

Такмазьян А.К., ООО «Програмпарк», г. Москва  
Шелудяков А.В., ОАО «НИИАС», г. Москва

## Мультиагентное решение методом аукционов многопродуктовой транспортной задачи с объединенными потребностями

### Введение

При планировании ремонтов тягового подвижного состава на железной дороге расчет прогнозной потребности в ремонтах производится на основе среднесуточного пробега локомотива, вычисленного для заданной серии для каждого эксплуатационного локомотивного депо (ТЧЭ). Например, на ДРЛ-1 Восточного полигона, к ТЧЭ «Нижеудинск» приписаны локомотивы серий ТЭМ2, ВЛ60К, ВЛ60ПК, ВЛ65, ВЛ80К, ВЛ80Р, ВЛ80Т, ВЛ85. Или к ТЧЭ «Иланская» приписаны локомотивы серий ВЛ60К, ТЭМ2А, ВЛ60ПК, ТЭМ2У, ТЭМ2К, ТЭМ2, 2ТЭ10У, 2ТЭ10В, ВЛ80Т, 2ТЭ10М, ВЛ80Р, ВЛ80С, ТЭМ2УМ. Для каждой серии известен средний пробег внутри каждого из депо, следовательно, можно сделать прогноз пробега локомотива на любую дату в будущем, что позволяет, при известных нормах межремонтных пробегов, прогнозировать количество ремонтов разной величины для каждой секции внутри депо.

В то же время мощности ремонтных предприятий (ремонтных депо или заводов) позволяют варьировать загрузку этих предприятий по типам серий внутри определенных групп. Например, Улан-Удэнский электровозоремонтный завод (ЭРЗ) может проводить в год капитальный ремонт 520 секций локомотивов любой из серий ВЛ85, ВЛ80С, ВЛ80Р, ВЛ80ТК, ВЛ80СК, 2ЭС5К, 3ЭС5К, причем количество ремонтов каждой серии может быть произвольным, при ограничении на сумму всех ремонтов.

При долгосрочном планировании ремонтов прогноз отправной точки при передислокации локомотива (секции) на ремонтное предприятие производится на основании депо приписки локомотива. Таким образом, процесс планирования формализуется с помощью транспортной задачи на графе железнодорожной сети, где истоками служат станции расположения эксплуатационных депо, стоками – станции расположения локомотиворемонтных заводов (для заводских ремонтов), и станции расположения локомотиворемонтных депо (для деповских ремонтов). Ценами ребер графа могут служить расстояния, либо времена следования по кратчайшему допустимому маршруту сети.

Транспортная задача в данной постановке – это многопродуктовая транспортная задача, мощности стоков в которой объединены по некоторым группам продуктов. Такая задача не сводится к связанной по-

следовательности однопродуктовых транспортных задач, и требует специальных усложненных методов решения.

В работе [1] был предложен эффективный алгоритм решения транспортной задачи методом аукционов, позволяющий реализацию в распределенных системах (например, в мультиагентной парадигме). Особенность предложенного метода в том, что он не требует какой-либо модификации при появлении дополнительных ограничений на решения, например, таких как в описанной выше задаче. В методе достаточно просто оставить в двудольном графе задачи только те ребра, которые ведут от истоков к стокам, имеющим среди своих мощностей указанную серию истока, и удалить остальные ребра, не соответствующие возможным размещениям ресурсов. В процессе мультиагентных аукционов заявки-ставки будут рассылаться только вдоль существующих ребер, и, таким образом, ограничения на тип ресурсов будут соблюдены без введения типов ресурсов, никак не влияя на сам метод.

В докладе представлена реализация метода работы [1] на языке AgentSpeak в среде Jason [2], выполненная в рамках разработки программного обеспечения для автоматизированной подсистемы «Управления тяговым хозяйством на Восточном полигоне». В качестве метода определения завершения работы использован алгоритм Хуана.

### Постановка задачи

Определить потоки  $f_{ij}$ , максимизирующие суммарную полезность (минимизирующие суммарную стоимость) перевозки разнотипных сущностей от  $N$  поставщиков с мощностями  $\alpha_{ik}$  к  $M$  потребителям с мощностями  $\beta_j$ , при заданной полезности (стоимости) перевозки  $u_{ij}$  одной единицы от поставщика  $i$  к потребителю  $j$ , варьируя потоки

$$\sum_{i=1}^N \sum_{j=1}^M \sum_{k=1}^{L_i} u_{ij} f_{ijk} \rightarrow \max$$

при ограничениях  $\sum_{j=1}^M f_{ijk} = \alpha_{ik}, \quad i = 1, \dots, N$

$$\sum_{k=1}^{L_i} \sum_{i=1}^N f_{ijk} = \beta_j, \quad j = 1, \dots, M \quad 0 \leq f_{ij}$$

При заданных  $u_{ij}, \alpha_{ik}, \beta_j$  определить  $f_{ijk}$ .

### Мультиагентная структура

Каждому поставщику и каждому потребителю ставится в соответствие интеллектуальный агент, отвечающий за хранение внутренней для данного агента информации и информационный обмен с другими агентами. Кроме того, вводится управляющий агент, координирующий работу остальных агентов. Агенты подразделяются на тех, кто назначает ставки и на тех, кто принимает и обслуживает эти заявки-ставки. Далее используются термины агент-заявитель (являющийся проактивным агентом) и агент-респондент (реактивный агент).

### Метод аукционов

#### Инициализация

Каждый респондент ( $j$ ) в каждый момент времени содержит набор пар (поток-цена) –  $(f_{ij}, p_{ij})$ , где  $i = 0, \dots, M$ , в котором содержатся распределенные уже потоки по заявителям, вместе со значением цены, которая была назначена при данном распределении. В начальный момент времени в наборе есть только одна пара  $(\beta_j, p_{0j})$ , состоящая из нулевой цены и полной вместимости респондента. При любом изменении (а также начальном возникновении) этого набора респондент рассылает всем заявителям его обновленный вариант, ассоциированный с собственным идентификатором (то есть подписанный).

#### Формирование заявок

Каждый заявитель ( $i$ ) при получении любого такого набора  $\{(f_{kj}, p_{kj})\}$  от любого респондента, прежде всего, проверяет наличие нераспределенных единиц своей мощности. При их отсутствии заявитель не производит никаких дальнейших действий. Если какие-то единицы еще не распределены по респондентам, инициируется процесс создания и рассылки заявок всем респондентам. В начале этого процесса из набора  $\{(f_{kj}, p_{kj})\}$  исключаются пары, принадлежащие данному заявителю. После чего, из оставшегося набора формируется список пар  $L = \{(u_{ij} - p_{kj}, f_{kj})\}$ , упорядоченный по первой компоненте, смысл которой – выигрыш данного заявителя от размещения собственных единиц у данного респондента. Потом из начала сформированного списка выбирается такая минимальная последовательность  $S$ , суммарные потоки которой,

$$\sum_{l=1}^m f_{k_l j_l},$$

превышают нераспределенное количество данного заявителя. Далее формируются новые значения потоков и цен для новой заявки. Потоки для респондентов, не включенных в сформированную последовательность  $S$ , не изменяются:  $\hat{f}_{ij} = f_{ij}$ ,  $j \neq j_1, \dots, j_m$ . Потоки для всех респондентов из сформированной последовательности  $S$ , кроме последнего, формируются так, чтобы увеличить их на все

потоки данного респондента, включенные в последовательность  $S$ . Наконец, последний поток вычисляется так, чтобы полностью распределить всю мощность заявителя.

После того, как сформированы потоки, для них формируются значения заявочной цены, таким образом, чтобы перебить существующую цену. Для этого заявитель вычисляет собственное значение «выгоды при отказе»  $w_i$  (это значение равняется выгоде заявителя при получении отказа от респондента и размещении его заявки у ближайшего по выгодности «свободного» респондента). После чего каждому запрашиваемому потоку  $\hat{f}_{ij}$  ставится в соответствие предлагаемая за него новая цена  $b_{ij} = u_{ij} - w_i + \varepsilon$ , где  $\varepsilon$  – малая положительная константа, на которую будет превышена цена «максимального выигрыша» заявителя.

В результате формируется список заявок  $B(i) = \{(\hat{f}_{ij}, b_{ij})\}$ , которые рассылаются респондентам ( $j$ ) из этого списка с подписью агента-заявителя.

#### Обработка заявок

Каждый респондент ( $j$ ) при получении любой заявки инициирует процесс выбора заявок для распределения своей мощности по заявителям. Итак, пусть  $P(j) = \{(\hat{f}_{ij}, b_{ij})\}$  – список заявок, полученных респондентом, упорядоченный по убыванию предложенной цены. Выберем из начала этого списка подряд список  $P_1(j)$ , следующим образом: если суммарный поток заявок не превышает мощность респондента, то  $P_1(j) = P(j)$ ; иначе,  $P_1(j)$  – минимальный головной подсписок  $P(j)$ , суммарный поток которого полностью покрывает мощность респондента. Новые потоки для респондента формируются следующим образом: все потоки членов из списка  $P_1(j)$ , кроме последнего, полностью размещаются у респондента по заявленным ценам. Последний поток ограничивается сверху остаточной мощностью респондента, с ценой, соответственно заявке. Потоки, не вошедшие в  $P_1(j)$ , не размещаются.

После чего, весь процесс, описанный в данном разделе, повторяется: респондент рассылает всем заявителям обновленную информацию о своих потоках и ценах за них, и т.д., до тех пор, пока у заявителей (суммарная мощность которых, по построению, меньше суммарной мощности респондентов) не останется нераспределенных единиц мощности. В работе [1] показано, что каждый шаг данного процесса улучшает суммарную полезность потоков системы, и, следовательно, алгоритм сходится.

#### Упорядочение обработки заявок агентами

Агенты-респонденты при получении заявок на обработку и агенты-заявители, при получении обновленных потоков, помещают сигнал о них в собственную очередь сигналов, которые затем обрабатываются последовательно.

**Метод Хуана для определения завершения работы алгоритма.**

Для определения фазы завершения работы была использована идея метода весов (Huang's algorithm). Набор переменных величин, хранимый каждым агентом, дополняется еще одним значением – текущим весом агента. Управляющий агент в начале работы содержит в себе вес всей системы, а вес остальных агентов равен нулю. Основная идея состоит в делегировании агентами друг другу, вместе с запросом на выполнение какой-либо деятельности, некоторой доли своего веса. При завершении любым агентом своей работы, его текущий вес добавляется к весу агента-диспетчера, после чего обнуляется. Таким образом, равенство веса агента нулю означает, что он в данный момент находится в состоянии ожидания. Поскольку при описанной процедуре суммарный вес всех агентов не меняется, то признаком завершения работы всего алгоритма будет ситуация, когда вес агента-диспетчера станет равным его начальному значению: это будет означать, что вес всех остальных агентов равен нулю и деятельность по поиску решения завершена.

В методе аукционов алгоритм Хуана применяется следующим образом: в начале агент-диспетчер, после порождения и инициализации агентов-заявителей и агентов-респондентов, рассылает заявителям команду к началу аукциона, присоединяя к ней долю своего

веса, обратно пропорциональную количеству заявителей плюс константа.

Агенты-заявители, при рассылке заявок, присоединяют к ним долю своего веса, обратно пропорциональную количеству респондентов плюс константа.

Агент-респондент, после обработки заявок и формирования и рассылки новых потоков всем заявителям, формирует список заявителей, которым отказано в размещении всей или части их заявки. Заявителям из сформированного списка рассылается дополнительное сообщение об отказе, вместе с долей своего веса респондента, обратно пропорциональной количеству заявителей плюс константа.

Обнуление и отсылка своего веса диспетчеру у заявителя происходит при отсутствии неразмещенных единиц своей мощности, а у респондента, при отсутствии необработанных заявок в очереди поступивших сообщений.

**Литература**

1. D.P. Bertsekas & D.A. Castanon. «The auction algorithm for the transportation problem.» *Annals of Operations Research* 20(1989), pp.67-96.
2. R. H. Bordini, J. F. Hübner, M. Wooldridge. «Programming Multi-Agent Systems in AgentSpeak using Jason». Wiley. 2007. 268 pp.

