УДК 681.007.05

В.Д. Липанов, С.А. Кичкайло

Харьковский университет Воздушных Сил им. И .Кожедуба, Харьков

МОДЕЛИРОВАНИЕ И УСОВЕРШЕНСТВОВАНИЕ АЛГОРИТМА ФОРМИРОВАНИЯ ОПТИМАЛЬНЫХ МАРШРУТОВ ТРАНСПОРТНЫХ ПЕРЕВОЗОК

В результате проведенного моделирования и анализа известных методов формирования оптимальных маршрутов движения транспорта при организации транспортных перевозок предложен усовершенствованный алгоритм формирования оптимального маршрута с учетом возможностей перевозчика и потребностей клиентов.

Ключевые слова: *граф, груз, критерий, логистика, маршрут, метод ветвей и границ, шкала порядка, кратчайший путь, деревянный алгоритм, жадный алгоритм.*

Введение

Проблема совершенствования методов управления предприятиями производственной инфраструктуры становится с каждым днём всё актуальнее. Это относится и к транспортному обслуживанию (в том числе и к почтовым перевозкам), поскольку конкуренция в этой отрасли развивается высокими темпами. Конкурентные преимущества сегодня – повышение качества и снижение себестоимости перевозок, предоставление большего спектра услуг, улучшение обслуживания клиентуры, своевременное реагирование на изменение транспортных услуг на территории страны и региона. Особенностью современных грузоперевозок, к тому же, являются требования, вызванные международной кооперацией производства, расширением международных торговых связей, в том числе Internet – торговля. К таким требованиям, в частности, относятся жёсткие временные рамки отгрузки и доставки товаров (в том числе и почтовых отправлений), что существенно усложняет задачу организации и управления потоками перевозок. Все вышеизложенное обуславливает поиск новых подходов к работе на транспортном рынке. Одним из таких прогрессивных подходов является логистический подход, ориентированный на увеличение объемов перевозки грузов, повышение доходности и прибыльности работы транспорта.

Наиболее просто принципы логистики могут быть использованы при перевозке массовых грузов в условиях, когда сформировались стабильные и мощные грузопотоки между отправителями и получателями. Значительно сложнее структура и функции логистической системы, когда распределяются товары широкой номенклатуры, предназначенные для удовлетворения потребностей десятков, а то и сотен потребителей. При доставке такой многономенклатурной продукции появляется необходимость в применении более широкого использования развозочных и сборочных маршрутов различных транспортных средств (авиационного, автомобильного транспорта).

Решению некоторых задач данной проблемы и посвящена эта работа.

Основная часть

Программная реализация системы построения маршрутов

1. Математическая модель построения маршрутов

Имеется N городов T[0] .. T[N-1]. Расстояние между каждой парой T[i], T[j] определяется длиной соединяющего их отрезка

$$A_{ij} = A_{ji} = \sqrt{(T[i].X - T[j].X)^2 + (T[i].Y - T[j].Y)^2} \quad (1)$$

где A – матрица расстояний между городами. Необходимо указать кратчайший маршрут, который начинается городом T[0], проходит через города T[1] .. T[n-2] и заканчивается городом T[N-1].

В теоретическом плане задача решается легко: достаточно перебрать все перестановки городов T[1] .. T[n-2] на маршруте и выбрать ту из них, которая доставляет кратчайший путь. Однако этот метод при существующих возможностях ПК дает результат за приемлемое время вычислений (от нескольких секунд до минуты), если N<10 быстродействие комбинаторного метода быстро снижается и его нельзя использовать в практических расчетах. Среди других методов решения подобных практических задач (к ним, в частности, можно отнести близкую к рассматриваемой задачу коммивояжера) обычно используют единственный альтернативный метод ветвей и границ (МВГ). Считается, что он обеспечивает точное решение за минимальное время вычислений. Метод, действительно, хорошо работает на "учебных" примерах, однако, как показали эксперименты с МВГ на практических (логистических) примерах решения рассматриваемой задачи, его быстродействие сильно зависит от вида матрицы A и в большинстве случаев МВГ не гарантирует результативности в приемлемое время даже при N=15.

При всей известности задачи не удалось ни в научной литературе, ни в Интернет найти быстрых методов, которые позволили бы приближенно решить задачу с достаточной для практики точностью до 10% за приемлемое время.

Ниже рассмотрено несколько сравнительно быстрых приближенных эвристических методов решения задачи, которые удовлетворяют упомянутому условию. Методы реализуют процессы поиска базового маршрута и последующего его улучшения. При их описании использованы терминология теории графов и средства языка Object Pascal среды Delphi.

2. Методы нахождения базового маршрута

Метод 1.1 («жадный»). Сначала на графе, образованном матрицей A, отыскивается и включается в маршрут вершина (город) T[k] , которая ближе всех к начальной. Далее отыскивается самая близкая к T[k] из числа еще не включенных в маршрут и т. д. В результате получается приближенное решение задачи – базовый маршрут.

Метод 1.2 («деревянный»*)*. Сначала в маршрут включаются две вершины начальная T[0] и конечная T[N-1]. Далее отыскивается вершина, которая характеризуется наименьшим расстоянием D(T[i]+T[k]) = D(T[k]+T[j]) - D(T[i] + T[j]), где i = 0, j = N-1, k – номера еще не включенных в маршрут вершин. Найденная вершина помещается в маршрут (0, k, N-1). На следующем шаге отыскивается вершина L, которая характеризуется наименьшим расстоянием DL от звена (0, k), и вершина M, имеющая наименьшее расстояние DM от звена (k, N-1). Среди L и M выбирается та, которая имеет наименьшее из DL и DM, и включается внутрь своего звена (0, k) или (k, N-1). Пусть это вершина M с номером m. Теперь маршрут состоит из трех звеньев (0, k), (k, m), (m, N-1). Процесс продолжается до тех пор, пока есть не включенные в маршрут вершины.

Метод 1.3 (простейший*)*. Промежуточные вершины в маршрут включаются случайным образом. В частности, базовым будет допустимый маршрут G[i]=i Маршруты, построенные этими методами, вычисляются с очень высокой скоростью (практически мгновенно). Однако длина этих маршрутов в подавляющем большинстве случаев далека от практически приемлемой. Для этих целей применено несколько методов улучшения базового маршрута.

3. Методы улучшения базового маршрута

Метод 2.1 (перестановок). Совершается последовательный проход по парам соседних вершин всех звеньев с перестановкой этих вершин. Если перестановка уменьшает длину маршрута, то этот маршрут считается текущим. Производятся новые попытки улучшить его тем же методом до тех пор, пока перестановки не дадут эффекта. Далее аналогичным образом выполняются перестановки по трем соседним вершинам из числа тех, которые не попали в число ранее проведенных операций с двумя соседними вершинами (перестановки более широкого диапазона, т. е. по 4 и более, не выполнялись). Эксперименты с графами показали, что процедура улучшения маршрута при помощи перестановок достаточно эффективна и быстродействие ее весьма высоко.

Метод 2.2 (удаление петель). Часто текущий маршрут содержит петли. Например, на рис. 1 цепочка вершин 5-7-3-8-2-4 образуют петлю. Петля начинается с левой по ходу маршрута вершины отрезка 5-7 и заканчивается правой вершиной отрезка 2-4. Существование петли определяется наличием пересекающихся отрезков маршрута. Если внутреннюю цепочку петли повернуть в противоположном направлении, то есть заменить указанную цепочку на 5-2-8-3-7-4, то петля исчезнет (рис. 2), а маршрут станет короче.

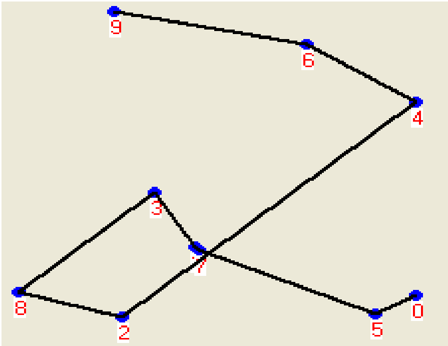


Рис. 1. Маршрут с петлей

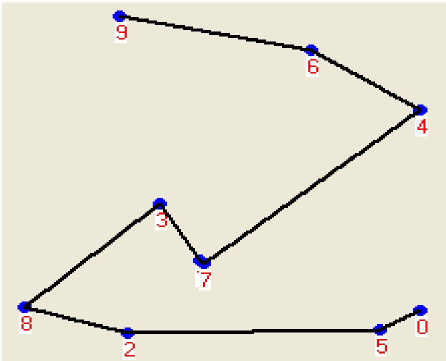


Рис. 2. Улучшенный маршрут

Метод отличается чрезвычайно высоким быстродействием и высокой эффективностью.

Метод 2.3 (разворот цепочек). Как показали эксперименты, отсутствие петлей еще не означает, что процедура разворота цепочек без петель неэффективна. Для оптимизации текущего маршрута применялась процедура разворота всех возможных цепочек. Метод имеет самое низкое быстродействие в сравнении с другими методами улучшения. Поэтому на практике его применяли для цепочек с числом звеньев не более шести.

Метод 2.4 (комбинированный). После нахождения какого-нибудь базового маршрута G к нему применялась комбинированная процедура улучшения по методам 2.1 – 2.3. Хотя метод 2.2 является частным случаем метода 2.3, его все равно применяли из-за высокого быстродействия и способности к эффективному развороту цепочек из любого числа звеньев. Код метода показан на листинге 1.

```
    procedure   CorrectPath(N:   Integer;
var G: TIntVec; var Path: Integer);
    begin
     repeat
     until not Permutations(N,G) and not
ChainTurnings(N,G) and
        not  CrossDeleting(N,G)  and  not
MoveTops(N,G);
     Path:= PathByG(N,G); // расчет дли-
ны маршрута
    end;
```

Листинг 1

4. Приближенные комбинированные методы нахождения кратчайшего маршрута

Применив три метода 1.1, 1.2, 1.3 расчета базового маршрута и комбинированный метод 2.4 их улучшения, получили три приближенных метода расчета маршрута (листинги 2 – 4).

метод 3.1:

```
    procedure   GreedilyCorrect(N:   Inte-
ger; var G: TIntVec; var Path: Integer);
    begin
     Greedily(N,G);
     CorrectPath(N,G,Path);
    end;
```

Листинг 2

метод 3.2:

```
    procedure   WoodyCorrect(N:   Integer;
var G: TIntVec; var Path: Integer);
    begin
     Woody(N,G);
     CorrectPath(N,G,Path);
    end;
```

Листинг 3

метод 3.3:

```
    procedure   SimplyCorrect(N:   Integer;
var G: TIntVec; var Path: Integer);
    begin
     Simply(N,G);
     CorrectPath(N,G,Path);
    end;
```

Листинг 4

В экспериментах с методами 3.1–3.3 установлено, что ни один из них не является предпочтительным.

В зависимости от матрицы A лучший результат с равной вероятностью мог дать любой из этих методов (интересно, что даже простейший базовый маршрут G[i] = i после улучшений нередко трансформировался в самый короткий маршрут, что свидетельствует о том, что решение задачи практически не зависит от выбора базового маршрута).

Поэтому в качестве рабочего применяли комбинированный метод 3.4 (комбинация всех), суть которого состоит в последовательном применении методов 3.1–3.3 к матрице A с последующим выбором лучшего маршрута среди сформированных этими методами.

Для того чтобы можно было оценить точность приближенной методики разработана рекурсивная процедура (RecoursiveMethod), позволяющая получить точное решение задачи переборным методом. Для повышения быстродействия в процедуру внесены некоторые очевидные эвристические усовершенствования.

Процедура позволила получить точное решение за приемлемое для проведения необходимых оценок время (до 5 минут на вариант размещения городов) при N<23.

Для оценки точности метода 3.4 при больших значениях N *(*N>22*)* процедуру RecoursiveMethod применить нельзя, поэтому составлена процедура Rand многократного применения метода 3.3 к одной и той же матрице A с различными случайными базовыми маршрутами. Процедура последовательно формирует маршруты до тех пор, пока последний лучший маршрут не повторится 5 раз подряд. Нельзя сказать, что такой способ позволяет найти самый короткий маршрут. Однако результаты работы процедуры дают интуитивную уверенность в том, что сравнение «быстрого» результата с результатом длительной работы метода 3.4 имеет достаточно высокую вероятность корректности за неимением точных методов. Уверенность в этом подкреплена выводом, который получен после обработки сотен различных матриц для NRecoursiveMethod и приближенной Rand (т.е. для данных N процедура Rand всегда находила точное решение задачи).

5. Результаты выполненных экспериментов

Результаты экспериментов программного модуля для анализа использования разных методов нахождения кратчайшего пути приведены в таблицах 1, 2.

Таблица 1

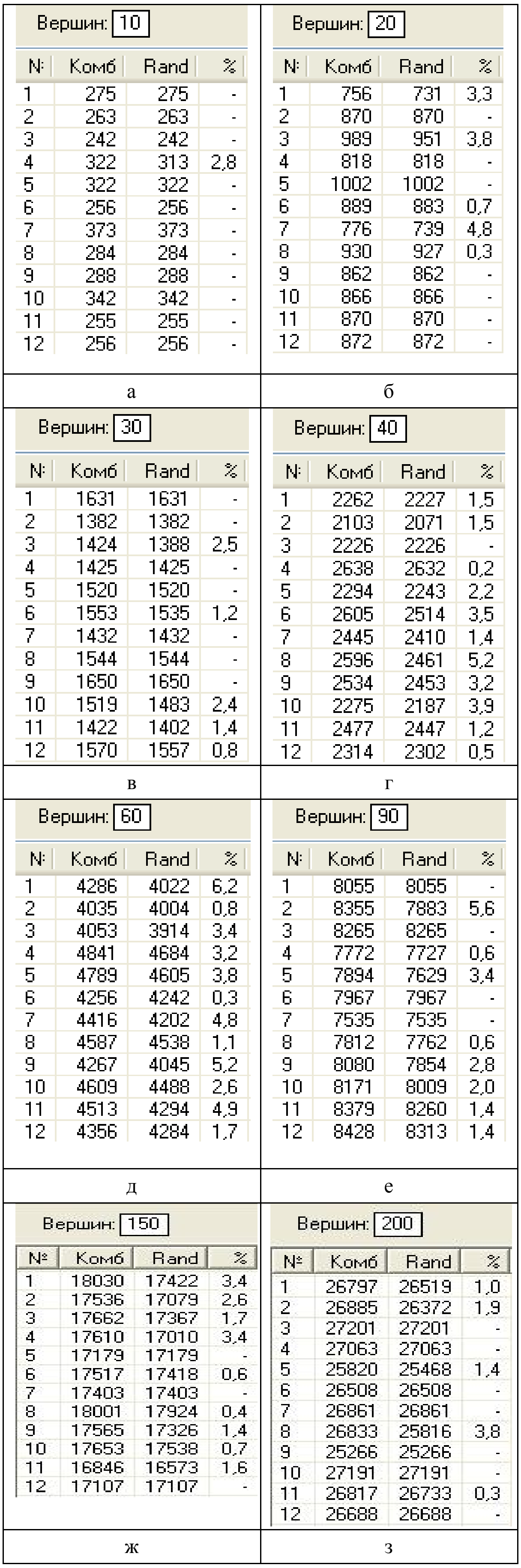
Результаты быстрых методов нахождения кратчайшего пути

Кол-во вер-шин	Жадный		Деревянный		Простейший	
	Длина	Время (сек)	Длина	Время (сек)	Длина	Время (сек)
10	312	0,005	299	0,008	299	0,009
20	869	0,012	886	0,037	919	0,025
50	3465	0,047	3528	0,062	3689	47
100	9453	0,057	10168	0,069	10240	453
250	38243	11,81	38478	12,45	39635	15,56
500	102122	90,55	105133	147	105633	200
1000	299614	1200	302782	900	305725	1980

На рис. 3 показаны результаты расчета маршрутов и их протяженности (Комб и Rand) для случайного расположения городов при помощи быстрой процедуры комбинированного метода 3.4 и процедуры Rand.

В последней колонке таблиц приведена процентная погрешность метода 3.4, которую рассчитывалась по формуле 2.

$$K = \frac{100 \cdot (\text{Ком} - \text{Rand})}{\text{Комб}}, \% \qquad (2)$$



Вершин: 10

№	Комб	Rand	%
1	275	275	-
2	263	263	-
3	242	242	-
4	322	313	2,8
5	322	322	-
6	256	256	-
7	373	373	-
8	284	284	-
9	288	288	-
10	342	342	-
11	255	255	-
12	256	256	-

а

Вершин: 20

№	Комб	Rand	%
1	756	731	3,3
2	870	870	-
3	989	951	3,8
4	818	818	-
5	1002	1002	-
6	889	883	0,7
7	776	739	4,8
8	930	927	0,3
9	862	862	-
10	866	866	-
11	870	870	-
12	872	872	-

б

Вершин: 30

№	Комб	Rand	%
1	1631	1631	-
2	1382	1382	-
3	1424	1388	2,5
4	1425	1425	-
5	1520	1520	-
6	1553	1535	1,2
7	1432	1432	-
8	1544	1544	-
9	1650	1650	-
10	1519	1483	2,4
11	1422	1402	1,4
12	1570	1557	0,8

в

Вершин: 40

№	Комб	Rand	%
1	2262	2227	1,5
2	2103	2071	1,5
3	2226	2226	-
4	2638	2632	0,2
5	2294	2243	2,2
6	2605	2514	3,5
7	2445	2410	1,4
8	2596	2461	5,2
9	2534	2453	3,2
10	2275	2187	3,9
11	2477	2447	1,2
12	2314	2302	0,5

г

Вершин: 60

№	Комб	Rand	%
1	4286	4022	6,2
2	4035	4004	0,8
3	4053	3914	3,4
4	4841	4684	3,2
5	4789	4605	3,8
6	4256	4242	0,3
7	4416	4202	4,8
8	4587	4538	1,1
9	4267	4045	5,2
10	4609	4488	2,6
11	4513	4294	4,9
12	4356	4284	1,7

д

Вершин: 90

№	Комб	Rand	%
1	8055	8055	-
2	8355	7883	5,6
3	8265	8265	-
4	7772	7727	0,6
5	7894	7629	3,4
6	7967	7967	-
7	7535	7535	-
8	7812	7762	0,6
9	8080	7854	2,8
10	8171	8009	2,0
11	8379	8260	1,4
12	8428	8313	1,4

е

Вершин: 150

№	Комб	Rand	%
1	18030	17422	3,4
2	17536	17079	2,6
3	17662	17367	1,7
4	17610	17010	3,4
5	17179	17179	-
6	17517	17418	0,6
7	17403	17403	-
8	18001	17924	0,4
9	17565	17326	1,4
10	17653	17538	0,7
11	16846	16573	1,6
12	17107	17107	-

ж

Вершин: 200

№	Комб	Rand	%
1	26797	26519	1,0
2	26885	26372	1,9
3	27201	27201	-
4	27063	27063	-
5	25820	25468	1,4
6	26508	26508	-
7	26861	26861	-
8	26833	25816	3,8
9	25266	25266	-
10	27191	27191	-
11	26817	26733	0,3
12	26688	26688	-

з

Рис. 3. Результаты расчета маршрутов и их протяженности

Таблица 2
Результаты метода 3.4 и процедуры Rand
для нахождения кратчайшего пути

Кол-во вершин	Комбинация		Rand	
	Длина	Время (сек)	Длина	Время (сек)
10	299	0,008	299	0,016
20	869	0,026	869	0,433
50	3465	0,187	3353	92,20
100	9453	1,377	9115	183
250	38243	40,3	38243	224
500	102122	456	101634	794
1000	299614	3665	290125	10833,5

В результате экспериментов с несколькими сотнями матриц расстояний для различных N, получены данные, которые свидетельствуют, что независимо от количества *N* городов погрешность метода 3.4 никогда не превосходила 8%. На основании обработки многочисленных расчетных данных получена формула ориентировочной оценки быстродействия метода 3.4.

Среднее время t(c) расчета на компьютере с процессором Intel 1400 кратчайшего маршрута с N городами составило $t = 4 \cdot 10^{-6} N^3$.

Для N=100 среднее время расчета маршрута составляет 4 секунды. Для практически используемых N<31 это время не превосходит 0,1 с.

Выводы

Проведенный анализ существующих методов построения маршрутов показал, что наряду с достаточной их теоретической проработанностью, на данный момент не существует алгоритма, который было бы просто реализовать в современных средах разработки. В результате проведенного исследования предложен достаточно эффективный алгоритм построения маршрута.

На практике, для получения оперативной информации о кратчайшем маршруте, где главным является время нахождения маршрута и время его прохождения можно использовать приближенный эвристический метод, который даёт погрешность результата, как показали эксперименты не более 8%.

Если в поставленной задаче делается акцент на максимально точный результат, например инженерные расчёты, надо использовать точные эвристические методы процедуры Rand.

Если учесть результаты всех выполненных исследований, можно значительно повысить производительность, быстродействие, точность и эффективность программных комплексов, разрабатываемых для решения транспортных задач.

Список литературы

1. *Акимов Е. Дискретная математика. Логика, группы, графы. / Е. Акимов. – М.: Изд. дом Лаборатория базовых знаний, 2003. – 376 с.*
2. *Ларичев О.И. Теория и методы принятия решений, а также хроника событий в Волшебных Странах: Учебник. / О.И. Ларичев , Р.В. Браун. – М.: Логос, 2000. – 296 с.*
3. *Бурков В.Н. Прикладные задачи теории графов. / В.Н. Бурков, И.А. Горгидзе, С.Е. Ловецкий. – Тбилиси: Мецниереба, 1974. – 234 с.*
4. *Емеличев В.А. Лекции по теории графов / В.А. Емеличев, О.И. Мельников, В.И. Сарванов, Р.И. Тышкевич. – М.: Наука, 1990. – 384 с.*
5. *Алексеев В.Е. Графы и алгоритмы. Структуры данных. Модели вычислений. / В.Е. Алексеев, В.А. Таланов. – Бином. Лаборатория знаний, Интернет-университет информационных технологий – интуит.ру, 2006.*

Поступила в редакцию 12.02.2013

Рецензент: канд физ.-мат. наук, ст. научн. сотр. А.А. Можаев, Национальный технический университет «ХПИ», Харьков.

МОДЕЛЮВАННЯ ТА ВДОСКОНАЛЕННЯ АЛГОРИТМУ ФОРМУВАННЯ ОПТИМАЛЬНИХ МАРШРУТІВ ТРАНСПОРТНИХ ПЕРЕВЕЗЕНЬ

В.Д. Ліпанов, С.А. Кічкайло

В результаті проведеного моделювання та аналізу існуючих методів формування оптимальних маршрутів руху при організації транспортних перевезень запропоновано удосконалений алгоритм формування оптимального маршруту з урахуванням можливостей перевізника і вимог клієнтів.

Ключові слова: *граф ,вантаж, критерій, логістика, маршрут, метод гілок та границь, шкала порядку, найкоротший шлях, дерев'яний алгоритм, жадібний алгоритм.*

MODELING AND IMPROVEMENT OF THE ALGORYTHM FORMATION FOR THE OPTIMIZED TRANSPORT ROUTES

V.D. Lipanov, S.A. Kichkaylo

The improved algorithm for the optimized transport route formation is offered as a result of modeling and the analysis of the methods of the algorithm formation for the optimized transport routes at the organization of the transportations. The given algorithm is considering the carrier's possibilities and client's needs.

Keywords: *graph, cargo, criterion, logistics, route, branch and bound, the order's, the shortest path, wooden algorithm, greedy algorithm.*