РЕШЕНИЕ СПЕЦИАЛЬНОЙ VRP ЗАДАЧИ РАСПРЕДЕЛЕНИЯГРУЗОВЫХ РЕСУРСОВ ТРАНСПОРТНЫХ КОМПАНИЙ, РАБОТАЮЩИХ ПО МОДЕЛИ FTL,МЕТОДАМИ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

© 2015 г. А.Н. Лада[1]
[1]Самарский Государственный Технический университет

Аннотация
Рассматривается специальная VRPзадача распределениятранспортныхресурсов в транспортных компаниях, развозящих грузы по бизнес модели FTL. Из практического опыта следует, что любая транспортнаякомпания работает в реальном времени и должна реагировать на входящие события, адаптивно перераспределяя имеющиеся ресурсы. Для данных целей хорошо зарекомендовали себя мультиагентные системы, которые используются во многих современных транспортных компаниях. Показана возможность улучшениякачества расписаний вночной период времени, когда никаких новых событий в рамках системы не происходит,за счет использованияклассическихметодовоптимизации в специальной задаче нахождения начального плана распределения ресурсов. В результате длительной работы с экспертамитранспортно-логистическихпредприятий сформирован необходимый набор ограничений длячастной задачи нахождения начального плана назначения транспортных ресурсов. Задача сформулирована в терминах классической задачи о назначениях линейного программирования. Рассмотрены ациклическиеи циклическиеклассы данной задачи. Показано, что для ациклическихклассовзадача может быть легко сведена к задаче о назначениях, для сведения циклическихклассовтребуется исключить условие соответствия заказовресурсам. Поиск решенияспециальной задачи построения начального плана предложено выполнить с помощьювенгерского алгоритма, который хорошо зарекомендовал себя как точный и быстрый метод для решения задачи о назначениях. Показано также, что этот способ не может быть применен в случае планирования в реальном времени, так как даже для статического циклическогокласса задачи невозможно учестьсоответствие ресурсаи заказадля будущих заказов, однако данный метод может быть применен в качестве дополнения к мультиагентному подходу для улучшения его эффективности на этапе построения первоначального плана распределения ресурсов.

*Ключевыеслова:*транспортнаялогистика, VRP, FTL, линейное программирование, задача о назначениях,венгерский алгоритм,мультиагентные технологии

1. Введение

Проблема оптимизациигрузоперевозок (VehicleRoutingProblem, VRP), впервые описанная в работе [1], является одной из актуальных и значимых задач современной теории оптимизации. Классификация оптимизационных задач транспортной логистики приведена в [2, 3]. Обзор и классификация задач VRPс предлагаемыми методами решения даетсятакже в [4].В настоящей работе формализуется и предлагается к решению специальная VRPзадача распределения грузовиков по заказам для крупных транспортных компаний, имеющих в своем управлении флот дальнемагистральных грузовиковболее 30 единиц. Такие транспортные компании,широко распространенные в странах с большой территорией и протяженностью автомобильных дорог (Россия, США, Канада и др.), осуществляют межрегиональные перевозки по схеме FTL (FullTruckLoad). Особенностью FTL-перевозок является наличие прямых контрактов с заказчиками на резервирование грузовика целиком, чтоисключает необходимость учитывать объем груза и строить консолидированные маршруты. Такоеупрощение мотивирует получениеболее точного метода решения задачи без использования эвристик. Различные модели организации грузоперевозок по схеме FTL приводятся в [5]. В данной работе дополнительно учитываются временные окна прибытия грузовика на погрузку, что приводит к разновидности задачи VRPTW(VehicleRoutingProblemwithTimeWindows),которая рассматривается в[6]. Дополнительно учитываютсяограничения на тип прицепа (наличие в нем дополнительного оборудования:холодильной установки, обрешётки для перевозки шин, подъемных приспособлений и др.). Ограничения на максимальную длину рейса не рассматриваются, потому что крупные автотранспортные компании могут позволить себе адаптивно менять водителей походу рейса, доставляя их к грузовикам другими видами транспорта, например, самолетом. Требуется «выбирать» из имеющегося набора заказов наиболее выгодные с точки зрения продолжительностирейса, т.к. в общем случае число заказов превосходит число грузовиков и после выполнения заказа грузовик не возвращается на базу, а продолжает движение к новому заказу с предыдущего места выгрузки до тех пор, пока не получит заказ с разгрузкойрядом с базой. Т.е. в отличие от большинства стандартных задач VRP, где возвращение на базу является обязательной конечной точкоймаршрута, в данной постановкеэто условие не является жестко заданным, а формируется динамически в процессе решения задачи. Вдальнейшемнеобходимо учитывать фактор «реального времени», когда после получения решения исходные данные начинают изменяться во времени, поступают новые заказы, происходят задержки при выполнении ранее запланированных заказов, грузовики становятся недоступными для планирования из-за проведения плановых и внеплановых работ по их техническому обслуживанию. При решении задачи нахождения распределения заказов по грузовикамв реальном времени заказы и ресурсы представляются как сеть потребностей (заказы) и возможностей (ресурсы) [10]. Считается, что всегда можно выделить достаточно длительный промежуток времени (более часа), когда эта сеть остается неизменной (ни один новый заказ не добавляется, ни один из параметров существующего заказа не изменяется, ни один новый грузовик не появляется и ни один из существующих грузовиков не исчезает). На практике это происходит потому, что в конце рабочего дня диспетчеры транспортной компании завершают работу и фиксируют для исполнения на следующий день часть плана, в котором некоторыезаказы фиксируется на грузовики, а другиеостаются незафиксированными, т.к. для принятия решения по ним еще остается время. В течение ночи до начала следующего рабочего дня есть возможность распределить их наиболее оптимально. Поэтому глобальная задача управления работой транспортной компании делится на двечасти:

- Построение начального опорного планапо имеющимся на начальный момент времени заказам с учетом базового набора основных критериев.
- Модификацияначального плана по событиям реального времени, когда необходимо принимать решения по ситуации с учетом расширенного набора критериев, которые плохо формализуются и не поддерживаются стандартными методами.

Для решения второй части задачи хорошо зарекомендовали себя мультиагентные технологии [7]. Известныхорошие результаты их примененияна реальных транспортных предприятиях, описанные, например,в [8] и [9].В данной статьебудет сделан акцент на разработку метода построения начального опорного плана распределения заказов по грузовикамс учетом основных, наиболее значимых факторов, влияющих на это распределение,которые используют руководители логистических отделов при построении подобныхпланов вручную,а такжеих формализацииив виде математических ограничений. В результате внедрения мультиагентной системы [8], путем анализа и обобщения накопленных сотрудниками транспортных компаний знаний был определен набор практических ограничений, который можно использовать для математической постановки задачи линейного программирования:

1. Момент времени прибытия заказа на место погрузки, рассчитанный как время высвобождения грузовика плюс значение длительности порожнего переезда, должно быть меньше, чем крайнее значение, соответствующее правой границе окна погрузки этого заказа, т.е приехать раньше допустимо, а вот опоздать не допустимо.
2. Порожний переезд до места погрузки заказа должен быть меньше 500 км, с учетом того, что средняя скорость движения грузовика полагается равной 50км/ч, длительность порожнего переезда не должна превышать 10 часов.
3. Значение длительности простоя, котораярассчитывается как начальное значение, соответствующее левой границе окна погрузки заказа минус время доставки этого заказа на место погрузки любым грузовиком,должно быть меньше 24 часов, т.е. если грузовик успевает доехать к пункту погрузки, но дополнительно вынужден простаивать больше суток,такое назначение не допустимо.

2. Постановка задачи

Пусть имеем набор заказов $O_i, i = 1, N$,каждый заказхарактеризуется пунктоми временным окном прибытия на погрузкуи разгрузку$[TOs_i;\ TOf_i]$, когда этот пункт доступен. Имеем также наборресурсов, представляющих собой грузовики с прицепами$R_j, j = 1, M$, каждый из которых характеризуетсяпунктомначальногоместонахожденияи временем высвобождения из этого пунктаTRf_j, которое соответствуетвремении местуразгрузки предыдущеговыполненного им заказа или базе.Для любого грузовика R_j известна длительность порожнего переезда D_{ij} к любому заказу O_i.Под каждый заказ O_iтребуется отдельный грузовик с прицепомR_j, удовлетворяющий ограничениям на тип прицепа, т.е. грузовик R_jможет как подходить, так и не подходить заказу

O_i.Все заказы считаются равноправными и от назначения любого заказа на грузовик можно отказаться без каких-либо штрафов со стороны заказчика (на практике эти заказы будут перепроданы другой внешней транспортной компании 3PL).Нужно найти такое назначение всех Mресурсов на заказы, при котором суммарный порожний переездбудет минимальным, при максимальном числе назначенных заказовQ и выполнении условий допустимости назначения:

$\sum_{i,j} D_{ij} \to \min, Q \to N$	(1)
$\begin{cases} \mathrm{TRf_j} + \mathrm{D_{ij}} < \mathrm{TO_{fi}} \\ \mathrm{D_{ij}} < 10 \\ \mathrm{TOs_i} - \mathrm{TR_{fj}} - \mathrm{D_{ij}} < 24 \end{cases}$	(2)

3. Метод решения задачи

Для решения поставленную задачу предлагается разделить на две части. В первой части выполняется построение матрицы допустимых назначений, для того чтобы определить пространство решений, удовлетворяющих заданным ограничениям задачи. Во второй части осуществляетсяприведение данной матрицы к виду классической задачи о назначениях, гдепоиск оптимального назначения выполняется одним из методов линейного программирования.

3.1. Построение матрицы допустимых назначений

Для задачи строится матрица допустимыхназначений, в которой строкисоответствуютзаказамO_i, а столбцыресурсам R_j. В ячейку матрицы,соответствующуюназначению O_iR_j, записывается длительность порожнего переездаD_{ij}из пункта нахождения грузовика R_j и времениего высвобождения TRf_j к пункту погрузки заказаO_i, при условии, что грузовик R_jподходит заказу O_iи выполняется система неравенств (2), в противном случае ячейка остается пустой.

Для наглядности рассмотримпримеры построения матрицы допустимых назначений для частной (ацикличной) и общей (цикличной) задачи.

3.1.1. Пример построение матрицы допустимых назначений в ацикличной задаче

Наборзаказовсихпунктомпогрузкиивременнымиокнамиотносительноначального времениT_0=0 даны в таблице 1.

	Погрузка	TOs	TOf
O_1	Москва	2	4
O_2	Самара	18	22
O_3	Екатеринбург	38	40

Таблица1: набор заказов

Наборресурсовсихначальнымместонахождениемивременем высвобождения относительноT_0=0 даны в таблице 2.

	Местонахождение	TRf
R_1	Москва	1
R_2	Самара	6
R_3	Екатеринбург	12

Таблица2: набор ресурсов

Длительностьпереездамеждулокациями (в часах)данав таблице 3.

	Москва	Самара	Екатеринбург
Москва	1	13	24
Самара	13	1	9
Екатеринбург	24	9	1

Таблица3: длительность переезда между локациями

Считается, что каждый грузовик R_jподходит заказу O_i, также полагается, что длительность выполнения всех заказов заведомо превосходит время начала погрузки самого позднего заказа, в этом смысл ацикличной задачи, т.е. ни один из грузовиков не успеет выполнить более одного заказа. Для каждого ресурсаR_jпроверяетсяпотенциальная возможность назначения на заказO_iсогласно системе неравенств (2):

$O_1R_1= \begin{cases} 1+1<4 \\ 1<10 \\ 2-1-1<24 \end{cases}$	$O_1R_2= \begin{cases} 6+13<4 \\ 13<10 \\ 2-6-13<24 \end{cases}$	$O_1R_3= \begin{cases} 12+24<4 \\ 24<10 \\ 2-12-24<24 \end{cases}$
$O_2R_1= \begin{cases} 1+13<22 \\ 13<10 \\ 18-1-13<24 \end{cases}$	$O_2R_2= \begin{cases} 6+1<22 \\ 1<10 \\ 18-6-1<24 \end{cases}$	$O_2R_3= \begin{cases} 12+9<22 \\ 9<10 \\ 18-12-9<24 \end{cases}$
$O_3R_1= \begin{cases} 1+24<40 \\ 24<10 \\ 38-1-24<24 \end{cases}$	$O_3R_2= \begin{cases} 6+9<40 \\ 9<10 \\ 38-6-9<24 \end{cases}$	$O_3R_3= \begin{cases} 12+1<40 \\ 1<10 \\ 38-12-1<24 \end{cases}$

Неравенства выполняются для назначений: O_1R_1; O_2R_2; O_2R_3; O_3R_2. Матрица допустимых назначений для задачи данав таблице 4.

	R_1	R_2	R_3
O_1	1		
O_2		1	9
O_3		9	

Таблица4: матрица допустимых назначений для ацикличной задачи

3.1.2. Пример построение матрицы допустимых назначений в цикличной задаче

В предыдущем примере считалось, что ни один из ресурсовне успевает выполнить более одного заказаиз-за того, что интервалывремен погрузок заказов [*TOs; TOf*]распределены плотно, а длительность выполнения любого заказа любым грузовикомвсегда ихпревосходит.Рассмотрим общий случай, когда заказыизвестны на широкий горизонт в будущем (например, регулярные заказы от постоянных клиентов, как правило, известны с хорошей точностью на неделю или даже месяц вперед). Поэтому у каждого ресурса появляется возможность выполнить более поздние по времени заказы после выполнения более ранних. Нужно понимать, что при такой постановке местоположениеи время высвобождения каждого ресурсабудут меняться по ходу решения задачи и матрица допустимых назначений будет иметь другой вид:

Наборзаказовсихпунктомпогрузки и разгрузкисвременнымиокнамиотносительноначального времениT_0=0 даны в таблице 5.

	Погрузка	Разгрузка	*TOs*	*TOf*
O_1	Москва	Самара	1	2
O_2	Самара	Екатеринбург	23	28
O_3	Екатеринбург	Москва	50	52

Таблица5: набор заказов

Набор ресурсовсихначальнымместонахождениемивременем высвобождения относительноT_0=0 даны в таблице6.

	Местонахождение	*TRf*
R_1	Москва	0
R_2	Самара	0
R_3	Екатеринбург	0

Таблица6: набор ресурсов

Длительностьпереездамеждулокациями(в часах) данав таблице 3. Для каждого ресурсаR_jпроверяется возможность его назначения на заказ O_iпо системе неравенств (2).Если назначение может быть выполнено, рассматривается возможность дальнейшего назначения на оставшиеся заказы согласно системе неравенств (2) с учетом сменившегося местоположения ресурсаR_j:

$O_1R_1= \begin{cases} 0+1<2 \\ 1<10 \\ 1-0-1<24 \end{cases}$ =>	$O_1R_1O_2= \begin{cases} 14+1<28 \\ 1<10 \\ 23-14-1<24 \end{cases}$ =>	$O_1R_1O_2R_1O_3= \begin{cases} (23+9)+1<52 \\ 1<10 \\ 50-23-9-1<24 \end{cases}$
$O_1R_2= \begin{cases} 0+13<2 \\ 13<10 \\ 1-0-13<24 \end{cases}$	$O_1R_3= \begin{cases} 0+24<2 \\ 24<10 \\ 1-0-24<24 \end{cases}$	$O_2R_1= \begin{cases} 0+13<28 \\ 13<10 \\ 23-0-13<24 \end{cases}$
$O_2R_2= \begin{cases} 0+1<28 \\ 1<10 \\ 23-0-1<24 \end{cases}$ =>	$O_2R_2O_3= \begin{cases} (23+9)+1<52 \\ 1<10 \\ 50-23-9-1<24 \end{cases}$	

$O_2R_3 = \begin{cases} 0+9<28 \\ 9<10 \\ 23-0-9<24 \end{cases}$ =>	$O_2R_3O_3 = \begin{cases} (23+9)+1<52 \\ 1<10 \\ 50-23-9-1<24 \end{cases}$	
$O_3R_1 = \begin{cases} 0+24<52 \\ 24<10 \\ 50-0-24<24 \end{cases}$	$O_3R_2 = \begin{cases} 0+9<52 \\ 9<10 \\ 50-0-9<24 \end{cases}$	$O_3R_3 = \begin{cases} 0+1<52 \\ 1<10 \\ 50-0-1<24 \end{cases}$

Получится матрица допустимых назначений,приведенная в таблице 7.В колонках матрицыO_1R_1; $O_1R_1O_2R_1$; O_2R_2; O_2R_3 представлены местоположения ресурсовR_1, R_2 и R_3 после возможного выполнения ими заказов O_1, O_1 затем O_2 и O_2 соответственно.

	R_1	R_2	R_3	O_1R_1	$O_1R_1O_2R_1$	O_2R_2	O_2R_3
O_1	1						
O_2		1	9	1			
O_3					1	1	1

Таблица7: матрица допустимых назначений для цикличной задачи

3.2. Поиск оптимальных назначений

После построенияматрицы допустимых назначенийможно полагать, что задача нахождения опорного плана сводится к задаче поиска такого назначенияресурсовна заказы в этой матрице, при котором суммарное время порожнего переездадля всех ресурсов по всем назначенным заказам будет минимальным при максимальном числе назначенных заказов (1).

Заметим, что полученные в приведенных выше примерах матрицы допустимых назначений аналогичны матрицам, с помощью которых формализуетсястандартная задача линейного программирования - задачао назначениях.Как показано в [12], в ряде частных случаев (ацикличная задача) для задачи о назначенияхможно получить точное решение, что и было целью данной работы. Известно, что задача о назначениях полиномиально разрешима,традиционное решение (венгерский алгоритм[13])имеетасимптотическуюсложностьO(n3),этого вполне достаточно для решения такой задачидаже при большойразмерности матрицыдопустимых назначений в реальных задачах транспортных компаний.

Сформулируем задачу о назначениях в терминах линейного программирования. Пусть O – множество заказов, содержащее Nэлементов, а Rмножество ресурсов, содержащее Mэлементов. Переменная x_{ij} представляет назначение O_i на R_j, принимая значение 1, если ресурс R_j назначен на заказ O_i , и 0 в противном случае.Введем$D(i,j)$ - длительность порожнего переезда ресурсаR_jк заказуO_i. Целевая функция и ограничения для задачи будут иметь вид:

$\sum_{i\in O, j\in R} D(i,j)x_{ij}$	(3)
$\sum_{j\in R} x_{ij} = 1, i \in O$	(4)
$\sum_{i\in O} x_{ij} = 1, j \in R$	(5)
$x_{ij} \geq 0, i,j \in O, R$	(6)

В зависимости от числа Nи Mограничения(4) и (5) из равенств сменятся на неравенства, так, например, в случае если $M>N$, какие-то ресурсы останутся незанятыми, напротив, если $N>M$, часть заказов останется не назначенной.

Рассмотрим, как можно свести поставленную в данной работе задачу к задаче о назначениях и решить ее венгерским алгоритмом. Начнем с рассмотрения частного ацикличного класса этой задачи, описанного в примере 1, где длительность выполнения всех заданных заказов заведомо превосходит время начала погрузки самого позднего заказа, т.е. ни один из ресурсов не успеет выполнить более одного заказа. Этот класс задач просто сводится к задаче о назначениях, даже если положить, что не каждый ресурс в общем случае подходит любому заказу, посколькуэто условие можнопроверить при построениисамой матрицы. При рассмотрении более общего цикличного класса задачи, описанного в примере 2, где местоположениеи время высвобождения каждого грузовика будут меняться по ходу решения, свести его к задаче о назначениях также возможно, однако для этого необходимо пренебречь условием совместимости ресурсов и заказов,положив, что любой ресурс подходит любому заказу. Принимая данное допущение, для сведения к задаче о назначениях необходимо преобразовать матрицу допустимых назначений к новому виду, отказавшись от рассмотрения всего пути конкретного грузовика до конкретного заказа, при этом матрица из таблицы7 будет преобразована к виду таблицы 8.

	R_1	R_2	R_3	RO_1	RO_2	RO_3
O_1	1					
O_2		1	9	1		

O_3					1	

Таблица8: матрица для задачи о назначениях в цикличной постановке

Здесь в колонках матрицы, которые идут после конкретных ресурсовR_1, R_2 и R_3, стоят уже обезличенныеRO_1,RO_2 и RO_3, соответствующие точкам разгрузки заказов O_1,O_2 и O_3.При этомна этапе формирования матрицы нельзя определить, какой конкретно ресурс прибудет на погрузкуследующего заказа,это будет определено только по ходу решения задачи.Поэтомуневозможно учесть для этих случаев совместимость ресурсов с заказами, однако для первых колонок, в которых указаны конкретные ресурсы,эту проверку выполнить можно.Когда для этой матрицы будет решаться задача о назначениях, для всех назначений вида RO_iO_jбудет вычисленресурс, который назначен на предыдущий заказ O_i.Важно также отметить, что прежде чем решать матрицу допустимых назначенийвенгерским алгоритмом, необходимо исключить из нее пустые строки и пустые столбцы, а также заполнить пустые значения в оставшихся ячейках (решение в которых недопустимо) заведомо большими числами. Если в результате будет получено решение, в котором какой-либо из ресурсов назначен на недопустимый заказ, это назначение можно просто исключить, оставив заказ не назначенным, а ресурс свободным.

В результате решения задачи описанным выше методом, мы получим точное решение, однако только с допущением, что любой заказ подходит любому ресурсу. Исключитьэто допущение и получить точное решениене представляется возможным. Однако, как показывает практикадля работы промышленной мультиагентной системы планирования [8],в этом нет необходимости, поскольку полученный начальный план будет далее видоизменяться по событиям реального времени, где для решения будет применен мультиагентныйподход[7]. В частности, еслиагент заказа получилпервоначальное назначение на грузовик с прицепом,не подходящийему, он будет адаптивно перепланироватьсяна другой подходящий грузовик с учетом возможных изменений плана по событиям реального времени.

4. Результаты экспериментов

Мы провели несколько экспериментов, чтобы проверить описанные методы с использованием реальных данных от наших компаний-клиентов. Каждый из наших клиентовиспользует мультиагентную систему [8].Первая компания ProLogics [15] имеет 140 ресурсов и около 25 новых заказов в день.Вторая компания MONOPOLY [16] имеет 300 ресурсов и около 76 новых заказов в день.Третья компания LORRY [17] имеет 680 ресурсов и около 240 новых заказов в день.Целью экспериментов было сравнение мультиагентного метода, ориентированногона работу в режиме реального времени, с венгерскималгоритмом в задаче построения первоначального плана. В ходе экспериментов осуществлялся запуск мультиагентной системы [8] для набора данных каждого клиента.Сразу после того как первоначальный план был создан, система останавливалась. На основе тех же наборов данных формировалисьматрицы для задачи о назначениях,которые затем решалисьодной из реализаций венгерского алгоритма [18]. Для получения более развернутой картины, экспериментальные данные былисняты с реальных данныхтрех вышеперечисленных компаний в разное время в течение рабочего месяца. Они были сгруппированы по плотности (% пустых клеток) в матрице назначений, которая в общем случае составляет от 5% до 95%. Результаты экспериментов представлены в таблице 9. Можно увидеть, что венгерский алгоритм дает более эффективные результаты и потребляет меньше времени почти в каждом эксперименте. Эксперименты проводились на рабочей станции с процессором 3,4 ГГц IntelCorei7-4770 и с 8 Гб оперативной памяти, работающей под управлением Windows 8.1.

Матрица для задачи о назначениях			Мультиагентный метод		Венгерский метод	
N заказов	*M*ресурсов	% пустых клеток	KPI (1)	Время работы, сек	KPI (1)	Время работы, сек
25	140	5	35	0,014	34	0,001
25	140	15	33	0,016	32	0,001
25	140	30	43	0,012	41	0,001
25	140	50	49	0,009	47	0,001
25	140	70	73	0,005	68	0,001
25	140	85	147	0,004	144	0,001
25	140	95	341	0,003	339	0,001
76	300	5	82	0,6	77	0,001
76	300	15	86	0,5	81	0,001
76	300	30	93	0,4	92	0,003
76	300	50	111	0,3	106	0,001
76	300	70	127	0,2	123	0,001
76	300	85	208	0,12	199	0,003
76	300	95	483	0,052	472	0,001
240	680	5	242	39,91	240	0,011
240	680	15	240	34,99	240	0,014
240	680	30	245	29,91	241	0,01
240	680	50	255	21,8	246	0,01
240	680	70	285	13,29	273	0,008
240	680	85	431	6,97	414	0,007
240	680	95	980	2,66	934	0,006

Таблица9: результаты экспериментов

5. Заключение

В настоящей работе применяются методы линейного программирования для специальной VRP задачи построения начального расписания грузовиков по заказам в крупной транспортной компании, осуществляющей межрегиональные перевозки по схеме FTL. Предлагается метод поискаточного решения с оговоренным допущением и с учетом минимально необходимогонабора критериев, знания о которыхбыли получены на основе опыта внедрения промышленной мультиагентной системы [8]. Показаныограничения применимости данного метода при переходе к планированию в реальном времени, поскольку даже в случае его использования на статичной цикличной задаче нет возможности учесть соответствие заказов параметрам прицепов грузовиков. Таким образом, можно сделать вывод, что при переходе к планированию в реальном времени, когда всистемевозникнут возмущения, связанные с появлением новых заказов или отменой/изменением существующих заказов,применение только классических методов недостаточно, но комбинация классического и неклассического мультиагентного подхода даст хорошее решение, которое можно применить на практике.

Списоклитературы

1. Dantzig G.B., Ramser J.H. The Truck Dispatching Problem // Management Science. 1959. Vol. 6, No. 1. P. 80-91.
2. Michael Browne; Alan McKinnon; Sharon Cullinane. Green Logistics by Anthony Whiteing Published by Kogan Page, 2010
3. Bertazzi, L., Savelsbergh, M., & Speranza, M. G. (2008). Inventory Routing. In Bruce Golden, S. Raghavan, & E. Wasil (Eds.), The Vehicle Routing Problem: Latest Advances and New Challenges (Vol. 43, pp. 49–72). New York: Springer Science+Business Media. doi:10.1007/978-0-387-77778-8
4. The vehicle routing problem: state of the art classification and review De Jaegere N, Defraeye M, Van Nieuwenhuyse, KU Leuven. [online] Available at: <https://lirias.kuleuven.be/bitstream/123456789/457452/1/KBI_1415.pdf> [14 January 2015].
5. Oleg Granichin, Petr Skobelev, Alexander Lada, Igor Mayorov, Alexander Tsarev. Cargo transportation models analysis using multi-agent adaptive real-time truck scheduling system. – Proceedings of the 5th International Conference on Agents and Artificial Intelligence (ICAART'2013), February 15-18, 2013, Barcelona, Spain. – SciTePress, Portugal, 2013, Vol. 2. – pp. 244-249. ISBN 978-989-8565-39-6
6. Hideki Hashimoto, Toshihide Ibaraki, Shinji Imahori, Mutsunori Yagiura. The vehicle routing problem with flexible time windows and traveling times. DiscreteAppliedMathematics. Volume 154, Issue 16, 1 November 2006, Pages 2271–2290
7. П.О.Скобелев. Мультиагентные технологии в промышленных применениях: к 20-летию основания Самарской школы мультиагентных систем // Мехатроника, Автоматизация, Управление. 2010. №12 – с.33-46.
8. Иващенко А. В,. Лада А., Майоров И., Скобелев П., Царев А. Анализ эффективности применения мультиагентной системы управления региональными перевозками в реальном времени // Материалы 4-й мультиконференции по проблемам управления МКПУ-2011, 3-8 октября 2011 г., с.Дивноморское, Геленджик, Россия. Т.1. – Таганрог: Изд-во ТТИ ЮФУ, 2011.2011. С. 353-356. ISBN 978-5-8327-0404-3.
9. А.В. Вайсблат, А.Р. Диязитдинова, А.В. Иващенко, П.О. Скобелев, А.В. Царев. Организация интерактивного взаимодействия в мультиагентной системе управления транспортно-экспедиционной компанией // Труды XII Международной конференции «Проблемы управления и моделирования в сложных системах» Самара: Самарский научный центр РАН, 2010. – с. 620 – 628.
10. Виттих В.А., Мультиагентные модели взаимодействия для построения сетей потребностей и возможностей в открытых системах [Текст] / В.А. Виттих, П.О. Скобелев – М.: Автоматика и Телемеханика, 2003, №1, с. 177-185
11. Michal Maciejewski and Kai Nagel. Towards Multi-Agent Simulation of the Dynamic Vehicle Routing Problem in MATSim. 9thInternationalConference, PPAM 2011, Torun, Poland, September 11-14, 2011.
12. Мальковский Н.В., Граничин О.Н., Амелин К.С. Распределение ресурсов в контексте мультиагентных систем // В сб. трудов XII Всероссийское совещание по проблемам управления (ВСПУ-2014), Россия, Москва, ИПУ РАН, 16-19 июня 2014. 2014. С. 9003-9013.
13. Kuhn H.W. The Hungarian method for the assignment problem // Naval Research Logistics Quarterly. 1955. Vol. 2, No. 1-2. P. 83-97.
14. J. Munkres, «Algorithms for the Assignment and Transportation Problems», Journal of the Society for Industrial and Applied Mathematics, 5(1):32—38, 1957 March.
15. PROLOGICS. [online] Available at: <http://eng.prologics.ru/> [14 January 2015].
16. MONOPOLY. [online] Available at: <http://monopoly.su/> [14 January 2015].
17. LORRY. [online] Available at: <http://en.lorry-ural.ru/> [14 January 2015].
18. GitHub. [online] Available at: <https://github.com/catbaxter/algorithms/blob/master/hungarian/hungarian.cpp> [14 January 2015].