ВЕСТНИК ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА МОРСКОГО И РЕЧНОГО ФЛОТА ИМЕНИ АДМИРАЛА С. О. МАКАРОВА

АВТОМАТИЗАЦИЯ И УПРАВЛЕНИЕ ТЕХНОЛОГИЧЕСКИМИ ПРОЦЕССАМИ И ПРОИЗВОДСТВАМИ

DOI: 10.21821/2309-5180-2024-16-6-992-1002

AUTOMATING THE SEARCH FOR THE SHORTEST CLOSED PATH IN TRANSPORT NETWORK BY MEANS OF MATLAB

A. A. Chertkov, V. G. Nikiforov

Admiral Makarov State University of Maritime and Inland Shipping,
St. Petersburg Russian Federation

The paper considers the problem of automating the search and construction of the shortest closed route for a group of merchant ships on a given set of transit nodes of the transport network with known coordinates. The goal is to find the shortest closed route that passes through all hosts only once and ends at the outgoing node. In general, a weighted graph can serve as a mathematical model of such a transport network, in which the criterion for the efficiency of the desired route can be, for example, the distance between cities or ports, time or operating costs. This allows you to identify time reserves that can be used to save fuel and energy, taking into account the load and cost of goods, travel costs and other factors. The work uses a combinatorial method using stochastic (probabilistic) programming, which is implemented using an annealing simulation algorithm supplemented by a recursive step-by-step optimization procedure. The problem belongs to the class of transcomputational even with a small network dimension, which makes it impossible to solve it by brute force on modern computers in a reasonable time. The proposed modification of the simulated annealing algorithm eliminates this limitation and allows not only to evaluate the shortest paths in the network, but also to build closed paths that pass through all the vertices of the network once. The use of an iterative strategy in the algorithm using a probabilistic scenario according to the Monte Carlo method allows you to avoid the solution falling into the local minimum and provides global optimization. The algorithm is implemented in MATLAB codes in the form of a recursive step-by-step optimization procedure with the construction of a curve of a closed route on the coordinate plane through the entire set of nodes of the transport network without mutual intersections of its sections. It is shown that the result of recursive optimization is to obtain a numerical estimate of the closed path of the minimum weight in the full weighted graph, which is a model of the transport network with the definition of an array of transit node numbers on this route. The developed algorithm and global optimization procedure can be used to automate the search for energy-efficient solutions in the control of robotic and unmanned objects, as well as ships in the performance of cargo transportation in water transport.

Keywords: search automation, transport network, circular closed route, Hamiltonian cycle, simulated annealing, control parameter, global optimization, energy state.

For citation:
Chertkov, Alexandr A. and V. G. Nikiforov, “Automating the search for the shortest closed path in transport network by means of MATLAB.” *Vestnik Gosudarstvennogo universiteta morskogo i rechnogo flota imeni admirala S. O. Makarova*. 16.6 (2024): 992–1002. DOI: 10.21821/2309-5180-2024-16-6-992-1002.

УДК 681.5

АВТОМАТИЗАЦИЯ ПОИСКА КРАТЧАЙШЕГО ЗАМКНУТОГО ПУТИ В ТРАНСПОРТНОЙ СЕТИ СРЕДСТВАМИ MATLAB

А. А. Чертков, В. Г. Никифоров

ФГБОУ ВО «ГУМРФ имени С. О. Макарова»
Санкт-Петербург, Российская Федерация

В работе рассмотрена задача автоматизации поиска и построения кратчайшего замкнутого маршрута для группы судов торгового флота на заданном множестве транзитных узлов транспортной сети с известными координатами. Цель — найти самый короткий замкнутый маршрут, который проходит через все узлы сети только по одному разу и завершается в исходящем узле. В общем случае математической моделью такой транспортной сети может служить взвешенный граф, в котором критерием эффективности искомого маршрута может служить, например, расстояние между городами или портами, время или эксплуатационные расходы. Это позволяет определить резервы времени, которые можно использовать для экономии топлива и энергии с учетом загрузки и стоимости грузов, путевых расходов и других факторов. В работе применяется комбинаторный метод с использованием стохастического (вероятностного) программирования, который реализуется с помощью алгоритма имитации отжига, дополненного рекурсивной процедурой пошаговой оптимизации. Задача относится к классу трансвычислительных даже при небольшой размерности сети, что делает невозможным ее решение методом перебора вариантов на современных компьютерах за разумное время. Предложенная модификация алгоритма имитации отжига устраняет это ограничение, позволяя не только оценивать кратчайшие пути в сети, но и строить замкнутые пути, проходящие через все вершины сети один раз. Применение в алгоритме итерационной стратегии с использованием вероятностного сценария по методу Монте-Карло позволяет избежать попадания решения в локальный минимум и обеспечивает глобальную оптимизацию. Алгоритм реализован в кодах MATLAB в виде рекурсивной процедуры пошаговой оптимизации с построением на координатной плоскости кривой замкнутого маршрута через все множество узлов транспортной сети без взаимных пересечений его участков. Показано, что результатом рекурсивной оптимизации является получение численной оценки замкнутого пути минимального веса в полном взвешенном графе, являющемся моделью транспортной сети с определением массива номеров транзитных узлов на этом маршруте. Разработанные алгоритм и процедура глобальной оптимизации могут быть использованы для автоматизации поиска энергоэффективных решений при управлении робототехническими и беспилотными объектами, а также судами при выполнении грузоперевозок на водном транспорте.

Ключевые слова: автоматизация поиска, транспортная сеть, кольцевой замкнутый маршрут, Гамильтонов цикл, имитация отжига, управляющий параметр, глобальная оптимизация, энергетическое состояние.

Для цитирования:
Чертков А. А. Автоматизация поиска кратчайшего замкнутого пути в транспортной сети средствами MATLAB / А. А. Чертков, В. Г. Никифоров // Вестник Государственного университета морского и речного флота имени адмирала С. О. Макарова. — 2024. — Т. 16. — № 6. — С. 992–1002. DOI: 10.21821/2309-5180-2024-16-6-992-1002.

Введение (Introduction)

В эпоху, когда экономика активно переходит на цифровые технологии, а беспилотные транспортные средства становятся неотъемлемой частью логистических процессов, особую актуальность приобретают вопросы энергоэффективности и оптимизации движения. Одним из ключевых аспектов этой проблемы является сбережение топливно-энергетических ресурсов на маршрутах движения судов. Поиск оптимальных маршрутов движения судов с учетом указанных факторов может быть осуществлен с помощью методов математического программирования и исследования операций. Для этого используются теории графов, позволяющие графически интерпретировать объекты и их модели. Существует множество алгоритмов поиска на графах кратчайших путей до конкретных целей, из которых наиболее известными являются алгоритмы линейного и динамического программирования Дейкстры, Беллмана – Форда, Флойда – Уоршелла, подробно рассмотренные в работах [1]–[5]. Однако они не адаптированы для поиска кратчайшего замкнутого пути, проходящего через все вершины сетевой модели и завершающегося в исходном пункте.

Подобную задачу оптимизации маршрута впервые сформулировал Карл Менгер в 1930 г. как задачу посыльного, которому нужно найти кратчайший путь, пройдя конечное множество мест с известным расстоянием между ними. По сути, эта задача сводилась к поиску самого выгодного маршрута с посещением (обходом) всех мест только по одному разу с возвращением в исходный пункт. Поэтому такая задача в научной общественности получила название *задачи коммивояжера*, которая впоследствии приобрела широкую известность как задача *комбинаторной оптимизации*.

С появлением мощных вычислительных средств, позволяющих реализовать методы дискретной оптимизации в задачах комбинаторики высокой размерности, группой исследователей из Англии (Дэвид Аплгейт, Роберт Биксби, Уильям Кук и др.), а затем из Германии (Герхард Райнельт) в рамках многоцелевого проекта «Конкорд» выполнялись работы по созданию различных вариаций решения задач коммивояжера для проведения сравнительного анализа эффективности полученных решений другими исследователями. Так, в марте 2005 г. ими была решена задача с 33810 узлами и вычислен путь длиной в 66048945 у.е., а в апреле 2006 г. получено решение для задачи с 85900 узлами. Результаты их работы подтверждают возможность получения решений задач методами декомпозиций с миллионами транзитных узлов, причем полученные оценки не превышают оптимальные более чем на 1 %.

Основными областями применения задачи коммивояжера являются:

– транспортная и дорожная сеть населенных пунктов, требующие оптимизации дорожного движения;

– организация, планирование и размещение логистических схем с минимальными эксплуатационными расходами на перевозку;

– телекоммуникационные узлы в сетях передачи данных и интернет-среде, осуществляющие локальную и глобальную маршрутизацию, от настройки которых зависит качество и скорость передачи данных и информационных потоков;

– обнаружение и распознавание различных визуальных объектов;

– разработка цифровых фильтров специального назначения.

Цифровизация технологических процессов за счет широкого использования технологий искусственного интеллекта, включая использование алгоритмов имитации отжига, генетических и других, позволит системно решать в отрасли вопросы робототехнического безэкипажного управления судами, а также вопросы оптимального планирования, организации и размещения логистических объектов, обеспечивая минимальные эксплуатационные расходы на погрузку и перевозку.

Методы и материалы (Methods and Materials)

Решение задачи оптимизации замкнутого маршрута. Для решения сформулированной ранее задачи оптимизации замкнутого маршрута, которая относится к классу задач комбинаторной оптимизации, теоретически разработаны алгоритмы точных решений (полного перебора, метод ветвей и границ), эвристические (метод включения дальнего, BV-метод) и поисковые (генетический алгоритм, муравьиный алгоритм — ACS-Q). Сравнительный анализ алгоритмов (см. работы [6]–[11]) по качеству получаемых решений и их быстродействию показал следующее: эвристические относятся к «жадным» алгоритмам, которые не всегда дают наилучшее оптимальное решение, а только приближенное, точные алгоритмы малопригодны для решения задач больших размерностей (неспособны решить задачу за разумное время), а алгоритмы поиска являются компромиссом между эвристическими и точными методами, требуя подбора параметров. Временные оценки работы алгоритмов позволяют оценить длительность решения задачи и выбрать наиболее подходящий метод, когда быстродействие является критичным параметром.

В связи с бурным развитием мощных вычислительных и программных средств появилась возможность исследовать один из поисковых алгоритмов, получивший широкую известность только в последнее десятилетие — алгоритм имитации отжига, автором которого считается В. Н. Метрополис благодаря известной работе [12], опубликованной в 1953 г. Анализ особенностей и практической реализации алгоритма имитации отжига для поиска кратчайшего замкнутого пути от любого узла сети, проходящего через все остальные узлы транспортной сети с возвращением к тому же начальному узлу, также рассматривались в работах [13]–[15]. При этом возникала сложность алгоритмического конструирования конфигурации замкнутого пути (без каких-либо пересечений между собой отдельных участков этого пути). Характерной особенностью алгоритма имитации отжига, позволяющей решать задачи глобальной оптимизации, является применение итерационной стратегии с использованием случайного сценария по методу Монте-Карло. Это позволило автору обнаружить сходство между процессом кристаллизации твердых тел и общими стохастическими комбинаторными задачами, относящихся к NP классу (недетерминированных полиномиальных) задач. Отличительной особенностью данного алгоритма от других алгоритмов локального поиска, является его способность преодолевать состояния вычислительного процесса, когда целевая функция попадает в локальный минимум. Этому способствует возможность вероятностного выбора на некотором этапе процесса худшего решения, имеющего большее целевое значение в окрестности поиска.

Согласно данной стратегии предполагается, что активность атомов тем больше, чем выше температура, но с понижением температуры атомы выстраиваются в узлах кристаллической решетки, переходя в состояния с меньшим уровнем энергии. В результате повышается устойчивость кристаллической решетки, которая в устойчивом состоянии соответствует минимуму энергии. При этом вероятность переходов атомов в состояния с большей энергией уменьшается, но еще допустимы переходы отдельных атомов в эти состояния. В такой стратегии оптимизации целевая функция эквивалентна внутренней энергии металла, а комбинированное пространство энергетических состояний эквивалентно множеству значений независимой векторной переменной задачи оптимизации.

Основным управляющим параметром алгоритма имитации отжига является температура T, процесс уменьшения которой рассматривается как итерация алгоритма Метрополиса. На начальном этапе, когда значение T велико, существует большая вероятность принятия плохих из худших решений, но по мере снижения параметра T вероятность принятия лучших из худших решений возрастает и, наконец, когда параметр T стремится к нулю, никакие ухудшающиеся решения не могут быть приняты.

Таким образом, цель алгоритма имитации отжига состоит в том, чтобы обеспечить такое управление процессом снижения температуры, при котором задача достижения глобальной оптимизации будет решена на множестве из комбинированных состояний, для которого значение целевой функции будет минимальным.

Алгоритм поиска глобального минимума методом имитации отжига предполагает свободное задание следующих параметров:

– $s_0 \in S$ — начального состояния из множества всех состояний системы (решений) задачи. Тогда s_i — состояние системы на i-м шаге алгоритма;

– начального T_0 и конечного T_f значений управляющего параметра: температуры, представляющей собой убывающую с течением времени функцию от аргумента T;

– параметра α затухания управляющего параметра T (для расчета понижающего ряда температур);

– длины L_k марковской цепи, определяющей количество итераций при любой температуре T;

– функции E энергии, подлежащей оптимизации;

– функции F, генерирующей новое состояние «кандидат», которое либо принимается, либо не принимается.

Основные шаги алгоритма:

1. Принимаем температуру отжига T, равную начальной T_0.

2. Генерируем случайным образом новое состояние s_c и вычисляем соответствующую ему энергию $E(s_c)$.

3. Уменьшаем параметр T до следующего значения T_i согласно расписанию охлаждения.

Если $T_i > T_f$, то с помощью функции F генерируется новое состояние «кандидат»: $s_c = F(s_{i-1})$, соответствующее значению целевой функции $E(s_c)$.

4. Вычисляем приращение $\Delta E = E(s_c) - E(s_{i-1})$, для которого проверяются следующие условия:

– если $\Delta E \le 0$, то $s_i = s_c$, т. е. принимается новое состояние (решение);

– если $\Delta E > 0$, то переход в новое состояние осуществляется с вероятностью $P(\Delta E) = e^{-\Delta E/T}$.

Таким образом, если энергия «кандидата» меньше, он становится новым состоянием, в противном случае переход будет вероятностным (поэтому метод относят к классу стохастических).

ВЕСТНИК ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА МОРСКОГО И РЕЧНОГО ФЛОТА ИМЕНИ АДМИРАЛА С. О. МАКАРОВА

5. Понижаем температуру $T_{i+1} = T_i$.

6. Далее при параметре T_i процесс возмущения и приема решения о переходе в новое состояние (см. пп. 3 и 4 алгоритма) повторяется L_k раз.

7. При достижении температуры конечного значения T_f алгоритм останавливается, в противном случае выполняется дальнейшее охлаждение.

Покажем способ реализации на практике рассмотренного ранее метода имитации отжига для решения задачи коммивояжера — поиска кратчайшего замкнутого маршрута в сети.

Формальная постановка задачи и генерация исходных данных. Задана транспортная сеть с множеством транзитных узлов с известными координатами (рис. 1), для прохождения которых необходимо построить кратчайший замкнутый маршрут, который проходит через все транзитные узлы сети только по одному разу и завершается в исходящем узле. В общем случае необходимо решить задачу коммивояжера для транспортной сети, математической моделью которой является взвешенный граф. Пусть транзитные узлы случайным образом разбросаны в квадрате 100 × 100 км^2. Каждый узел сети представлен парой координат. Всего 50 узлов.

Рис. 1. Координатная плоскость для построения кратчайшего замкнутого пути между всеми узлами транспортной сети

Для получения исходных координат воспользуемся функцией rand из инструментария MATLAB и на ее основе построим следующий код:

```
>> N=50; % количество городов
>> cities = rand(N,2)*100; % генерация координат городов (x, y)
>> XY=cities(:,1:2); % вывод координат двумя столбцами X и Y
```

При этом генерируются N пар случайных чисел по закону равномерного распределения вероятностей, соответствующие координатам (x_i, y_i) узлов сети. Чтобы координаты находились в заданной плоскости, увеличиваем их значения в 100 раз. Каждый узел транспортной сети на плоскости представляется с соответствующим индексом (рис. 1). Решением оптимизационной задачи поиска является выбор кратчайшего из маршрутов, проходящих через каждый город только один раз. В этом случае множество состояний S представляет собой все возможные маршруты, проходящие через все узлы на графе, каждый из которых имеет свой порядковый номер в транспортной сети. Реализация задачи автоматизации поиска кратчайшего замкнутого пути методом имитации отжига требует выполнения следующих этапов.

На первом этапе для использования метода имитации отжига необходимо определить две функции, зависящие от каждой конкретной задачи: функцию энергии E (или «целевую функцию» в общепринятой терминологии) и функцию F, порождающую новое состояние. Так как необходимо минимизировать расстояние, оно и будет «энергией». Следовательно, целевая функция будет имеет следующий вид:

$$E(s_i) = E_i = \left[\sum_{i}^{N-1} \sqrt{(x_{k+1} - x_k)^2 + (y_{k+1} - y_k)^2} \right] + \sqrt{(x_{|C|} - x_1)^2 + (y_{|C|} - y_1)^2}. \quad (1)$$

Первое слагаемое в формуле (1) означает не что иное, как сумму евклидовых расстояний между парами узлов, составляющих маршрут s_i. Так как маршрут замкнутый, эта формула дополняется вторым слагаемым: $\sqrt{(x_{|C|} - x_1)^2 + (y_{|C|} - y_1)^2}$, которое представляет собой расстояние между последним и исходными узлами. Евклидово расстояние между точками A и B на плоскости с координатами $A = (x_a, y_a)$, $B = (x_b, y_b)$ задается следующей формулой:

$$d(A, B) = \sqrt{(x_a - x_b)^2 + (y_{a|} - y_b)^2}. \quad (2)$$

В связи с этим необходимо найти по формуле (2) матрицу евклидовых расстояний между всеми возможными парами узлов, которую можно реализовать за один шаг путем алгебраических операций над векторами координат. Предлагаемый программный код:

```
coor _ x _ 1 = cities(:,1)*ones(1,N);
coor _ x _ 2 = coor _ x _ 1';
coor _ y _ 1 = cities(:,2)*ones(1,N);
coor _ y _ 2 = coor _ y _ 1';
Sd=sqrt((coor _ x _ 1-coor _ x _ 2).^2 + (coor _ y _ 1-coor _ y _ 2).^2); % матрица расстояний
```

На втором этапе необходимо получить новое состояние-кандидат. Самым простым решением в этом случае будет произвольная перестановка местами двух номеров узлов в списке их маршрута следования. Однако это может привести к непредсказуемому значению энергии E_i и в конечном итоге к зацикливанию (зависанию) вычислительного алгоритма. Выходом из этой ситуации может быть инверсия пути между двумя случайно выбранными узлами на маршруте. К примеру, если по случайному закону произведен выбор двух чисел (3 и 8), на интервале из десяти номеров городов в порядке их перечисления, то новое состояние (маршрут) будет следующим: 1, 2, 8, 7, 6, 5, 4, 3, 9, 10. Для реализации этой процедуры предлагается следующий фрагмент программы:

```
D=randi[N,L,2] %Предыдущее состояние (маршрут), являющееся состоянием-кандидат
i = randi(D,2); % Генерируем целое случайное число
j = randi(n,2); % Генерируем целое случайное число
  if(i>j)
     D(j:i)=flipud(D(j:i)); % обращаем подпоследовательность
  else
     D(i:j)=flipud(D(i:j)); % обращаем подпоследовательность
  End
```

На третьем этапе начинаем процесс оптимизации. Для начала получим функцию энергии, соответствующую текущему состоянию цепи, или то, что оптимизируем, а именно длину замкнутого пути в транспортной сети: E: $S\_current \rightarrow Sopt$. Приведем фрагмент кода программы с пояснениями в виде комментариев:

```
ROUTE=randperm(n); % %Задаём начальный (случайный) маршрут из номеров узлов
S _ current=inf; %Текущее расстояние в цепи, являющееся текущим решением;
S _ opt=0;% Оптимальное расстояние (решение)
% Оптимизация путем выполнения итераций
for k = 1:L % Задаем число итераций
    Dk = D(k,[1,2]); % Выбор двух случайных узлов на k-м шаге
    if Dk(1) < Dk(2)
        if Dk(1)~= 1 && Dk(2)~= N
            S _ curent=Sd(ROUTE(Dk(1)-1),ROUTE(Dk(1))) + ...
              Sd(ROUTE(Dk(2)),ROUTE(Dk(2)+1));
      elseif Dk(1)~=1 && Dk(2)==N
            S _ curent =Sd(ROUTE(Dk(1)-1),ROUTE(Dk(1))) + ...
                Sd(ROUTE(Dk(2)),ROUTE(1));
      elseif Dk(1)==1 && Dk(2)~=N
            S _ curent=Sd(ROUTE(end),ROUTE(Dk(1))) + ...
                 Sd(ROUTE(Dk(2)),ROUTE(Dk(2)+1));
        end
      end
```

Как видно из приведенного фрагмента, в нем циклически выполняются итерации вычисления текущего расстояния цепи путем выборки из матрицы *Sd* расстояний согласно построенному маршруту ROUTE из номеров узлов. Далее покажем, как на основе предыдущего состояния *S_ current* вычисляется новое расстояние в цепи (состояние-кандидат *S_opt*), в которое система может перейти, а может и отбросить. Способ получения «кандидата» полностью зависит от решаемой задачи на основе аналогичной процедуры, показанной ранее:

```
ROUTEp = ROUTE; % Задаем потенциальный (новый) маршрут
% Вычисляем новое расстояние в цепи (состояние - кандидат)
if Dk(1)<Dk(2)
        ROUTEp(Dk(1):Dk(2))=ROUTEp(Dk(2):-1:Dk(1));
        if Dk(1)~=1 && Dk(2)~=N
           Sopt=Sd(ROUTEp(Dk(1)-1),ROUTEp(Dk(1))) + ...
                Sd(ROUTEp(Dk(2)),ROUTEp(Dk(2)+1));
         elseif Dk(1)~=1 && Dk(2)==N
           Sopt=Sd(ROUTEp(Dk(1)-1),ROUTEp(Dk(1))) + ...
                Sd(ROUTEp(Dk(2)),ROUTEp(1));
         elseif Dk(1)==1 && Dk(2)~=N
            Sopt=Sd(ROUTEp(end),ROUTEp(Dk(1))) + ...
                Sd(ROUTEp(Dk(2)),ROUTEp(Dk(2)+1));
        end
```

В следующем фрагменте кода программы выполнено сравнение нового расстояния (состояния-кандидата) с текущим расстоянием (состоянием). Если расстояние «кандидата» меньше, то оно становится новым состоянием, в противном случае переход будет вероятностным, поэтому метод относят к классу стохастических. Далее приведен фрагмент программы, соответствующий этим операциям:

```
RANDONE = rand(L,1); % формируем заранее список вероятностей перехода
Sp<S
        ROUTE=ROUTEp;
```

```
        iter=k;
    else
        P=exp((-(Sopt-S _ current))./t); % вычисляем вероятность перехода
          if RANDONE(k)<=P
             ROUTE=ROUTEp;
          end
     end
      % t = t.*a;% Управляющий параметр t (температура) уменьшается
      t=t0/k; % уменьшаем температуру
end
```

Вероятностный переход обеспечивается функцией *rand* генерации равномерно распределенного случайного числа в MATLAB. Иллюстрация такого перехода демонстрируется с помощью шкалы вероятностей, распределенных по равномерному закону от нуля до единицы, где вероятность перехода к новому состоянию (расстоянию) обозначена символом *P*:

Если значение сгенерированной вероятности *P* попало в левую часть, то выполняем переход, если в правую — не делаем ничего. На этом завершается описание основных шагов алгоритма и можно перейти к его реализации на этапе моделирования алгоритма поиска кратчайшего замкнутого пути на заданной координатной плоскости (см. рис. 1).

Результаты (Results)

В результате пошаговой реализации алгоритма поиска оптимального замкнутого маршрута на транспортной сети получены численные значения кратчайшего пути Sp замкнутого маршрута и компонентов вектора ROUTE, соответствующих номерам вершин (узлов), лежащих на этом пути:

```
>> simotg4d
Sp =
   106.4496

ROUTE =

Columns 1 through 17

1   14   22   28   6   38   9   40   30   27   41   34   36   31   42   5   12

Columns 18 through 34

25   49   10  13   45   4   35   50   47   44   39   43   23   32  2   16   7

Columns 35 through 50

20   8   33  37   26   18   48   11   15   29   24   17   46   3   19   21

Elapsed time is 0.455323 seconds.

>>
```

На рис. 2 средствами MATLAB показан замкнутый маршрут кратчайшего пути, проходящего через все вершины координатной плоскости (графа) и соответствующего виртуальной транспортной сети, сгенерированной по закону равномерного распределения вероятностей.

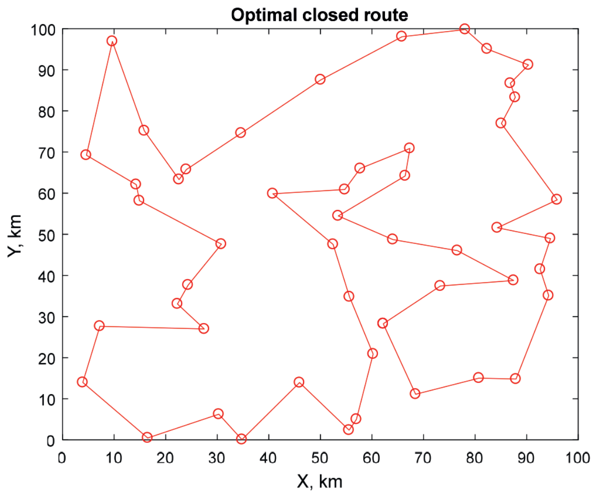

Рис. 2. Кратчайший замкнутый маршрут в транспортной сети

Как видно из полученных результатов, затраченное время на поиск решения составляет всего 0, 45523 с. Особенностью построенного маршрута является то, что замкнутый путь может исходить из любой вершины полносвязного взвешенного графа (сетевой модели), проходить через все вершины графа только один раз (без пересечений ранее пройденных ребер графа) и возвращаться в ту же самую вершину. На интервале времени от $t_{\min} = 0{,}1$ до $t_{\max} = 10$ с программа выдает гарантированный результат. При этом необходимо всего L итераций (шагов), численно равных заданной длине марковской цепи. При меньшей длине цепи нет гарантии от застревания вычислительного процесса в локальном минимуме и его завершения оптимальной оценкой расстояния замкнутого пути.

Выводы (Summary)

На основе выполненного исследования можно сделать следующие выводы:

1. Разработанный алгоритм автоматизации поиска является универсальным для любого множества транзитных узлов потоковой сети, от числа которых зависит лишь выбор длины марковской цепи, определяющей количество пошаговых итераций, необходимых для получения оптимального решения.

2. Алгоритм и построенная на его основе компьютерная модель применимы для оперативного сравнительного анализа предлагаемых маршрутов движения судов по критерию минимальных эксплуатационных расходов с учетом воздействия внешних факторов, а также для оптимального планирования и размещения логистических объектов в транспортной сети.

3. Предложена модификация алгоритма имитации отжига, предусматривающая возможность получения нового состояния (расстояния) не только за счет случайной перестановки местами номеров вершин сетевого графа, но и инверсии маршрута между ними при каждой последующей итерации.

4. Достоинствами предложенной модификации алгоритма имитации отжига является возможность программной реализации стохастической комбинаторной оптимизации, реализуемой с применением средств из Приложения Optimization Toolbox среды MATLAB.

5. Работоспособность и эффективность алгоритма оптимизации поиска кратчайшего замкнутого пути на множестве транзитных узлов подтверждена многократными испытаниями, при которых не было ни одного случая застревания алгоритма в локальном минимуме, которое могло вызвать «зависание» процедуры его поиска на сетевой транспортно-логистической модели и не дать гарантированного решения.

СПИСОК ЛИТЕРАТУРЫ

1. *Сахаров В. В.* Автоматизация поиска оптимальных маршрутов и грузовых потоков в транспортных сетях средствами целочисленного линейного программирования / В. В. Сахаров, И. А. Сикарев, А. А. Чертков // Вестник Государственного университета морского и речного флота имени адмирала С. О. Макарова. — 2018. — Т. 10. — № 3(49). — С. 647-657. DOI 10.21821/2309-5180-2018-10-3-647-657. — EDN UTPDIL.
2. *Чертков А. А.* Рекурсивный метод оптимизации логистических путей средствами MATLAB / А. А. Чертков, А. А. Вардомская, А. А. Дмитриев // Вестник государственного университета морского и речного флота им. адмирала С. О. Макарова. — 2015. — № 6(34). — С. 196–204. — DOI: 10.21821/2309-5180-2015-7-6-196-204. — EDN VCKLFP.
3. *Чертков А. А.* Автоматизация выбора кратчайших маршрутов судов на основе модифицированного алгоритма Беллмана-Форда / А. А. Чертков // Вестник государственного университета морского и речного флота им. адмирала С. О. Макарова. — 2017. — Т. 9. — № 5. — С. 1113–1122. — DOI: 10.21821/2309-5180-2017-9-5-1113-1122. — EDN ZSRYGJ.
4. *Сахаров В. В.* Маршрутизация сетей с отрицательными весами звеньев в пакете оптимизации MATLAB / В. В. Сахаров, А. А. Чертков, Л. Б. Очина // Вестник государственного университета морского и речного флота им. адмирала С. О. Макарова. — 2019. — Т. 11. — № 2. — С. 230–242. — DOI: 10.21821/2309-5180-2019-11-2-230-242. — EDN GFTRWW.
5. *Sakharov V. V.* Network routing method for ships and other moving objects using MATLAB / V. V. Sakharov, A. A. Chertkov, I. B. Ariefjew // Scientific Journal of the Maritime University of Szczecin. — 2020. — Is. 62(134). — Pp. 61–68.
6. *Hornik K.* TSP – Infrastructure for the traveling salesperson problem / K. Hornik // Journal of statistical software. — 2007. — Is. 23 (2). — Pp. 1–21. DOI: 10.18637/jss.v023.i02.
7. *Борознов В. О.* Исследование решения задачи коммивояжера / В. О. Борознов // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. — 2009. — № 2. — С. 147–151. — EDN KWYPNF.
8. *Stützle T.* Parameter Adaptation in Ant Colony Optimization / T. Stützle, M. López-Ibáñez, P. Pellegrini, M. Maur, M. de Oca, M. Birattari, Michael Maur, M. Dorigo // Technical Report — IRIDIA, Université Libre de Bruxelles 2010-002, 2010. — 26 p.
9. Кормен Т. Х. Алгоритмы. Построение и анализ. — 2 изд. / Т. Х. Кормен, Ч. И. Лейзерсон, Р. Р. Ривест, К. Штайн — М.: Вильямс, 2012. — 1296 с.
10. *Костюк Ю. Л.* Эффективная реализация алгоритма решения задачи коммивояжёра методом ветвей и границ // Прикладная дискретная математика. — 2013. — № 2(20). — С. 78–90. — EDN QCAKVP.
11. *Кирсанов М. Н.* Анализ алгоритмов выбора оптимальных маршрутов группы судов / М. Н. Кирсанов // Вестник государственного университета морского и речного флота им. адмирала С. О. Макарова. — 2016. — № 2(36). — С. 183–190. DOI: 10.21821/2309-5180-2016-8-2-183-190. — EDN VTNQHD.
12. *Metropolis V. N.* Equations of state calculations by fast computing machines / V. N Metropolis., A. W. Rosenbluth, M. N.Rosenbluth, A. H. Teller, E. J. Teller // Chem. Phys. — 1953. — Vol. 21. — Pp. 1087–1092.
13. *Winkler G.* Image analysis, random fields and Markov chain Monte Carlo methods: a mathematical introduction — Vol. 27. — Springer Science and Business Media, 2012.
14. *Ермаков С. М.* К анализу метода имитации отжига в многоэкстремальном случае / С. М. Ермаков, Д. В. Куликов, С. Н. Леора // Вестник Санкт-Петербургского университета. Математика. Механика. Астрономия. — 2017. — Т. 4. — № 2. — С. 220–226. DOI: 10.21638/11701/spbu01.2017.205. — EDN ZDJRWD.
15. *Костин А. С.* Исследование моделей и методов маршрутизациии практического выполнения автономного движения беспилотными транспортными системами для доставки грузов / А. С. Костин, Н. Н. Майоров // Вестник государственного университета морского и речного флота им. адмирала С. О. Макарова. — 2023. — Т. 15. — № 3. — С. 524–536. DOI: 10.21821/2309-5180-2023-15-3-524-536. — EDN SBJQBU.

REFERENCES

1. Sakharov V, I. A. Sikarev and A. A. Chertkov “Avtomatizatsiya poiska optimal’nykh marshrutov i gruzovykh potokov v transportnykh setyakh sredstvami tselochislennogo lineynogo programmirovaniya.” *Vestnik gosudarstvennogo universiteta morskogo i rechnogo flota im. admirala S.O. Makarova* 10.3 2018:647–657. — DOI: 10.21821/2309-5180-2018-10-3-647-657.

2. Chertkov A, A. A. Vardomskaya and A. A. Dmitriev "Rekursivnyy metod optimizatsii logisticheskikh putey sredstvami MATLAB." *Vestnik gosudarstvennogo universiteta morskogo i rechnogo flota im. admirala S.O. Makarova* 6(34). 2015:196–204. — DOI: 10.21821/2309-5180-2015-7-6-196-204.

3. Chertkov A "Avtomatizatsiya vybora kratchayshikh marshrutov sudov na osnove modifitsirovannogo algoritma Bellmana-Forda." *Vestnik gosudarstvennogo universiteta morskogo i rechnogo flota im. admirala S.O. Makarova* 9.5 2017:1113–1122. — DOI: 10.21821/2309-5180-2017-9-5-1113-1122.

4. Sakharov V, A. A. Chertkov and L. B. Ochina "Marshrutizatsiya setey s otritsatel'nymi vesami zven'ev v pakete optimizatsii MATLAB." *Vestnik gosudarstvennogo universiteta morskogo i rechnogo flota im. admirala S.O. Makarova* 11.2 2019:230–242. — DOI: 10.21821/2309-5180-2019-11-2-230-242.

5. Sakharov Vladimir. V., Alexandr. A. Chertkov, Igor. B. Ariefjew. "Network routing method for ships and other moving objects using MATLAB". *Scientific Journal of the Maritime University of Szczecin* 62(134) (2020): 61–68.

6. *Hornik K*. TSP – Infrastructure for the traveling salesperson problem *Journal of statistical software* 23 (2) (2007): 1–21. DOI: 10.18637/jss.v023.i02.

7. Boroznov V "Issledovanie resheniya zadachi kommivoyazhera." *Vestnik Astrakhanskogo gosudarstvennogo tekhnicheskogo universiteta. Seriya: Upravlenie, vychislitel'naya tekhnika i informatika* 2. 2009:147–151.

8. T. Stützle, M. López-Ibáñez, P. Pellegrini, M. Maur, M. de Oca, M. Birattari, Michael Maur, M. Dorigo *Parameter Adaptation in Ant Colony Optimization.* Technical Report. IRIDIA, Université Libre de Bruxelles, 2010.

9. Cormen Thomas H., Leiserson Charles E., Rivest Ronald L., Stein Clifford. *Algorithms:* Postroenie i analiz. 2 izd. M.: Vilyams, 2011.

10. Kostjuk. Ju. L. "Effektivnaya realizatsiya algoritma resheniya zadachi kommivoayzhera metodom vetvey i granits" // Prikladnaya diskretnaya matematika, vychislitelnye metody v diskretnoy matematike, 2 (20) (2010):78-90.

11. Kirsanov, M. N. "Analiz algoritmov vybora optimalnykh marshrutov gruppy sudov." *Vestnik gosudarstvennogo universiteta morskogo i rechnogo flota im. admirala S.O. Makarova* 2(36) (2016): 183–190. DOI: 10.21821/2309-5180-2016-8-2-183-190.

12. *Metropolis V. N*., Rosenbluth A. W., Rosenbluth M. N., Teller A. H., Teller E. "Equations of state calculations by fast computing machines." *J. Chem. Phys.* 21 (1953): 1087–1092

13. Winkler, Gerhard. *Image analysis, random fields and Markov chain Monte Carlo methods: a mathematical introduction*. Vol. 27. Springer Science & Business Media, 2012.

14. Ermakov, S. M., D. V. Kulikov and S. N. Leora "K analizu metoda imitatsii otzhiga v mnogoekstremalnom sluchae." *Vestnik Sankt-Peterburgskogo universiteta. Matematika. Mekhanika. Astronomiya* 4.2 (2017): 220–226. DOI: 10.21638/11701/spbu01.2017.205.

15. Kostin, A. S. and N. N. Majorov "Issledovanie modelej i metodov marshrutizatsiii prakticheskogo vypolneniya avtonomnogo dvizheniya bespilotnymi transportnymi sistemami dlya dostavki gruzov." *Vestnik gosudarstvennogo universiteta morskogo i rechnogo flota im. admirala S.O. Makarova* 15.3 (2023): 524–536. DOI: 10.21821/2309-5180-2023-15-3-524-536.

ИНФОРМАЦИЯОБАВТОРАХ

Чертков Александр Александрович —
доктор технических наук, доцент,
ФГБОУ ВПО «ГУМРФ имени
адмирала С. О. Макарова»
198035, Российская Федерация,
Санкт-Петербург, ул. Двинская, 5/7
e-mail: *chertkov51@mail.ru*

Никифоров Владимир Григорьевич —
доктор технических наук, профессор
ФГБОУ ВО «ГУМРФ имени
адмирала С.О. Макарова»
198035, Российская Федерация,
Санкт-Петербург, ул. Двинская 5/7
e-mail: *nikiforovvg@gumrf.ru*

INFORMATION ABOUT THE AUTHORS

Chertkov, Alexandr A. —
Dr. of Technical Sciences, associate professor
Admiral Makarov State University
of Maritime and Inland Shipping
5/7 Dvinskaya Str., St. Petersburg,
198035, Russian Federation
chertkov51@mail.ru

Nikiforov Vladimir G. —
Dr. of Technical Sciences, professor,
Admiral Makarov State University
of Maritime and Inland Shipping
5/7 Dvinskaya Str., St. Petersburg,
198035, Russian Federation
e-mail: *nikiforovvg@gumrf.ru*

Статья поступила в редакцию 04 Ноября 2024
Received: Nov. 04, 2024