ИПМ им.М.В.Келдыша РАН • Электронная библиотека

Препринты ИПМ • Препринт № 65 за 2018 г.

ISSN 2071-2898 (Print)
ISSN 2071-2901 (Online)

Бахтин В.А., Богданов И.П.,
Осипов В.П., Рыков Ю.Г.,
Смирнов А.А., Судаков В.А.

Оптимизация перевозок
однородной продукции
между оптовыми складами

Рекомендуемая форма библиографической ссылки: Оптимизация перевозок однородной продукции между оптовыми складами / В.А.Бахтин [и др.] // Препринты ИПМ им. М.В.Келдыша. 2018. № 65. 26 с. doi:[10.20948/prepr-2018-65](https://doi.org/10.20948/prepr-2018-65)
URL: <http://library.keldysh.ru/preprint.asp?id=2018-65>

О р д е н а Л е н и н а
ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ
имени М.В.Келдыша
Р о с с и й с к о й а к а д е м и и н а у к

В.А.Бахтин, И.П.Богданов, В.П.Осипов, Ю.Г.Рыков,
А.А.Смирнов, В.А.Судаков

**Оптимизация перевозок однородной
продукции между оптовыми складами**

Москва — 2018

***Бахтин В.А., Богданов И.П., Осипов В.П., Рыков Ю.Г., Смирнов А.А.,
Судаков В.А.***

Оптимизация перевозок однородной продукции между оптовыми складами

В работе рассматривается задача формирования оптимальных расписаний перевозок однородной продукции между оптовыми складами при наличии ограничений. Предложены три подхода к решению данной задачи: метод модельных событий, способ, сочетающий имитационное моделирование и метод ветвей и границ, а также подход, основанный на использовании технологических карт и генетических алгоритмов. Описаны достоинства и недостатки каждой из указанных методологий, а также преимущества их совместного применения.

Ключевые слова: логистика, задачи составления расписаний, дискретная оптимизация

***Vladimir Aleksandrovich Bakhtin, Ilya Petrovich Bogdanov, Vladimir Petrovich
Osipov, Yuri Germanovich Rykov, Alexander Andreevich Smirnov, Vladimir
Anatolievich Sudakov***

Optimal scheduling of homogeneous products transportation

The paper regards the problem of optimal planning of homogeneous products transportation. Three solution methods are presented: model events approach, technique, comprising simulation modeling and branch and bound algorithm, and the method, based on technological charts and genetic algorithms. Peculiarities of each approach are discussed. The paper also considers the advantages of the combined application of the developed methodologies.

Key words: logistics, scheduling problems, discrete optimization

Введение

Одной из наиболее актуальных задач, стоящих перед компаниями, занимающимися производством, транспортировкой и продажей различных видов товаров, является задача формирования оптимальных расписаний перевозок между оптовыми складами при наличии ограничений. Компании стремятся выбирать маршруты движения, порядок объезда складов, очередности загрузки/разгрузки и др. таким образом, чтобы максимизировать прибыль от своей операционной деятельности при выполнении ограничений на количество, грузоподъемность и скорость доступных транспортных средств, себестоимости перевозок и хранения, вместимости складов и т.д. Дополнительные ограничения возникают вследствие необходимости осуществления перегрузочных работ, недоступности некоторых участков дорожной сети либо необходимости сопровождения груза при помощи специальных средств.

На практике поиск решения указанной задачи сопряжен с рядом сложностей. Во-первых, необходимость учета значительного количества факторов на протяжении достаточно продолжительных горизонтов планирования существенно затрудняет сведение задачи к одной из традиционных постановок задач разработки производственных расписаний [1] либо транспортных задач [2]. Таким образом, первостепенное значение приобретает вопрос построения корректной и полной модели объекта исследования, учитывающей все особенности изучаемой производственной системы.

Во-вторых, рассматриваемая задача относится к классу NP-трудных задач [3], и для ее решения требуется перебор возможных вариантов расписаний, осуществляемый, в общем случае, за время, неприемлемое с практической точки зрения. Поэтому особый интерес представляет разработка:

1. способов сокращения полного перебора при поиске точного решения;
2. методов определения допустимых расписаний, оценивающих точное решение снизу;
3. псевдополиномиальных алгоритмов решения [3];
4. методов вычисления приближенного решения.

В-третьих, возможное изменение условий функционирования моделируемой системы предъявляет ряд требований к технологиям решения в части относительно несложной адаптации к модификации исходных данных.

В настоящей работе предложены три подхода к решению рассматриваемой задачи, построенные с учетом вышеприведенных замечаний и основанные на различных способах формализации пространства задачи.

Общая постановка задачи

Пусть у некоторой компании имеется два вида оптовых складов ограниченной вместимости: A и B . На склады типа A поступает продукция (которую мы будем считать однородной и бесконечно делимой), либо вследствие осуществления производственного процесса компании, либо вследствие транспортировки извне. Поставки на склады типа A осуществляются в соответствии с фиксированным графиком, от которого допускается отклонение в заданных пределах. Также на складах типа A оборудованы погрузочные терминалы, на которых с заданной скоростью может одновременно загружаться не более определенного числа транспортных средств.

Со складов типа B продукция либо продается внешнему потребителю, либо поступает в дальнейший производственный процесс компании. На складах указанного типа оборудованы разгрузочные терминалы, на каждом из которых в любой момент времени может разгружаться не более определенного числа транспортных средств (при этом может быть дополнительно введено требование разгрузки продукции с конкретного склада типа A на конкретном терминале рассматриваемого склада типа B). Вывоз продукции со складов типа B осуществляется в соответствии с заданными правилами.

Движение между складами осуществляется по набору маршрутов вида (A_i, B_j) , где A_i – один из складов типа A , а B_j – один из складов типа B (при этом каждая пара складов может быть соединена более чем одним маршрутом). Все маршруты состоят из участков, характеризующихся протяженностью, дорожными условиями, а также ограничениями на необходимость сопровождения специализированными средствами (например, из соображений безопасности). Перевозку грузов по указанным маршрутам осуществляет заданное количество курсирующих транспортных средств перевозки (ТСП). Указанные ТСП характеризуются значениями своих вместимостей и скоростей, техническими особенностями (габаритами, наличием/отсутствием дополнительного оборудования, проходимостью), а также способностью или неспособностью преодолевать определенные участки маршрутов без сопровождения.

Каждое отдельное ТСП задействовано на доставке продукции с одного из складов типа A на один из складов типа B . При наличии технической возможности ТСП загружает товары на складе A_i , доставляет и разгружает их на складе B_j . При невозможности загрузиться (соответственно, разгрузиться) на складе (вследствие непригодности складских терминалов для обслуживания ТСП из-за его габаритов, отсутствия нужного оборудования, запрета на въезд грузовиков в города и т.д. или вследствие невозможности преодолеть определенный участок маршрута из-за дорожных или погодных условий) ТСП должно принять продукцию от другого ТСП (соответственно,

перегрузить продукцию на другое ТСП), обладающего нужными техническими характеристиками.

При проезде по опасным участкам маршрута некоторым ТСП может требоваться специальное сопровождение, осуществляемое заданным набором транспортных средств сопровождения (ТСС), каждое из которых характеризуется своей скоростью движения, а также предельным числом ТСП в кортеже.

Один из простейших вариантов сети маршрутов, по которым осуществляется доставка продукции, приведен на рис. 1.

Рис. 1. Пример схемы маршрута

ТСП могут быть приписаны к определенному пути следования на протяжении всего периода планирования либо обладать возможностью выбирать из нескольких маршрутов.

В силу необходимости производить текущий ремонт и техобслуживание, а также своевременно пополнять запасы горюче-смазочных материалов, каждое ТСП должно с заданной периодичностью делать перерывы в работе и прибывать на место базирования.

Если на складе одновременно скапливается больше ТСП, чем могут обслужить терминалы данного склада, то вступают в действие правила очередности загрузки/разгрузки, основанные на заданной системе приоритетов.

Наконец, для каждого ТСП, ТСС, участка маршрута, а также терминала загрузки/разгрузки могут быть заданы временные интервалы, в которые они будут недоступны (может закончиться срок аренды ТСП или склада, дорога может стать непригодной для проезда, ТСС могут потребоваться для работ в другом регионе и т.д.).

Задача построения оптимального расписания перевозки товаров между складами заключается в выборе режима функционирования производственного комплекса, обеспечивающего максимум целевой функции, отражающей баланс максимизации стоимости/объема перевезенной и переданной на продажу (либо в дальнейший технологический процесс) продукции и минимизации выбранного набора затрат (аренда и ремонт ТСП и ТСС, приобретение горюче-смазочных материалов, эксплуатация складов, штрафы за отклонение от графика подвоза продукции на склады типа A и т.д.) при выполнении всех вышеперечисленных ограничений.

О возможных подходах к решению задачи

Одна из возможных методологий решения поставленной задачи основывается на следующем способе формализации пространства задачи:

1. производственный цикл компании представляется в виде последовательностей работ (движение по определенному участку маршрута с грузом, движение по определенному участку без груза, ожидание в очереди на загрузку/разгрузку, процесс загрузки/разгрузки, ожидание ТСС и т.д.);
2. весь горизонт планирования разбивается на конечное число временных периодов (продолжительность которых равна времени выполнения самой быстрой технологической операции);
3. вводится набор фазовых булевых переменных — индикаторов выполнения конкретными производственными единицами (ТСП, ТСС и т.д.) конкретных работ на конкретных маршрутах в конкретные периоды времени.

Пусть $i=1,...,I$ — набор рассматриваемых производственных единиц, $m(i)=1,...,M_i$ — маршруты, на которых может работать i -я производственная единица, $j(i,m(i))=1,...,J_{i,m(i)}$ — работы, выполняемые i -й производственной единицей на $m(i)$ -м маршруте, $\tau=1,...,T$ — временные периоды. Тогда указанный набор фазовых переменных имеет вид:

$$x_{i,m,j}^{\tau} = \begin{cases} 1, & \text{если } i\text{-я производственная единица} \\ & \text{выполняет } j\text{-ю работу на } m\text{-м маршруте} \\ & \text{в период времени } \tau, \\ 0, & \text{иначе.} \end{cases}$$

При добавлении дополнительных дискретных переменных, необходимых для формализации процессов перегрузки, сопровождения, поставок на склады типа A и вывоза со складов типа B , а также дополнительных непрерывных

переменных, соответствующих объемам перевозимой и хранящейся на складах продукции, исходную задачу можно записать как задачу математического программирования. В общем случае данная задача относится к классу задач дискретно-непрерывной нелинейной оптимизации, и поиск ее решения сопряжен со значительными трудностями.

Введением ряда упрощающих предположений, а также добавлением блока новых булевых переменных — индикаторов завершения выполнения текущей работы производственной единицей:

$$y_{i,m,j}^{\tau} = \begin{cases} 1, & \text{если } i\text{-я производственная единица} \\ & \text{завершает выполнение } j\text{-й работы} \\ & \text{на } m\text{-м маршруте в период времени } \tau, \\ 0, & \text{иначе,} \end{cases}$$

рассматриваемую задачу можно привести к частично целочисленной задаче линейного программирования и применить для ее решения, например, метод ветвей и границ в сочетании с методами внутренней точки либо симплекс-методом.

Приведем примеры линейных ограничений, определяющих допустимое множество задачи. Для упрощения записи далее в данном разделе будем предполагать, что каждая производственная единица работает только на одном маршруте, и будем опускать соответствующие индексы. Также для простоты не будем рассматривать ограничения на недоступность производственных единиц в заданные временные интервалы (учет указанных ограничений в общем случае потребует тривиальных модификаций построенных уравнений). Необходимость отдыха и ремонта будем моделировать с помощью специальной дополнительной работы.

Очевидные ограничения на возможные состояния производственной единицы:

1. в каждый период времени τ каждая производственная единица i выполняет одну и только одну работу:

$$\sum_{j=1}^{J_i} x_{i,j}^{\tau} = 1, \quad \forall i = 1, \dots, I, \quad \forall \tau = 1, \dots, T;$$

2. если в период времени τ производственная единица i выполняет работу j , за которой, в соответствии со схемой осуществляемого технологического процесса, может непосредственно следовать выполнение одной из работ из множества $\mathbf{J}(i, j)$, то в следующий период времени

производственная единица i будет выполнять работу из множества $\{j\} \cup \mathbf{J}(i, j)$:

$$x_{i,j}^{\tau} \leq x_{i,j}^{\tau+1} + \sum_{h \in \mathbf{J}(i,j)} x_{i,h}^{\tau+1}, \quad \forall i=1, \dots, I, \quad \forall j=1, \dots, J_i, \quad \forall \tau=1, \dots, T-1.$$

Ограничения — связи между индикаторами выполнения и завершения выполнения работы:

1. после завершения выполнения текущей работы j производственная единица i должна приступить к выполнению одной из работ из множества $\mathbf{J}(i, j)$:

$$y_{i,j}^{\tau} \leq \sum_{h \in \mathbf{J}(i,j)} x_{i,h}^{\tau+1}, \quad \forall i=1, \dots, I, \quad \forall j=1, \dots, J_i, \quad \forall \tau=1, \dots, T-1;$$

2. производственная единица не может завершать выполнение работы, если в данный период времени она не выполняет указанную работу:

$$y_{i,j}^{\tau} \leq x_{i,j}^{\tau}, \quad \forall i=1, \dots, I, \quad \forall j=1, \dots, J_i, \quad \forall \tau=1, \dots, T.$$

Ограничения для работ с известной продолжительностью выполнения (движения между двумя пунктами с заданной скоростью, процессов загрузки/разгрузки и др.): например, пусть производственная единица i выполняет за L временных периодов работу j , которой, в соответствии со схемой осуществляемого технологического процесса, может непосредственно предшествовать выполнение одной из работ из множества $\mathbf{G}(i, j)$. Для простоты предположим, что за работами из множества $\mathbf{G}(i, j)$ может следовать только работа j . Тогда:

$$\sum_{h \in \mathbf{G}(i,j)} y_{i,h}^{\tau} = y_{i,j}^{\tau+L}, \quad \forall \tau=1, \dots, T-L.$$

Ограничения на пропускные способности (участков дорог, терминалов загрузки/разгрузки и др.): например, пусть $\mathbf{I}$ — множество ТСП, а работы $j_v(i)$, $i \in \mathbf{I}$, соответствуют разгрузке данных ТСП на одном из терминалов, способном одновременно обслуживать не более R ТСП. Тогда:

$$\sum_{i \in \mathbf{I}} x_{i,j_v(i)}^{\tau} \leq R, \quad \forall \tau=1, \dots, T.$$

Ограничения, связанные с необходимостью организации сопровождения (простейший случай): пусть $\mathbf{I}_1$ — множество ТСП, которым требуется сопровождение для проезда по заданному участку транспортной сети, $\mathbf{I}_2$ — множество ТСС, работы $j_v(i)$, $i \in \mathbf{I}_1$ (соответственно, $j_w(i)$, $i \in \mathbf{I}_2$),

соответствуют движению ТСП с сопровождением (соответственно, осуществлению сопровождения) на рассматриваемом участке. Дополнительно предположим, что каждое ТСС может сопровождать единственное ТСП. Тогда:

$$\sum_{i \in \mathbf{I}_1} x_{i, j_v(i)}^\tau = \sum_{i \in \mathbf{I}_2} x_{i, j_w(i)}^\tau, \quad \forall \tau = 1, \dots, T.$$

Ограничения, связанные с необходимостью организации перегрузочных работ (простейший случай): пусть $\mathbf{I}_1$ – множество ТСП, которые не могут разгружаться непосредственно на складах типа B и вынуждены выгружать продукцию в заданном узле транспортной сети, $\mathbf{I}_2$ – множество ТСП, способных принимать продукцию в указанном узле и отвозить ее для разгрузки на склад типа B , работы $j_v(i)$, $i \in \mathbf{I}_1$ (соответственно, $j_w(i)$, $i \in \mathbf{I}_2$), соответствуют разгрузке (соответственно, загрузке) в рассматриваемом узле. Дополнительно предположим, что вместимости всех ТСП одинаковы. Тогда (аналогично предыдущему пункту):

$$\sum_{i \in \mathbf{I}_1} x_{i, j_v(i)}^\tau = \sum_{i \in \mathbf{I}_2} x_{i, j_w(i)}^\tau, \quad \forall \tau = 1, \dots, T.$$

Проиллюстрируем основные принципы формирования целевой функции задачи на следующем примере. Пусть $\mathbf{I}$ – множество ТСП (для простоты предположим, что все ТСП разгружаются на единственном терминале, способном одновременно обслуживать не более одного ТСП), а работы $j_v(i)$, $i \in \mathbf{I}$, соответствуют разгрузке данных ТСП. Обозначим через c_i объем продукции, загружаемой i -м ТСП в хранилище в течение одного временного периода. Тогда слагаемое целевой функции, соответствующее суммарному объему продукции, загруженной на склад в течение всего горизонта планирования, равно

$$\sum_{i \in \mathbf{I}} \sum_{\tau=1}^T (c_i \cdot x_{i, j_v(i)}^\tau).$$

В теории построенная задача с линейными ограничениями решается значительно проще и быстрее, чем задача с нелинейными ограничениями. Тем не менее, подход к решению, основанный на формализации в виде частично целочисленной задачи линейного программирования, обладает существенными недостатками, затрудняющими его применение на практике. Во-первых, для реальных производственных комплексов допустимое множество задачи имеет очень большую размерность. Во-вторых, уменьшение продолжительности временного периода, на который разбивается горизонт планирования, приводит к экспоненциальному росту размерности. В-третьих, при изменении условий функционирования моделируемого производственного комплекса может потребоваться существенное перестроение математической формализации,

закключающееся как в модификации системы ограничений, так и в добавлении новых переменных.

Принципиально иная методология решения указанной задачи, находящаяся, так сказать, на другой границе спектра возможных подходов, основывается на интенсивно развивающихся в настоящее время технологиях мультиагентного моделирования (мультиагентные технологии – МАТ), например, см. [4]. МАТ представляют собой относительно недавно (конец 1990-х гг.) появившийся, но уже достаточно разработанный для индустриального применения подход, являющийся далеко идущим развитием классических подходов линейного/выпуклого и динамического программирования. МАТ являются достаточно общей методологией и предназначены для решения задач в условиях высокого уровня неопределенности. Они основаны на «локальной модели рынка», где «элементарные» агенты реализованы программно, имеют достаточный уровень автономности, способны договариваться и реализовывать свои цели в условиях быстро меняющихся условий. По существу, МАТ являются глубоко эвристическим методом, предназначенным для поиска точек глобального/локального равновесия целевых функций с очень большим числом переменных и к тому же способных динамически меняться.

Однако, вследствие высокого уровня универсальности и необходимости задавать для каждого агента значительное количество правил поведения, учитывающих все особенности моделируемого процесса, для решения рассматриваемой задачи, по-видимому, не имеет смысла использовать МАТ, если только в задачу не будет привнесен элемент внешней неопределенности, например, поступление в случайном порядке заявок на перевозку и такое же случайное изменение состояния транспортных путей между складами. Но это будет уже другой класс задач.

Таким образом, рассмотрены две методологии: «классическая», основанная на дискретизации горизонта планирования и введении булевых переменных — индикаторов выполнения работ (строго обоснованная с формальной точки зрения, но требующая значительных вычислительных и временных ресурсов для решения и существенных усилий для учета изменяющихся внешних условий), и «эвристическая», основанная на применении МАТ (обладающая высокой степенью универсальности, но не обоснованная математически и не гарантирующая достижения оптимума). Далее будет представлено три подхода (построенных авторами настоящей работы), которые по своим степеням универсальности, обоснованности, адаптируемости и сложности находятся в диапазоне между вышеприведенными «классической» и «эвристической» методологиями.

Алгоритм, основанный на комбинированном применении имитационного моделирования и метода ветвей и границ

Как уже было показано, в общем случае множество всевозможных вариантов расписаний работы рассматриваемого производственного комплекса настолько велико, что его невозможно охватить полным перебором за разумное время. В задаче имеется достаточное количество ограничений, приводящих к тому, что среди всех возможных расписаний многие являются недопустимыми. Объединение процесса перебора с моделированием объектов задачи позволяет реализовать отсечение заведомо недопустимых расписаний. Далее приводится описание метода на примере упрощенной задачи и возможных модификаций для учета дополнительных условий.

Пусть имеется один склад типа A . В него поступают товары по заданному плану, при этом вместимость склада ограничена. Необходимо вовремя вывозить продукцию, чтобы не допустить переполнения склада. Пусть на складе одновременно может загружаться не более одного ТСП. Таким образом, ТСП, загружающиеся одно за другим, образуют некоторую последовательность. Выбор этой последовательности является основой для перебора.

Количество всевозможных последовательностей не превышает числа n^k , где n – число ТСП, k – максимально возможная длина последовательности, $k = V_p / V_{\min}$, где V_p – объем продукции, поступившей за время планирования, а $V_{\min}$ – минимальная вместимость ТСП. Решение поставленной задачи придется искать среди огромного числа вариантов, большинство из которых заведомо недопустимо. Рассмотрим один из способов сокращения перебора.

Построить все варианты можно следующим образом. Имеется начальное состояние, означающее, что ни одно ТСП еще не загружалось. Оно порождает состояния, каждое из которых означает, что первым на складе загружается соответствующее ТСП: $s_1, s_2, \dots, s_n$ (всего у нас есть n ТСП). Следующее поколение состояний определяет, какое ТСП загружается вторым, и т.д. (рис. 2).

Допустим, что у каждого ТСП имеется только один маршрут (см. рис. 3). Он состоит из точек, циклически следующих друг за другом. Точка 5 означает загрузку на складе A , точка 9 означает разгрузку на складе B , а точка 1 – ожидание пустого ТСП возле склада B .

Состояние содержит дополнительную информацию, а именно: для каждого ТСП список точек пути и интервал времени, в котором ТСП может находиться в данной точке; объем продукции на складе A для текущего состояния. Заводим начальное состояние (рис. 3) и для каждого ТСП в соответствующий список добавляем точку, к которой оно сейчас направляется. Рассчитываем

время, через которое оно туда доберется, и записываем его в нижнюю границу интервала.

Рис. 2. Построение всевозможных вариантов последовательностей загрузки

Рис. 3. Начальное состояние

Создавая новое состояние, мы хотим смоделировать загрузку очередного ТСП на складе A и его разгрузку на складе B . Процесс порождения нового состояния выглядит следующим образом. Содержимое нового состояния копируется из родительского. Это состояние соответствует определенному

ТСП, которое будет загружаться после ТСП из предыдущего состояния. В нашем случае – s_1 . Добавляем в соответствующий ему список точек пути точки до склада A (4 и 5), и устанавливаем в интервалах минимальное время, когда ТСП сможет туда прибыть из точки 3, если отправится из нее во время $t_{1,\min}$. Сверху ограничить интервалы пока нечем (рис. 4).

Рис. 4. Шаг алгоритма. Нижняя граница интервала

Далее в точке 5, соответствующей загрузке на складе A , определяем времена t_1 и t_2 таким образом, что если ТСП прибудет раньше t_1 , то не хватит продукции сразу полностью его загрузить, а если прибудет позже t_2 , то склад переполнится. В качестве нижней границы возьмем большее из t_1 и $t_{3,\min}$. Если t_1 получилось больше периода моделирования, то, значит, мы достигли максимальной длины последовательности, и порождать новые состояния от этого не нужно. В противном случае, для точки 4 определяем время $t_{2,\max}$, не позже которого ТСП должно выйти из точки 4, чтобы успеть к t_2 в точку 5. То же самое делаем для точки 3 и в общем случае так далее до начала списка (рис. 5).

После того как ТСП загрузилось, его надо отправить разгружаться на склад B . Добавляем последовательно точки и минимальное время, когда ТСП будет в очередной точке, если считать, что загружаться оно начало во время t_1 . Далее вычитаем из объема продукции текущего состояния объем ТСП s_1 . Теперь состояние готово (рис. 6), и можно переходить к следующим шагам.

Рис. 5. Шаг алгоритма. Верхняя граница интервала

Рис. 6. Переход на следующий шаг алгоритма

Каждое новое состояние добавляет к последовательности очередное ТСП, что, в свою очередь, накладывает дополнительные ограничения на множество расписаний. Эти ограничения запоминаются в состоянии. Если новое состояние вносит ограничения, исключающие расписание (например, в каком-либо элементе списка $t_{\min} > t_{\max}$), то состояние удаляется. Состояние удаляется и в случае, если все его потомки удалены. При выборе очередного ТСП следует учитывать его «качество», которое зависит от скорости ТСП, объема перевозимой продукции, стоимости обслуживания и др.

После применения данного алгоритма получается дерево состояний, каждый лист которого скрывает за собой некоторую группу расписаний. В

процессе построения дерева отбрасываются заведомо недопустимые варианты расписаний. Группы не пересекаются, поскольку каждое расписание из группы подразумевает определенную последовательность загрузки ТСП, которая однозначно задается состоянием. Кроме того, зная такую последовательность, можно оценить максимально возможную выгоду лучшего расписания из группы. И если выгода прогнозируется недостаточно хорошей, то можно не рассматривать группу целиком. Такую оценку можно получить еще на этапе построения дерева, что позволит не строить его полностью, что необходимо, поскольку потенциальный размер дерева может быть очень большим.

Данный алгоритм реализует упорядочивание ТСП на складе А и обеспечивает своевременный вывоз продукции во избежание переполнения склада. Теперь можно предложить модификации для решения задачи с дополнительными условиями.

Предположим, что очередное выбранное ТСП не способно само загрузиться на складе А и может быть загружено только из другого ТСП. В этом случае состояние для него строится иначе. Выбранное ТСП отправляется в точку, где в него будет перегружена продукция из другого ТСП, после чего выбирается ТСП, которое будет осуществлять погрузку. Оно отправляется загружаться, а оттуда следует в точку для осуществления перевалки груза в первое ТСП. В случае, когда первое ТСП заполняется полностью, оно отправляется на разгрузку.

В случае, когда у ТСП имеется возможность двигаться по разным маршрутам, это учитывается при выборе очередного ТСП.

Дополнительные ограничения решаются отдельными алгоритмами, которые работают над полученной таблицей списков точек пути для каждого транспортного средства ($ТС = ТСП + ТСС$), сужая вычисленные временные интервалы в определенных точках. Для манипуляций над списками точек пути необходимо задавать связи между отдельными точками разных путей. Связи бывают двух типов. Первый тип указывает, что действия связанных ТС в указанных точках должны выполняться одновременно и, соответственно, их временные интервалы приравниваются, а длительность действия у всех приравнивается к самой большой. Второй тип указывает, что одно ТС должно выполнить действие в заданной точке после другого, и нижняя граница его интервала должна быть больше, чем нижняя граница интервала другого ТС плюс время, которое другое ТС будет выполнять положенное действие. Например, точка, в которой ТСП выполняет загрузку, связывается с точкой, в которой загружалось предыдущее ТСП. Тогда как точки, в которых одно ТСП переваливает груз в другое ТСП, связываются первым типом связи. Механизм связей позволяет обеспечить целостность таблицы, что позволяет решать подзадачи упорядочивания ТСП при разгрузке, распределении запросов на сопровождение ТСС и объединении таблиц разных складов. Ограничения на точках маршрута, протяженные во времени, задаются множествами интервалов,

которые принимаются во внимание при построении временного интервала в данной точке пути.

В зависимости от параметров конкретной задачи алгоритмы решения подзадач должны быть разумным образом скомбинированы. Например, если имеется большое количество ТСС, которые всегда способны выполнить запросы на сопровождение без значительных затрат, то есть смысл отложить решение подзадачи сопровождения до того, как в основном алгоритме дойдем до листа дерева поиска. Если же количество ТСС невелико и решение этой подзадачи сильно изменяет построенную таблицу, то решать эту задачу можно после получения каждого нового состояния для того, чтобы оценить его допустимость и качество.

Приведенный алгоритм можно рассматривать как комбинацию метода имитационного моделирования и метода ветвей и границ. Он хорошо распараллеливается, т.к. различные последовательности состояний могут строиться и оцениваться одновременно. Применяя этот алгоритм, можно получить точное решение, но, в худшем случае, он может свестись к полному перебору. В процессе построения дерева состояний все расписания разбиваются на группы, и становится возможным оценить верхнюю границу качества расписаний из группы. Таким образом, выбирая группы с потенциально лучшими расписаниями, можно относительно быстро найти хорошее решение.

Поиск рациональных решений методом модельных событий (ММС)

Решение задачи базируется на использовании двух графов. 1-й граф – это граф, соответствующий реальной логистической сети. Он может быть изменен пользователем системы. Основными особенностями являются тип дуги графа и интервал времени, в котором эта дуга принадлежит заданному типу. Для ТСП указывается скорость прохождения дуг заданных типов и необходимость сопровождения. Обслуживание ТСП организуется в вершинах графа. В вершинах графа также образуются очереди на обслуживание. В вершинах графа для ТСП могут быть указаны вместимости складов для хранения перевозимой продукции. Для складов может быть задан план пополнения. 2-й граф – это граф модельных событий. Вершины соответствуют событиям, дуги показывают, какие события могут вызвать планирование следующих событий. Можно говорить, что граф модельных событий «работает» в каждой вершине логистического графа.

Оптимизация осуществляется эвристическим алгоритмом на уровне правил принятия решений по планированию тех или иных событий на графе второго типа. Планируемые события заносятся в список будущих событий (СБС). Его структура:

1. время свершения события;

2. код события (соответствует вершине графа второго типа);
3. место события (соответствует вершине графа первого типа);
4. массив идентификаторов ТС, к которым относится событие.

Общий алгоритм ММС [5]:

1. извлечь из СБС событие с минимальным временем;
2. если время события больше времени моделирования, то окончание работы, иначе запустить обработчик соответствующего события;
3. перейти на шаг 1.

Таким образом, моделирование осуществляется с переменным по времени шагом, что позволяет достичь максимальной точности моделирования и избежать обработки «лишних» квантов времени, которые возникают в системах с фиксированным по времени шагом.

На рис. 7 показан граф взаимосвязи событий для рассматриваемой задачи. В системе используются события со следующими кодами:

1. переход из точки в точку;
2. окончание загрузки ТСП;
3. окончание разгрузки ТСП;
4. пополнение склада типа А с производства;
5. окончание перегрузочных работ.

Рис. 7. Граф взаимосвязи событий

Перед запуском имитационной модели СБС заполняется известными на момент запуска событиями:

1. планируются события прихода ТС в точки маршрута в соответствии с их начальным размещением;

2. планируются моменты пополнения складов типа А с производства.

Транспортный граф в исходных данных задается в формате JSON. На верхнем уровне он содержит следующие блоки:

1. блок arcs – множество дуг. Для каждой дуги задается доступность с указанием времени и типа дуги (например, "available": [{"dateb": "2017-01-01", "datee": "2018-01-01", "condition": "no_danger"}]). В тестовых расчетах использовались дуги двух типов danger – сложные/опасные условия, no_danger – обычные/безопасные условия, однако тип дуги – это не свойство модели, а элемент исходных данных, программа позволяет задать произвольное количество типов дуг;

2. блок routes – маршруты. Состоят из множества неориентированных дуг. Содержат начальную точку. Для ТСП начальная точка – это склад типа А с производством;

3. блок points – точки маршрута. Обязательно указываются только точки, содержащие склады. Для складов типа А задаются графики пополнения (например, "plan": [{"dateb": "2017-06-05", "datee": "2017-06-30", "vol": 19800, "max_vol": 21000, "step": 0}, {"dateb": "2017-07-01", "datee": "2017-07-17", "vol": 19000, "max_vol": 19500, "step": 500}]);

4. блок vehicles – ТС. ТСП содержат емкости для перевозки продукции. У ТСС емкостей нет. Для типов дуг указывается скорость и необходимость сопровождения (например, "move":{"no_danger": {"speed": 12.5}, "danger": {"speed": 8, "sup": true}});

5. блоки dateb и datee – начало и конец периода моделирования.

Алгоритм обработки события «Переход из точки в точку»

Цикл по всем ТС, с которыми связано событие:

Если точка конечная на маршруте, то

Если не ТСС

Если есть склад в точке, то запланировать обслуживание

Иначе поместить в очередь на сопровождение

Иначе

Если есть, кого сопровождать, то запланировать сопровождение

Иначе

Если везет продукцию и есть очередь на сопровождение, то запланировать сопровождение

Иначе запланировать переход в следующую точку

Конец цикла по ТС.

Алгоритм обработки события «Окончание загрузки ТСП»

Запланировать переход к следующей точке по направлению к источнику продукции.

Если очередь в точке не пуста, то извлечь первое по приоритету ТСП (среди равных приоритетов дисциплина FIFO) и запланировать его обслуживание.

Алгоритм обработки события «Окончание разгрузки ТСП»

Запланировать переход к следующей точке по направлению к приемнику продукции.

Если очередь в точке не пуста, то извлечь первое по приоритету ТСП (среди равных приоритетов дисциплина FIFO) и запланировать его обслуживание.

Алгоритм обработки события «Пополнение склада типа А с производства»

Цикл по всем точкам всех складов типа А

Пополнить склад

Запланировать обслуживание ТСП, если есть очередь в точке

Конец цикла по складам типа А.

Алгоритм обработки события «Окончание перегрузочных работ»

Цикл по ТСП в очереди на перегрузку в точке

Загрузить ТСП из очереди

Если удалось полностью загрузить ТСП, то запланировать его движение

Если ТСП, с которого перегружали продукцию, освобождено полностью, то выход из цикла

Конец цикла по ТСП.

Запланировать возвращение ТСП, с которого осуществлялась перегрузка, к складу типа А.

Для минимизации стоимости владения парком ТС используется период доступности ТС "available": ["2017-01-01","2017-12-01"], [...]], для максимизации суммарной вывозимой (для продажи либо для передачи в дальнейший производственный процесс) продукции варьируется назначение

ТС на маршруты: "allocation": [{"dateb": "2017-06-05", "datee": "2017-07-17", "route": "ni", "isfix": *false*}].

Основные достоинства рассмотренного подхода: расписание составляется с максимальной точностью по времени (отсутствует искусственная дискретизация времени), относительно быстро и гарантированно находим допустимое решение, близкое к оптимальному, сравнительно легко адаптируем, так как граф логистической сети задается в исходных данных и может быть быстро перестроен, а алгоритмы обработки событий и граф событий задаются на языке высокого уровня в нотации, приближенной к описанию бизнес-правил, и могут быть быстро и легко изменены.

Основные недостатки: нахождение глобального оптимума не гарантируется, для поиска оптимального решения необходим перебор комбинаций значений параметров имитационной модели, и для каждой такой комбинации значений нужны повторные прогоны модели. Такие прогоны могут быть организованы параллельно на узлах вычислительного кластера. Технологически такая процедура организована на веб-сервисах поддержки принятия решений *ws-dss.com*. Для этого используется система диспетчеризации заданий *sidekiq*. Система *sidekiq* запускает отдельные рабочие потоки на произвольном числе узлов вычислительного кластера. Рабочий поток получает задание с параметрами имитационной модели из *InMemory* базы данных *Redis*. Результаты моделирования сохраняются в базу данных *PostgreSQL*. В случае многокритериальных оценок результатов моделирования, они могут быть ранжированы с помощью оригинальных процедур многокритериального анализа альтернатив: гибридной функции предпочтений [6] и метода, основанного на анализе нечетких областей предпочтений [7].

Методология, основанная на формировании технологических карт и применении генетического алгоритма

В основе третьего предлагаемого подхода – отбор наилучшего расписания производственного комплекса среди всех возможных вариантов расписаний (и допустимых и недопустимых), осуществляемый с помощью генетического алгоритма.

Для каждого типа производственных единиц: ТСП и ТСС, относящихся к конкретному классу и работающих на конкретном маршруте, средств доставки продукции на склады типа *A* (СДП) и средств вывоза продукции со складов типа *B* (СВП) – формируется своя технологическая карта – ориентированный граф, узлы которого соответствуют состояниям производственной единицы, а дуги – видам деятельности (загрузке, разгрузке, движению, ожиданию, отдыху/ремонту и т.п.). Каждой дуге приписываются характеристики соответствующей технологической операции: временные и финансовые затраты на выполнение работы, а также стоимость и объем загружаемой или

разгружаемой продукции. Например, рассмотрим производственный комплекс, схема которого изображена на рис. 1. В состав данного комплекса могут входить следующие типы производственных единиц:

1. ТСП, работающие на маршруте № 1 и требующие сопровождения при движении по опасному участку;
2. ТСС, работающие на маршруте № 1;
3. ТСП, работающие на маршруте № 2 и способные передвигаться от склада А до склада В;
4. ТСП, работающие на маршруте № 2 и способные передвигаться от склада А до узла, где осуществляются перегрузочные работы;
5. СДП;
6. СВП и т.д.

Примеры упрощенных технологических карт для указанных производственных единиц приведены на рис. 8–10. Кроме состояний производственных единиц и операций, осуществляемых ими, на данных картах отображены складские хранилища. При выполнении ТСП, СДП и СВП работ по загрузке/разгрузке в указанных хранилищах происходит изменение объема продукции (см. узлы, выделенные цветом, и инцидентные им дуги на рис. 9, 10).

Рис. 8. Технологическая карта ТСС, работающего на маршруте № 1

Рис. 9. Технологическая карта ТСП, работающего на маршруте № 2 и способного передвигаться от склада А до склада В

Рис. 10. Технологическая карта СВП

Расписание каждой производственной единицы строится как траектория ее движения по соответствующей технологической карте за весь период планирования. Набор расписаний всех производственных единиц, учитываемых в расчете, формирует расписание производственного комплекса.

Рассматривая произвольные траектории на технологических картах и объединяя их в наборы, получаем множество всех возможных расписаний, из которого будет осуществляться отбор. При этом большая часть указанных наборов соответствует недопустимым расписаниям, поэтому для целей расчета необходимо ввести штрафы за нарушения технологических, операционных и т.д. ограничений отдельными производственными единицами, например:

1. за попытку загрузить ТСП объемом продукции, превосходящим его вместимость;
2. за попытку разгрузить пустое ТСП и др.;

и штрафы за нарушения ограничений всей совокупностью производственных единиц, например:

1. за попытку ТСП проехать по опасному участку без ТСС, если сопровождение необходимо;
2. за несоответствие количества разгружаемых и загружаемых ТСП при осуществлении перегрузочных работ;
3. за превышение максимально допустимого числа ТСП в кортеже;

4. за превышение максимального числа ТСП, которые могут одновременно загружаться/разгружаться на терминале;
5. за попытку загрузить ТСП на пустом складе;
6. за нарушение установленной очередности при загрузке/разгрузке и т.д.

Выбор наилучшего расписания с помощью генетического алгоритма осуществляется по следующей схеме.

В соответствии с заданными правилами, необходимыми для исключения заведомо недопустимых случаев, а также с учетом ретроспективной статистической информации генерируется начальная популяция (особи в популяции – расписания всего производственного комплекса, гены – расписания отдельных производственных единиц). Далее путем скрещивания (т.е. обмена генами) и добавления случайных мутаций формируется следующее поколение. В зависимости от значений

1. величин, характеризующих выполненную полезную работу (например, стоимость/объем перевезенной и переданной на продажу либо в дальнейший технологический процесс продукции за весь период планирования);
2. затрат на выполнение работ производственными единицами (например, стоимости топлива, необходимого для проезда заданного участка маршрута);
3. введенных выше штрафов

вычисляется оценка каждой новой особи. Далее выбираются особи с наилучшими оценками, после чего повторяется процедура формирования нового поколения.

Основные достоинства приведенного метода состоят в его быстроедействии и возможности относительно несложной доработки для учета новых технологических, временных, экономических и т.д. ограничений (путем модификации технологических карт и системы штрафов).

Основной недостаток заключается в том, что описанный алгоритм может разойтись, либо сойтись к локальному экстремуму, не являющемуся глобальным.

Заключение

В настоящей работе приведено описание трех методологий решения задачи формирования оптимальных расписаний перевозок однородной продукции между оптовыми складами при наличии ограничений различных типов. В основе данных методологий – три различных способа построения модели объекта исследования, учитывающие специфику изучаемой системы, позволяющие избежать излишней искусственной дискретизации времени, (и, тем самым, резкого роста размерности задачи при расширении горизонта планирования либо уменьшении времени выполнения самой быстрой технологической операции), а также допускающие относительно несложную

адаптацию при ряде возможных изменений схемы работы производственного комплекса.

Более того, совместное применение предложенных подходов может преодолеть недостатки, присущие отдельным методологиям. Например, метод модельных событий позволяет относительно быстро построить набор допустимых расписаний и выбрать из них наилучшее. Значение целевой функции для сформированного расписания можно применить в подходе, основанном на использовании имитационного моделирования и метода ветвей и границ, для отсечения ветвей, заведомо не приводящих к увеличению целевой функции, и тем самым существенно сократить время работы, необходимое для поиска глобального решения с помощью данного алгоритма. Информация о рассчитанном точном решении задачи, в свою очередь, может быть применена для улучшения сходимости метода, основанного на построении технологических карт и использовании генетического алгоритма, при решении семейства задач, получающихся из уже решенной задачи не очень значительными изменениями исходных данных.

Дальнейшее направление исследований состоит в применении построенных методологий для анализа задач оптимизации расписаний перевозок, возникающих в реальных производственных системах.

Библиографический список

1. Лазарев А.А., Гафаров Е.Р. Теория расписаний. Задачи и алгоритмы. М.: МГУ, 2011. – 222 с.
2. Гладков Л.А., Гладкова Н.В. Особенности и новые подходы к решению динамических транспортных задач с ограничением по времени // Известия ЮФУ. Технические науки. 2014. № 7 (156). С. 178–187.
3. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982. – 416 с.
4. Ржевский Г.А., Скобелев П.О. Как управлять сложными системами? Мультиагентные технологии для создания интеллектуальных систем управления предприятиями. Самара: ООО «Офорт», 2015. – 290 с.
5. Куренных А.Е., Судаков В.А. Поддержка принятия решений на основе имитационного моделирования // Научно-технический вестник информационных технологий, механики и оптики. 2017. Т. 17. № 2. С. 348–353.
6. Осипов В.П., Судаков В.А. Комбинированный метод поддержки принятия многокритериальных решений // Препринты ИПМ им. М.В.Келдыша. 2015. № 30. 21 с. URL: <http://www.keldysh.ru/papers/2015/preprint.asp?id=2015-30>
7. Осипов В.П., Судаков В.А. Многокритериальный анализ решений при нечетких областях предпочтений // Препринты ИПМ им. М.В.Келдыша. 2017. № 6. 16 с. URL: <http://library.keldysh.ru/preprint.asp?id=2017-6>

Оглавление

Введение	3
Общая постановка задачи	4
О возможных подходах к решению задачи	6
Алгоритм, основанный на комбинированном применении имитационного моделирования и метода ветвей и границ	11
Поиск рациональных решений методом модельных событий (ММС).....	16
Методология, основанная на формировании технологических карт и применении генетического алгоритма	20
Заключение.....	24
Библиографический список.....	25