{j \epsilon \text{НБ}} (-\gamma_j^k)\, x_j, z \geq 0,$$

где x_j – небазисные переменные оптимального решения -ой задачи ЛП – «$\Omega_k F$ – задачи», γ_0, γ_j – определяются по коэффициентам разложения базисной переменной, не удовлетворяющей требованиям целочисленности (дискретности) исходной задачи по небазисным переменным в последней симплекс-таблице $\Omega_k F$ – задачи ЛП (строка симплекс-таблицы с найденным оптимальным значением задачи ЛП) [32].

Для решения задач расписания часто применяют метод ветвей и границ.

Метод ветвей и границ

Метод ветвей и границ – один из комбинаторных методов. Его суть заключается в упорядоченном переборе вариантов и рассмотрении лишь тех из них, которые оказываются по определенным признакам перспективными, и отбрасывании бесперспективных вариантов.

Метод ветвей и границ состоит в следующем: множество допустимых решений (планов) некоторым способом разбивается на подмножества, каждое из которых этим же способом снова разбивается на подмножества. Процесс продолжается до тех пор, пока не получено оптимальное целочисленное решение исходной задачи.

Алгоритм решения:

Первоначально находим симплексным методом или методом искусственного базиса оптимальный план задачи без учета целочисленности переменных. Пусть им является план X_0. Если среди компонент этого плана нет дробных чисел, то тем самым найдено искомое решение данной задачи и $F_{max} = F\ (X_0)$.

Если же среди компонент плана X_0 имеются дробные числа, то X_0 не удовлетворяет условию целочисленности и необходимо осуществить упорядоченный переход к новым планам, пока не будет найдено решение задачи. Покажем, как это можно сделать, предварительно отметив, что $F\ (X_0) \geq F\ (X)$ для всякого последующего плана X_0.

Предполагая, что найденный оптимальный план X_0 не удовлетворяет условию целочисленности переменных, тем самым считаем, что среди его компонент есть дробные числа. Пусть, например, переменная x_{i_0} приняла в плане X_0 дробное значение. Тогда в оптимальном целочисленном плане ее значение будет, по крайней мере, либо меньше или равно ближайшему меньшему целому числу K_{i_0}, либо больше или равно ближайшему большему целому числу $K_{i_0} + 1$. Определяя эти числа, находим симплексным методом решение двух задач линейного программирования:

$$\begin{cases} F = \sum_{j=1}^{n} c_j x_j \to max, \\ \sum_{j=1}^{n} a_{ij} x_j = b_i \ (i = \overline{1,m}), \\ x_{i0} \leq K_{i0}, \\ x_j \geq 0 \ (j = \overline{1,n}); \end{cases} \qquad \text{I}$$

$$\begin{cases} F = \sum_{j=1}^{n} c_j x_j \to max, \\ \sum_{j=1}^{n} a_{ij} x_j = b_i \ (i = \overline{1,m}), \\ x_{i0} \leq K_{i0} + 1, \\ x_j \geq 0 \ (j = \overline{1,n}); \end{cases} \qquad \text{II}$$

Найдем решение задач линейного программирования (I) и (II). Очевидно, здесь возможен один из следующих четырех случаев:

- Одна из задач неразрешима, а другая имеет целочисленный оптимальный план. Тогда этот план и значение целевой функции на нем и дают решение исходной задачи.
- Одна из задач неразрешима, а другая имеет оптимальный план, среди компонент которого есть дробные числа. Тогда рассматриваем вторую задачу и в ее оптимальном плане выбираем одну из компонент, значение которой равно дробному числу, и строим две задачи, аналогичные задачам (I) и (II).
- Обе задачи разрешимы. Одна из задач имеет оптимальный целочисленный план, а в оптимальном плане другой задачи есть дробные числа. Тогда вычисляем значения целевой функции на этих планах и сравниваем их между собой. Если на целочисленном оптимальном плане значение целевой функции больше или равно ее значению на плане, среди компонент которого есть дробные числа, то данный целочисленный план является оптимальным для исходной задачи, и он вместе со значением целевой функции на нем дает искомое решение.

- Если же значение целевой функции больше на плане, среди компонент которого есть дробные числа, то следует взять одно из таких чисел и для задачи, план которой рассматривается, необходимо построить две задачи, аналогичные (I) и (II).
- Обе задачи разрешимы, и среди оптимальных планов обеих задач есть дробные числа. Тогда вычисляем значение целевой функции на данных оптимальных планах и рассматриваем ту из задач, для которой значение целевой функции является наибольшим. В оптимальном плане этой задачи выбираем одну из компонент, значение которой является дробным числом, и строим две задачи, аналогичные (I) и (II).

Таким образом, описанный выше итерационный процесс может быть представлен в виде некоторого дерева, на котором исходная вершина отвечает оптимальному плану X_0 задачи (1) – (3), а каждая соединенная с ней ветвью вершина отвечает оптимальным планам задач (I) и (II). Каждая из этих вершин имеет свои ветвления. При этом на каждом шаге выбирается та вершина, для которой значение функции является наибольшим. Если на некотором шаге будет получен план, имеющий целочисленные компоненты, и значение функции на нем окажется больше или равно, чем значение функции в других возможных для ветвления вершинах, то данный план является оптимальным планом исходной задачи целочисленного программирования и значение целевой функции на нем является максимальным.

Итак, процесс нахождения решения задачи целочисленного программирования (1) – (4) методом ветвей и границ включает следующие основные этапы:

Находят решение задачи линейного программирования (1) – (3).

Составляют дополнительные ограничения для одной из переменных, значение которой в оптимальном плане задачи (1) – (3) является дробным числом.

Находят решение задач (I) и (II), которые получаются из задачи (1) – (3) в результате присоединения дополнительных ограничений.

В случае необходимости составляют дополнительные ограничения для переменной, значение которой является дробным, формулируют задачи, аналогичные задачам (I) и (II), и находят их решение. Итерационный процесс продолжают до тех пор, пока не будет найдена вершина, соответствующая целочисленному плану задачи (1) – (3) и такая, что значение функции в этой вершине больше или равно значению функции в других возможных для ветвления вершинах.

Описанный выше метод ветвей и границ имеет более простую логическую схему расчетов, чем метод Гомори. Поэтому в большинстве случаев для нахождения решения конкретных задач целочисленного программирования применяется именно этот метод [33, 34].

Все точные методы включают в том или ином виде перебор. Полный перебор – метод решения задачи путем перебора всех возможных вариантов. Сложность полного перебора зависит от количества всех возможных решений задачи. Если пространство решений очень велико, то полный перебор может не дать результатов в течение нескольких лет или даже столетий, поэтому важно, используя данный метод, осуществить перебор эффективно, используя как можно больше информации для сокращения дерева поиска.

Приближенные методы

Приближенные алгоритмы – это любые схемы, которые могут привести к получению допустимого решения с низкой стоимостью.

Неудовлетворительное состояние развития точных методов решения задач теории расписаний обусловило разработку приближенных методов, позволяющих получать приемлемые решения при сравнительно небольших затратах времени и средств. Условно приближенные методы делятся на эвристические и вероятностные.

Эвристические алгоритмы основаны на приеме, который называется приемом снижения требований. Он заключается в отказе от поиска оптимального решения за приемлемое время. Эвристические алгоритмы используют различные разумные соображения без строгих обоснований, поэтому довольно сложно сделать обобщение эвристических методов. Сталкиваясь с конкретной задачей, человек может обдумывать способ получения решения, обладающего невысокой стоимостью, очень мало заботясь при этом о теоретическом обосновании. Однако эвристические процедуры при всей своей произвольности и слабости могут быть с успехом применены для решения задач высокой вычислительной сложности (задачи, принадлежащие классу NP). Например, в методе ветвей и границ эвристические алгоритмы применяются как средство получения начальных решений и верхних оценок.

Под «эффективным алгоритмом» понимается алгоритм, для которого число требуемых шагов растет как полином от размера входной задачи. Задачи, имеющие эффективные (полиномиальные) алгоритмы решения, принадлежат классу *P*-задач.

Класс *NP*-полных задач обладает следующими свойствами:

- никакую *NP*-полную задачу нельзя решить никакими известными полиномиальными алгоритмами;
- если существует полиномиальный алгоритм для какой-нибудь *NP*-полной задачи, то существуют полиномиальные алгоритмы для всех *NP*-полных задач.

Практическое значение понятия *NP*-полноты состоит в следующем: такие задачи по существу труднорешаемы с вычислительной точки зрения, они не поддаются эффективному алгоритмическому решению и для алгоритма, корректно решающего *NP*-полную задачу, потребуется в худшем случае экспоненциальное количество времени и, следовательно, он не будет применим на практике ни к каким, за исключением очень малых, задачам.

В качестве примера рассмотрим *NP*-полную задачу Джонсона, свойства оптимальных расписаний которой подробно исследуются в статье Севастьянова С. В., Чемисовой Д. А., Черных Д. А. «О некоторых свойствах оптимальных расписаний в задаче Джонсона с прерываниями» [35].

Пример 1. Задача Джонсона (полиномиально разрешимая задача). Имеем конвейер из двух машин: $m = 2$. Каждая работа состоит из двух операций с длительностями a_i и b_i, минимизируется общее время обслуживания работ.

Последовательность, минимизирующая общее время работы, такова: сначала запускаются работы, для которых $a_i \leq b_i$ в порядке неубывания a_i, затем все работы, для которых $a_i < b_i$ в порядке невозрастания b_i (тем самым минимизируем простои 2-ой машины из-за того, что 1-ая еще не успела обработать какую-либо работу). Доказана оптимальность такой последовательности. Однако на случай $n > 2$ результаты не распространяются.

NP-полнота задачи является веским доводом при обосновании необходимости построения приближенных или эвристических алгоритмов ее решения, применения схем направленного перебора вариантов (таких, как метод последовательного конструирования, анализа и отсеивания вариантов (обобщение метода ветвей и границ)), а также при обосновании необходимости исследования частных случаев задачи.

Павловым А. А. и др. предложен следующий путь решения *NP*-полных задач:

- исследуется труднорешаемая задача, и находятся теоретические свойства, которым удовлетворяет ее оптимальное решение;
- на основе этих свойств разрабатывается полиномиальный алгоритм решения, при этом вводится понятие полиномиальной разрешимости задач из класса *NP*, под которым понимается существование полиномиального алгоритма, удовлетворяющего следующим условиям:
- если при решении произвольной индивидуальной задачи выполняются определенные аналитические условия, то эта индивидуальная задача решается строго (т.е. получено строго оптимальное решение);
- в результате работы полиномиального алгоритма всегда известно решена или нет данная индивидуальная задача точно;
- полиномиальный алгоритм является эффективным и статически значимым, т.е. при моделировании произвольных индивидуальных задач в большинстве случаев полученные решения являются точными.

Несмотря на кажущуюся уязвимость приближенных методов, существует несколько методов, которые представляются эффективными для широкого круга задач. Одним из них является, поиск в локальной окрестности, другим – метод ветвей и границ без возвратов.

При использовании метода ветвей и границ без возвратов, можно специально сократить объем используемой памяти, при условии, что конечный результат не обязательно должен быть точным. Поскольку размеры множеств вершин заранее ограничены, то лишняя информация просто отбрасывается, но обязательно хранится наименьшая нижняя оценка, связанная с отброшенными вершинами, что позволяет получить нижнюю оценку решения в момент завершения. Такая нижняя оценка совместно с верхней оценкой, вычисленной для лучшего полного решения, используется для обеспечения точности решения. Другим ограничивающим фактором может стать время вычислений. Если в ситуации, когда превышается время счета, мы имеем полное решение, то оно, совместно с наименьшей нижней оценкой для вершин из активного множества, определяет границы точности нашего решения.

Широко применяется так называемый метод локального поиска. При этом заранее выбранное множество перестановок используется для последовательного улучшения начального решения до тех пор, пока такое улучшение возможно, в противном случае оказывается достигнутым локальный оптимум.

Применение персонального компьютера для решения задач расписания позволили привлечь методы моделирования, в которых используются многие приемы (локальные правила), применяющиеся при ручном способе составления расписаний. Локальное правило – это такое правило, которое используется, например, рабочим-станочником при выполнении запланированных работ и требует от него только знания работ, ожидающих своей очереди перед его станком. Известно множество локальных правил, каждое из которых дает расписание, близкое к оптимальному для конкретной модели календарного планирования. Предпочтительность того или иного правила в зависимости от размерности задачи или от последовательности выполнения работ определить трудно. Во всех случаях определяется лишь расписание, близкое к условно оптимальному, т. е. оптимальному в условиях применения определенных локальных правил.

Для каждого -ого задания из множества ожидающих выполнения заданий, вычисляется значение функции f_i предпочтения и выбирается то задание, для которого f_i достигает максимума или минимума.

Примеры правил предпочтения:

Правило *SPT* (*shirtest processing time*). Предпочтение отдается той работе (операции) из множества готовых к обработке на освободившейся машине, у которой время выполнения на этой машине минимально.

Правило *LRT* (*longest remaining time*). Требует выбора напряженной работы, т. е. той, у которой сумма времен выполнения оставшихся операций наибольшее.

Правило *LPT* (*longest processing time*). Предпочтение отдается той работе (операции) из множества готовых к обработке на освободившейся машине, у которой время выполнения на этой машине максимально.

Достоинством эвристических методов является удобство реализации их на ЭВМ даже при решении громоздких задач.

Недостатки эвристических методов заключаются в сложности оценки близости полученных расписаний к оптимальному. Кроме того для каждой функции предпочтения существуют задачи, для которых применение данной функции приводит к плохим результатам. Один из путей совершенствования метода функций предпочтения состоит в их привязке к классам задач.

Еще одной разновидностью методов решения задач составления расписаний являются вероятностные методы. Вероятностные методы связаны с -кратным моделированием расписаний. Выбор работ из множества ожидающих выполнения осуществляется случайным образом. После -кратного проигрывания выбирается наилучшее расписание, которое принимается за решение задачи. При этом различают:

- ненаправленный случайный поиск;

- направленный случайный поиск без самообучения;
- направленный случайный поиск с самообучением.

В основе эвристических методов, как правило, лежит использование различного рода эвристик или эвристических алгоритмов, при разработке которых используются интуитивные предположения, не подкрепленные соответствующим математическим обоснованием. Формирование расписания с помощью некоторых правил (эвристик) позволяет несколько ускорить поиск «наилучшего» расписания, но использование таких алгоритмов в большинстве случаев гарантирует лишь нахождение приближенного решения (достижение локального экстремума). В этом случае возникает проблема оценки близости найденного локального экстремума к глобальному экстремуму.

В ряде работ эта проблема решается путем сравнения расписания, полученного эвристическим методом и расписания, полученного методом перебора для близкой задачи малой размерности. Несмотря на указанный недостаток, эвристические алгоритмы продолжают оставаться достаточно эффективным инструментом поиска «лучшего» в некотором смысле решения в тех случаях, когда нахождение наилучшего решения крайне затруднено или невозможно[36].

Методы, основанные на ограничениях

Задача составления расписания сводится к задаче удовлетворения ограничений. Процесс составления расписания представляется как процесс распределения времени и места между занятиями таким образом, чтобы выполнялось множество ограничений. Обычно для этого задается множество правил.

Данный метод предполагает «жесткие» (*hard*) и «мягкие» (*soft*) ограничения. Выполнение «жестких» ограничений обязательно, а «мягких» желательно. На каждом шаге расставляется одна группа или курс. Если на каком-то шаге невозможно назначить время или аудиторию таким образом, чтобы не нарушились ограничения, то ослабляется одно из «мягких» ограничений. И так повторяется до тех пор, пока не станет возможным назначить время или аудиторию без нарушений оставшихся ограничений.

Недостатком этого метода является то, что не всегда можно расставить группу или курс и поэтому это приходится делать вручную.

Основным плюсом данного метода является сокращение пространства поиска.

В рамках данного подхода решение оптимизационной задачи осуществляется с помощью нейронных сетей (НС), где каждой целочисленной переменной x_{ij} решаемой задачи ставится в соответствие выходной сигнал -го нейрона V_{ij}, стоящего в i-й строке и j-м столбце матрицы, т. е. строится отображение. Далее с учетом полученного отображения интерпретируются ограничения и целевая функция и строится энергетическая функция НС.

На следующем этапе определяются параметры НС: матрица синаптических связей и вектор внешних смещений. В конечном итоге, поиск решения задачи с помощью данного метода сводится к реализации динамического процесса перехода НС из некоторого начального состояния в некоторое конечное состояние, которое и принимается за решение задачи.

Недостатком данного метода является сложность выбора начального состояния НС.

Поэтому при наличии достаточного временного ресурса на поиск решения следует реализовывать многократный переходный процесс сети и в качестве решения принимать наилучший из результатов.

Алгоритм имитации отжига

Одним из распространенных эвристических методов, который применяется для составления расписания занятий, является метод имитации отжига [38].

Алгоритм имитации отжига (анг. *Simulated annealing*) основывается на имитации физического процесса. Этот процесс происходит при кристаллизации вещества из жидкого состояния в твердое. Предполагается, что существуют допустимые переходы отдельных атомов из одной ячейки в другую, даже когда атомы уже выстроились в кристаллическую решетку. Этот процесс протекает при постепенно понижающейся температуре. Атом переходит из одной ячейки в другую с некоторой вероятностью, причем вероятность уменьшается с понижением температуры. Устойчивая кристаллическая решетка соответствует минимуму энергии атомов, поэтому атом либо переходит в состояние с меньшим уровнем энергии, либо остается на месте.

Данный алгоритм для задачи составления расписания можно представить следующим образом:

- На первой итерации алгоритм генерирует некоторое начальное корректное решение Z_0 (оно считается текущим $Z = Z_0$) и для этого решения задается начальное высокое значение контрольного параметра температура T_0.
- Далее происходит мутация расписания: может изменяться время проведения занятия или аудитория или перестановка занятий в расписании. В итоге генерируется новое корректное расписание Z'.
- Вычисляется целевая функция $\Delta f = f(Z') - f(Z)$. Если новое решение лучше предыдущего ($\Delta f \leq 0$), то оно его заменяет. Если же оно хуже ($\Delta f > 0$), то вероятность того, что оно заменит предыдущее, определяется формулой $p = e - \Delta f / T$.
- Происходит изменение текущей температуры. Температура, а вместе с ней и вероятность замены старого решения новым уменьшается с каждой итерацией (или через несколько итераций).
- Пока не выполнен критерий остановки переход к пункту 2. В качестве критерия остановки можно использоваться заданное число итераций без улучшения значения целевой функции.

Такой метод эффективен для составления небольших расписаний.

Генетические алгоритмы

Интенсивно развиваются в последнее время методы решения большеразмерных задач целочисленного программирования, объединенных термином «генетические алгоритмы».

Основные отличия и преимущества генетических алгоритмов в сравнении с классическими методами заключаются в следующем:

- генетический алгоритм (ГА) работает с кодами, в которых представлен набор параметров, напрямую зависящих от аргументов целевой функции;
- в процессе поиска ГА использует несколько точек поискового пространства (процесс распараллеливается), а не переходит от точки к точке, как это происходит в традиционных методах, т. е. ГА оперирует со всей совокупностью допустимых решений;
- ГА в процессе работы не использует дополнительной информации, что повышает скорость его работы;
- ГА использует как вероятностные правила для порождения новых точек поиска, так и детерминированные правила для перехода от одних точек к другим и др.

Генетический алгоритм состоит из следующих компонент:

- Хромосома. В качестве хромосомы выступает решение рассматриваемой задачи. Хромосома состоит из совокупности генов – параметров.
- Начальная популяция хромосом.
- Набор операторов для генерации нового поколения. Генетическими операторами являются оператор кроссовера (crossover operator) и оператор мутации (mutation operator). За счет кроссовера производится обмен генами между особями, то есть процесс скрещивания особей. Пусть имеются две родительские особи с хромосомами $X = \{x_i, i = 1, N\}$ и $Y = \{y_i, i = 1, N\}$.
- Случайным образом определяется точка разрыва (crossover point), внутри хромосомы, в которой обе хромосомы делятся на две части и обмениваются ими.
- Оператор мутации инвертирует случайно выбранный бит (ген) в хромосоме.
- Целевая функция для оценки приспособленности (fitness) решения.

ГА – это итерационный процесс, который продолжается до тех пор, пока не выполнятся критерий останова (например, заданное число поколений). На каждом поколении ГА реализуется отбор (селекция), кроссовер и мутация.

Данный алгоритм можно представить следующей схемой (рис. 3).

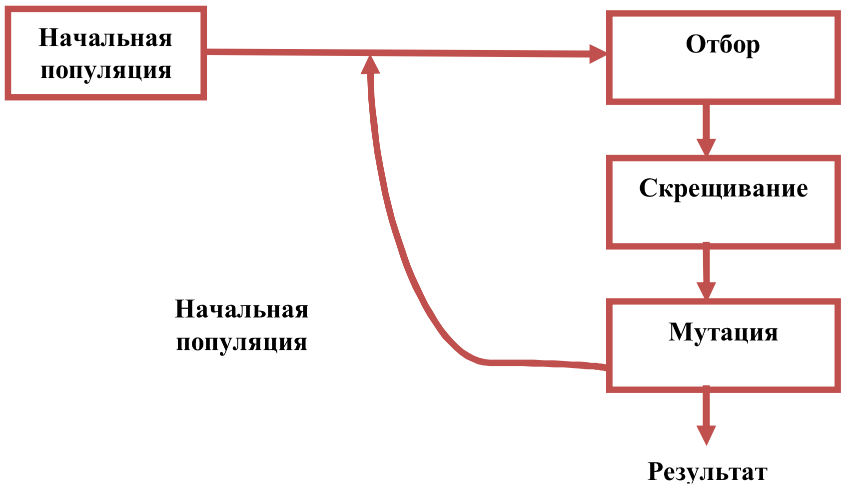

Рис. 3. Схема работы генетического алгоритма

Генетические алгоритмы имеют недостатки, которые можно представить в виде трех групп: к первой группе относится недостаточное разнообразие хромосом в популяции, которое может привести к преждевременному окончанию работы алгоритма и, как следствие, к получению «некорректного» расписания. Далее, завершение работы генетического алгоритма происходит по достижению заданного (не всегда обоснованного) числа итераций, что в ряде случаев препятствует поиску лучшего расписания [39].

Вторая группа недостатков вызвана слабым учетом специфики задачи составления расписания учебных занятий при организации ее решения с помощью генетических алгоритмов. Так, в известных генетических алгоритмах составления расписания занятий не учитывается наличие связей между объектами расписания и большое многообразие описаний самих объектов расписания.

Третья группа вызвана недостаточной систематизацией исходных данных как на этапе представления объектов генетической оптимизации, так и при организации генетических операций поиска [10, 40]. Так, расписание учебных занятий образовательных систем массового обучения является сложным информационным объектом, в котором сочетаются свойства учебных групп, дисциплин, преподавателей, аудиторий и т. д. Соответственно, хромосомы, являющиеся информационными моделями расписания, также являются сложными объектами, для которых целесообразно применение агрегативных логических моделей, основанных на рассмотрении хромосомы как многоуровневой системы, с последующей ее декомпозицией.

Исходя из проведенного анализа методов составления расписания, сделаны следующие выводы:

- Существующие методы составления расписания можно разбить на две группы: полный перебор (точные методы) и методы сокращения полного перебора (приближенные методы).
- Выбор алгоритма непосредственно зависит от условий поставленной задачи. Для задачи расписания – это количество и вид ресурсов, заданий и ограничений. Несмотря на простоту постановок, лишь немногие задачи решены точно, в частности, остаются нерешенными многие вопросы, относящиеся к общей задаче теории расписаний. В случае если независимые переменные являются дискретными и могут принимать одно значение из некоторого фиксированного набора, задача многомерной оптимизации несколько упрощается. При этом множество точек поиска становится конечным, а, следовательно, задача может быть, хотя бы в принципе, решена методом полного перебора, оптимиза-

ционные задачи с конечным множеством поиска называются задачами комбинаторной оптимизации.

4. Проблемы решения задачи расписания

В результате проведенных исследований постановок задач составления расписания, сфер применения, пространства состояний, методов решения было выявлено следующее:

- Теория изучает широкий круг задач, 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.
- Существуют разные формы представления расписаний, и довольно широк выбор терминов, определяющих одни и те же понятия.
- В задачах составления можно выделить две составляющие: общую, характеризующая особенности присущие всем задачам (атрибуты или компоненты расписания), и частную – особенности, отражающие предметную направленность (ограничения, требования к результату).
- Трудности получения и практического использования научных результатов в значительной степени обусловлены разнообразием ограничений, встречающихся в конкретных ситуациях, что приводит к неизбежной идеализации исследуемых систем.
- 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.
- Существующие методы составления расписания можно разбить на две группы: полный перебор (точные методы) и методы сокращения полного перебора (приближенные методы).
- Выбор алгоритма непосредственно зависит от условий поставленной задачи. Для задачи расписания – это количество и вид ресурсов, заданий и ограничений. Несмотря на простоту постановок, лишь немногие задачи решены точно, в частности, остаются нерешенными многие вопросы, относящиеся к общей задаче теории расписаний. В случае если независимые переменные являются дискретными и могут принимать одно значение из некоторого фиксированного набора, задача многомерной оптимизации несколько упрощается. При этом множество точек поиска становится конечным, а, следовательно, задача может быть, хотя бы в принципе, решена методом полного перебора, оптимизационные задачи с конечным множеством поиска называются задачами комбинаторной оптимизации.

Исходя из выводов по результатам исследования были выявлены причины отсутствия единой технологии составления расписания и составлен план проведения исследования по разработке данной технологии. Результаты представлены в виде причинно-следственной диаграммы Исикавы на рис. 4 и рис. 5.

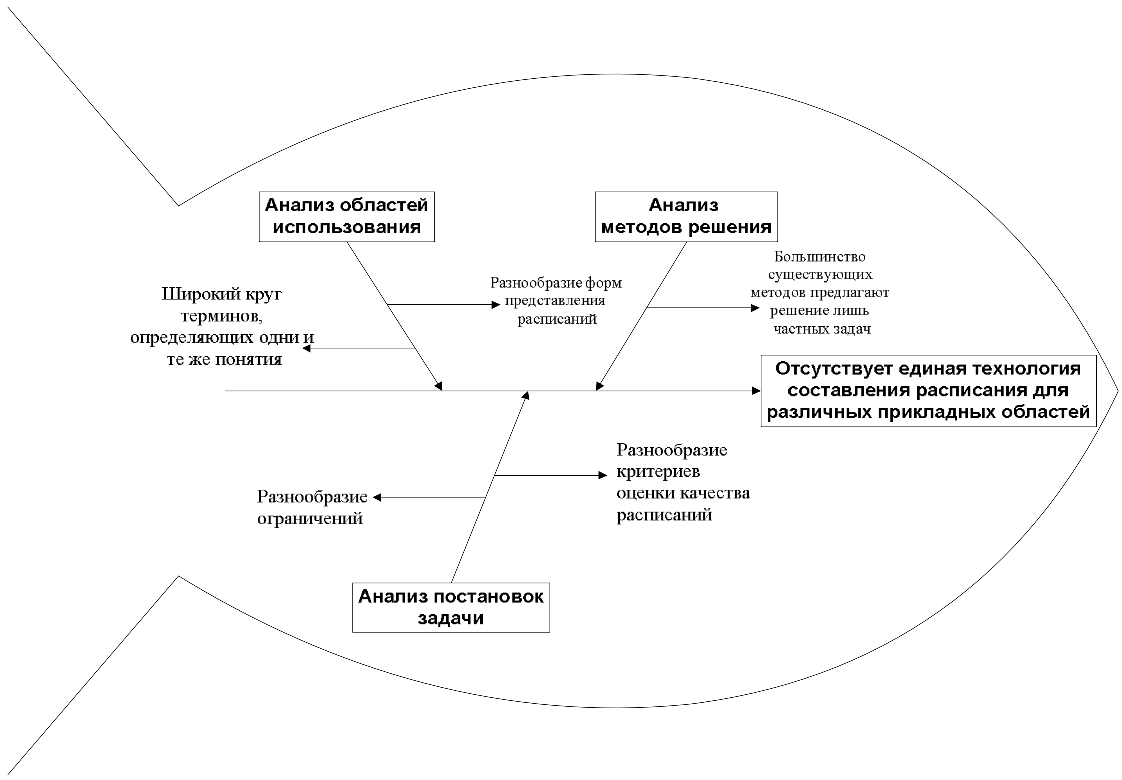

Рис. 4. Диаграмма причин отсутствия единой технологии составления расписаний

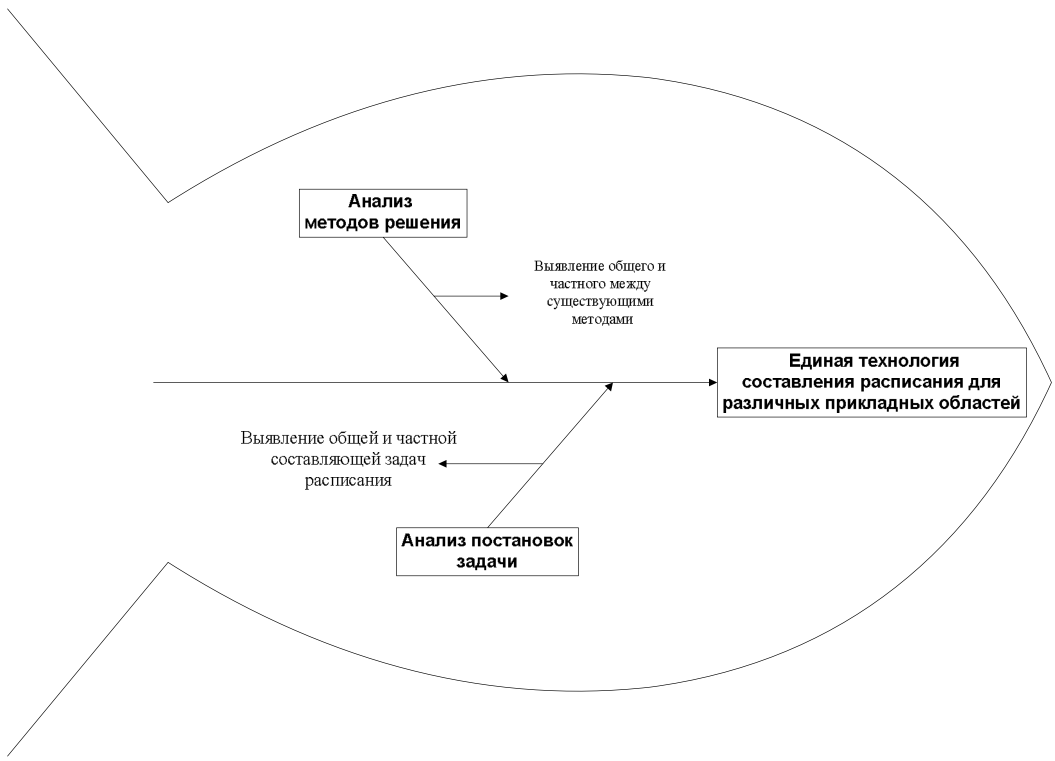

Рис. 5. План проведения исследований по разработке единой технологии составления расписаний

5. Формальная постановка задачи расписания

Сначала рассмотрим понятие расписания и понятие задачи.

Расписание является некоторым обобщенным понятием календарного плана, временного графика и т. п.

Расписание – некоторая совокупность указаний относительно того, какие именно требования какими именно ресурсами обслуживаются в каждый момент времени [2].

Задача представляет собой:

- цель (C);
- модель объекта исследования (M);
- исходное (X);
- результат (Y);
- метод преобразования исходного в результат (F);
- критерий оценки результата (K).

Задача составления расписания:

1. Цель (C).

Многие транспортные, учебные и производственные события организованы периодичным способом, повторяясь в одно и то же время через сутки, неделю, иное целое число суток. Подобная организация позволяет уменьшить затраты на планирование, что и является основной целью, поставленной задачи.

Планирование расписания – процесс планирования работ по сбору централизации и оценки качества исходных сведений необходимых и достаточных для составления расписания.

2. Модель объекта исследования (M).

Модель объекта исследования включает исходное (X) и результат (Y), и метод преобразования исходного в результат (F) (см. рис. 6.).

3. Исходное (X).

Расписание чаще всего характеризуется тремя основными характеристиками: работами, ресурсами, обеспечивающими выполнения работ, и временем, в котором данные работы упорядочены.

4. Результат (Y).

Результат – план работ включает упорядоченную во времени совокупность работ, сроки выполнения работ, результат работ, необходимые ресурсы для выполнения работ, ответственное лицо.

5. Метод преобразования исходного в результат (F).

Возможный широкий спектр частных вариантов задачи и критериев оценки результата обуславливает широкий спектр используемых подходов к решению.

Методов преобразования исходного в результат существует довольно много, какой из методов решения использовать определяется структурными и семантическими особенностями прикладной задачи расписания.

Однако есть нечто объединяющее все методы – это цель – упорядочить во времени работы и обеспечивающие их выполнения ресурсы, в соответствии с заданными условиями и ограничениями, таким образом, чтобы данное упорядочение удовлетворяло требованиям к результату (заданным критериям).

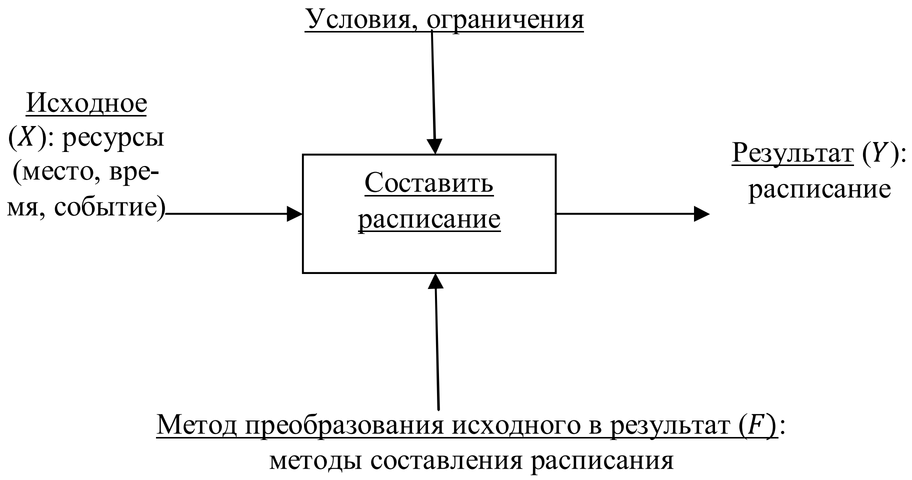

Рис. 6. Задача составления расписания

6. Критерий оценки результата (K).

Составление расписания – процесс формирования расписания удовлетворяющее основным требованиям.

Эффективное расписание – расписание, удовлетворяющее заданному набору критериев.

Критерии расписания – условия, накладываемые на показатели расписания.

Задача составления расписания считается заданной, если определены подлежащие работы; количество и типы ресурсов, выполняющих операцию; ограничения и условия, накладываемые на ресурсы, работы и на порядок распределения работ и ресурсов во времени, критерии оценки расписаний (рис. 6).

6. Связь задачи расписания и задачи упорядочения

Методы составления расписания объединены общей целью – упорядочить во времени работы и обеспечивающие их выполнения ресурсы, в соответствии с заданными условиями и ограничениями, таким образом, чтобы данное упорядочение удовлетворяло заданным критериям.

Рассмотрим общую постановку задачи упорядочения:

- есть множество объектов – T;
- P_t – набор перестановок T;
- f – функция предпочтения (ранжирования) из P_t на множестве вещественных чисел.

Найти $T' \in P_t$ такой, что: $(\forall T'')\ (T'' \in P_t)\ (T'' \neq T')\ [f(T') \geq f(T'')]$.

В контексте задачи упорядочения задача расписания представляет себя следующим образом: T – множество компонент расписания, P_t – множество всех возможных вариантов упорядочения (вариантов расписаний), а f – функция, которая при применении к любому такому упорядочению, выдает его вес (предполагается, что большие значения весов предпочтительнее малых) [41].

На основе упорядочения можно рассмотреть все варианты возможных расписаний в обобщенном виде и провести оценку мощности задачи расписания (мощность множества P_t).

Заключение

Необходимость данного исследования была вызвана ростом потребности в системах динамического планирования деятельности на предприятиях, необходимостью унификации задач расписания – решения технологической задачи сведения разнообразных содержательных постановок с целью сокращения времени, затрачиваемого на выбор метода решения для фиксированной предметной области.

Результаты исследования были использованы при постановке задачи составления и оперативной корректировки расписания на железной дороги для диспетчерских служб. Анализ исходного материала и предложенная методика позволили аргументировано выбрать в качестве метода составления расписания – динамическое моделирование. Применение методики при постановке и решении задачи управления технологическими процессами на горно-обогатительных фабриках позволило привести задачу управления к задаче динамической коррекции расписания на основе балансовой модели.

Список литературы

1. Буланова В. Б. Тайм-менеджмент как инструмент повышения эффективности // Вестник МГОУ. Серия «Экономика». – М.: Изд-во МГОУ, 2009. – № 2. – С. 88.
2. Башев В. Г. Разработка моделей и программных средств управления технологическими участками машиностроительного производства: автореферат магистерской диссертации / ДонНТУ. – Донецк, 2008.
3. Степанов С. Жизнь по расписанию: плюсы и минусы управления временем // Газета «Школьный психолог». – М.: «Первое сентября», 2007. – № 24. – [Электронный сурс]. URL: http://www.businesstest.ru/art.asp?id=275.
4. Конвей Р. В., Максвелл В. Л., Миллер Л. В. Теория расписаний. – М.: Наука, 1975. – С. 395.
5. Танаев В. С., Сотский Ю. Н., Струсевич В. А. Теория расписаний. Многостадийные системы. – М.: Наука, 1989. – С. 256.
6. Танаев В. С., Шкуба В. В. Введение в теорию расписаний. – М.: Наука, 1975. – С. 256.
7. Строкина Ю. Г. Алгоритмические процедуры формирования гетерогенных расписаний для производственных систем: диссертация на соискание ученой степени кандидата технических наук / УГАТУ. – Уфа, 1997. – С. 150.
8. Танаев В. С. Теория расписаний. – М.: Знание, 1988. – С. 32.
9. Маслов М. Г. Эвристический алгоритм решения задачи составления расписания учебных занятий в ВУЗе // Математические методы в технике и технологиях: Сб. трудов XV Международной научной конференции. В 10–и т. 2 – 4 июня 2002 г. – Тамбов, 2002. – Т. 9. – С. 86-88.
10. Кабальнов Ю. С. Композиционный генетический алгоритм составления расписания учебных занятий / Кабальнов Ю. С., Шехтман Л. И., Низамова Г. Ф., Земченкова Н. А. // Вестник УГАТУ. – 2006. – № 2.
11. Thompson J., Dowsland K. Variants of simulated annealing for the examination timetabling problem. Annals of Operational Research. – 1996. – № 63. – Pp. 105-128.
12. Cowling P., Kendall G., Soubeiga E. Hyperheuristics: A Robust Optimisation Method Applied to Nurse Scheduling // Proceedings of the VII Parallel Problem Solving From Nature (PPSN VlI), Lecture Notes in Computer Science, 2002. – Vol. 2439, Springer. – Pp. 7-11.
13. Horn J. Niche Distributions on the Pareto Optimal Front // Proceedings of the 2nd International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003), Faro Portugal, Lecture Notes in Computer Science, 2003 – Vol. 2632, Springer. – Pp. 365-375.
14. Jin H., Wong M. L. Adaptive Diversity Maintenance and Convergence Guarantee in Multiobjective Evolutionary Algorithms // Proceedings of the 2003 Congress on Evolutionary Computation (CEC 2003), 2003 – Camberra Australia, IEEE Press. – Pp. 2498-2505.

15. Kumar R., Rockett P. Improved Sampling of the Pareto-front in Multiobjective Genetic Optimization by Steady-state Evolution: A Pareto Converging Genetic Algorithm // Evolutionary Computation, 2002 – Vol. 10. – № 3. – Pp. 283-314.

16. Laumams M. Combining Convergence and Diversity in Evolutionary Multiobjective Optimization / Laumams M., Thiele L., Deb K., Zitzler E. // Evolutionary Computation, 2002 – Vol. 10. – № 3. – Pp. 263-282.

17. Socha K., Knowles J., Samples M. A Max-Min Ant System for the University Course Timetabling Problem // Ant Algorithms: Proceedings of the Third International Workshop (ANTS 2002), Lecture Notes in Computer Science, 2002. – Vol. 2463, Springer. – Pp. 1-13.

18. Socha K., Kisiel-Dorohinicki M. Agent-based Evolutionary Multiobjective Optimization // Proceedings of the 2002 Congress on Evolutionary Computation (CEC 2002), 2002. – Hawaii USA, IEEE Press. – Pp. 109-114.

19. Petrovic S., Burke E. University Timetabling // Handbook of Scheduling: Algorithms, Models, and Performance Analysis. – Chapman & Hall CRC, 2004. – Pp. 45.1-45.23.

20. Burke E., Petrovic S. Recent Research Directions in Automated Timetabling // European Journal of Operational Research, 2002. – Pp. 266-280.

21. Вагнер Г. Основы исследования операций – М.: Мир, 1973. – Т. 2. – С. 488.

22. Клеванский Н. Н., Макарцова Е. А. Формирование расписания с использованием динамических критериев загруженности // XI Международная конференция-выставка «Информационные технологии в образовании». Часть IV. – М.: МИФИ, 2001. – С. 139-140.

23. Cowling P., Kendall G., Han L. An Investigation of a Hyper-heuristic Genetic Algorithm Applied to a Trainer Scheduling Problem // Proceedings of the 2002 Congress on Evolutionary Computation (CEC 2002), 2002 – Pp. 1185-1190.

24. Deb K., Manikanth M., Mishra S. Towards a Quick Computation of Well-Spread Pareto Optimal Solutions // Proceedings of the 2nd International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003), Faro Portugal, Lecture Notes in Computer Science, 2003 – Vol. 2632, Springer. – Pp. 222-236.

25. Ross P. Learning a Procedure that Can Solve Hard Binpacking Problems: A New GA-based Approach to Hyperheuristics / Ross P. Marin-Blazquez J. G., Schulenburg S., Hart E. // Proceedings of the 2003 Genetic and Evolutionary Computation Conference (GECCO 2003), Lecture Notes in Computer Science, 2003. – Vol. 2724, Springer. – Pp. 1295-1306.

26. Soubeiga E. Development and Application of Hyperheuristics to Personnel Scheduling // PhD Thesis, School of Computer Science and Information Technology, University of Nottingham, June 2003.

27. Юсупова Н. И. Сметанина О. Н., Ахтариев А. А. Об одной классификации задач составления расписаний // Вестник УГАТУ Управление в социально-экономических и технически системах: сб. научных трудов УГАТУ. – Уфа, 2007. – № 9.

28. Батуринец Ю. А., Орехов Э. Ю. Составление плана-графика работы и отдыха летного состава. // Информационные и кибернетические системы управления и их элементы: Всероссийская молодежная научно-техническая конференция УГАТУ. – Уфа, 1997. – С. 13.

29. Орехов Э. Ю. О составлении расписания занятий в учебном заведении // Принятие решений в условиях неопределенности: Межвузовый научный сборник УГАТУ. – Уфа, 2000. – С. 172-176.

30. Попов Г. А. Формализация задачи составления расписания в высшем учебном заведении // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. – Астрахань, 2007. – № 1. – С. 5-19.

31. Методы решения задач календарного планирования. – [Электронный ресурс]. URL: http://automation-system.ru/asup/subsystem-operational-management/62-metody-resheniya-zadach-kalendarnogo-planirovaniya.html [Дата обращения: 12.04.2010].

32. Дроздов Н. Д. Алгоритмы дискретного программирования // Учебное пособие ТГУ. – Тверь, 2000. – С. 82.

33. Зайченко Ю. П. Исследование операций. – Киев.: «Высшая школа», 1975.

34. Кузнецов Ю. Н., Кузубов В. И., Волощенко А. Б. Математическое программирование. – М.: «Высшая школа», 1980.

35. Севастьянов С. В., Чемисова Д. А., Черных И. Д. О некоторых свойствах оптимальных расписаний в задаче Джонсона с прерываниями» // Дискретный анализ и исследование операций, 2006. – № 3. – С. 83-102.

36. Жданова Е. Г. Теория расписаний: учебник. – М.: МГУ, 1999.

37. Сидорин А. Б., Ликучева Л. В., Дворянкин А М. Методы автоматизации составления расписания занятий. Часть 2. Эвристические методы оптимизации // Известия ВолгГТУ. Серия "Актуальные проблемы управления, вычислительной техники и информатики в технических системах". Вып. 7: межвуз. сб. науч. ст. ВолгГТУ. – Волгоград, 2009. – № 12. – С. 120-123.

38. Сидорин А. Б., Ликучева Л. В., Дворянкин А. М. Методы автоматизации составления расписания занятий. Часть 1. Классические методы // Известия ВолгГТУ. Серия "Актуальные проблемы управления, вычислительной техники и информатики в технических системах". Вып. 7: межвуз. сб. науч. ст. ВолгГТУ. – Волгоград, 2009. – № 12. – С. 116-120.

39. Кабальнов Ю. С. Композиционный генетический алгоритм составления расписания учебных занятий / Кабальнов Ю. С., Шехтман Л. И., Низамова Г. Ф., Земченкова Н. А. // Вестник УГАТУ, 2006. – № 2. – С. 99-107.

40. Костенко В. А., Винокуров В. А. Локально-оптимальные алгоритмы построения расписаний, основанные на использовании сетей Хопфилда // Программирование, 2003. – № 4. – С. 27-40.

41. Микони С. В., Козченко Р. В., Созоновский П. Г. Выбор и упорядочение объектов с иерархической системой показателей // SCM'99: сборник докладов конференции по мягким вычислениям и измерениям СПГЭТУ. – СПб, 1999.

42. Литвинцева Л.В., Ульянов С.В., Ульянов С.С. Квантовый нечеткий вывод для создания баз знаний в робастных интеллектуальных регуляторах // Изв. РАН. – ТиСУ, 2007. – № 6.

43. Минзов А.С., Шевяхов М.Ю. Некоторые подходы к оценке информационного риска с использованием нечетких множеств // Системный Анализ в Науке и Образовании: электрон. науч. журнал. – 2010. – №1. – [Электронный ресурс]. URL: http:/www.sanse.ru/archive/15. – 0421000111\0007.