Геометрические методы решения псевдогеометрической версии задачи коммивояжера[1]

С. Б. Макаркин, аспирант
Б. Ф. Мельников, д. ф.-м. н.[2]
Самарский государственный университет
s.makarkin@gmail.com, bormel@rambler.ru

Одной из наиболее изученных версий задачи коммивояжера является геометрическая. В результате многолетних исследований для решения ее частных случаев было разработано множество подходов. Однако за пределами геометрической версии алгоритмы, соответствующие этим подходам, обычно неприемлемы. В данной статье рассматривается возможное обобщение геометрической версии — так называемая псевдогеометрическая — и предлагается использование для решения ее частных случаев один из стандартных геометрических алгоритмов. В статье приводятся краткое описание нашей версии алгоритма и некоторые результаты вычислительных экспериментов.

Ключевые слова: математическая модель, задача коммивояжера, геометрическая версия, псевдогеометрическая версия, геометрический подход, эвристические алгоритмы.

1. Введение. Псевдогеометрическая версия задачи коммивояжера

Нередко математическая модель, а также алгоритмы, основанные на данной модели, созданные для одной области, находят применение и во многих других предметных областях. Примером такой модели является задача коммивояжера (ниже – ЗКВ, [1–4] и мн. др.). Особенностью этой задачи является то, что при относительной простоте ее постановки нахождение оптимального решения (оптимального маршрута) является весьма сложной проблемой и относится – причем как в обобщенной ее постановке, так и для большинства ее вариаций – к классу NP-полных. Более того, согласно классификации, приведенной в [3] и др., задача коммивояжера является

[1]Работа частично поддержана грантом РФФИ № 13-01-97003 (р_поволжье_а).

примером оптимизационной проблемы, входящей в самый сложный класс NPO(V): он содержит все оптимизационные задачи, для которых (при некотором дополнительном «естественном» предположении, например, $\mathrm{P} \neq \mathrm{NP}$) временна́я сложность всех возможных полиномиальных алгоритмов не может быть ограничена никакой полилогарифмической функцией. (Другим примером подобной задачи является проблема максимальной клики.)

Приведем описание ЗКВ согласно [3].

<u>*Задача коммивояжера*</u>

- *Вход.* Полный взвешенный граф (G, c), где $G = (V, E)$ и $c : E \to \mathbb{N}_0$. Пусть $V = \{v_1, \ldots, v_n\}$, $n \in \mathbb{N}$.

- *Обозначения.* Для каждого частного случая проблемы (G, c)

$$\mathcal{M}(G, c) = \Big\{ v_{i_1}, v_{i_2}, \ldots, v_{i_n}, v_{i_1} \mid (i_1, i_2, \ldots, i_n) - \\ \text{некоторая перестановка чисел } (1, 2, \ldots, n) \Big\};$$

  т. е. $\mathcal{M}(G, c)$ можно рассматривать как множество всех Гамильтоновых циклов графа G.

- *Стоимость.* Для каждого цикла

$$H = v_{i_1} v_{i_2} \ldots v_{i_n} v_{i_1} \in \mathcal{M}(G, c)$$

  полагаем

$$cost\Big((v_{i_1}, v_{i_2}, \ldots v_{i_n}, v_{i_1}), (G, c)\Big) = \sum_{j=1}^{n} c\Big(\{v_{i_j}, v_{i_{(j \bmod n)+1}}\}\Big);$$

  т. е. стоимость каждого Гамильтонова цикла равна сумме весов всех ребер, входящих в этот цикл.

- *Цель: minimum.*

Среди множества различных версий (вариантов) ЗКВ одной из наиболее изученных является *геометрическая* (именуемая также *Евклидовой*, [3, 5] и мн. др.): стоимость маршрута равна расстоянию между точками на плоскости, вычисляемому как Евклидова норма. Для дальнейшего изложения необходимо отметить, что характерной особенностью данной задачи является выполнение неравенства

треугольника для любых трех городов – т. е. для любых $u, v, w \in E$ выполнено следующее:

$$c(\{u, v\}) \leq c(\{u, w\}) + c(\{w, v\});$$

отметим, что данная формула предполагает рассмотрение только т. н. *симметричных* вариантов ЗКВ (этот термин также определен в [3]) – однако в *не*симметричных вариантах все аналогично. Вообще, согласно [3], варианты проблемы ЗКВ с выполненным для любых трех городов неравенства треугольника называются *метрическими* – т. е. все геометрические ЗКВ представляют собой собственное подмножество метрических.

Однако основные исследования авторов данной статьи направлены на изучение не геометрической, а т. н. *псевдогеометрической* версии ЗКВ (см. [4–6] и др.), в которой входные данные формируются следующим образом. К данным некоторой *заранее заданной* ЗКВ, являющейся *геометрической*, добавляется вектор

$$R = (r_1, \ldots, r_m), \quad (\text{где} \quad m = |E|);$$

при этом все r_i – н. о. р. с. в., рассматривается нормальное распределение с $\mu = 1$ и некоторым заранее заданным «приемлемым» значением σ. При этом каждый из элементов матрицы стоимостей ($c(\{u, v\})$; пусть это c_i для некоторого $i \in \{1, \ldots, m\}$) заменяется на следующее значение:

$$\max(c_i \cdot r_i, 0).$$

Важным отличием псевдогеометрического варианта ЗКВ от геометрического является возможность нарушения неравенства треугольника для некоторых троек городов. Также важно отметить, что геометрический вариант ЗКВ можно считать частным случаем псевдогеометрического (со значением $\sigma = 0$).

Приведем наш взгляд на «экологическую нишу» рассматриваемой нами версии задачи коммивояжера. Определенная выше геометрическая версия ЗКВ является одной из наиболее изученных; в результате теоретических и практических исследований было разработано множество подходов к решению ее частных случаев (см., например, сборник трудов [7]). Среди них можно выделить т. н. геометрические подходы – использующие при поиске решения информацию о том, что координаты городов *ранее были получены* как

частный случай геометрической ЗКВ. В качестве примеров алгоритмов, относящихся к группе таких геометрических подходов и описанных 15 и более лет назад, можно привести т. н. алгоритмы "луковой шелухи" (onion-peeling, [8]) и эластичной сети ([9]). При небольшой вычислительную сложности эти подходы позволяют получить достаточно хорошее решение: для частных случаев, содержащих миллионы точек, оно практически совпадает с оптимальным. Однако за пределами геометрической версии ЗКВ эти алгоритмы обычно оказываются бессмысленными – и данную статью можно рассматривать как попытку применить подобные алгоритмы к другим версиям ЗКВ.

2. Примеры практического применения задачи коммивояжера

По поводу проблемы, рассматриваемой в данной статье, можно сказать следующее. Математические модели и алгоритмы, созданные для задачи коммивояжера, первоначально были разработаны для решения группы проблем построения маршрутов; среди таких проблем отметим:

- маршрутизацию транспортных потоков,
- выбор оптимальной траектории движения рабочего инструмента

(в обоих случаях необходимо найти оптимальный *маршрут*).

Однако в настоящее время они используются и при решении многих других прикладных проблем – в областях, на первый взгляд с маршрутизацией не связанных:

- для секвенирования нуклеотидных последовательностей биополимеров, см. [10];
- для эвристического определения схожести строк (причем являющихся как упомянутыми выше нуклеотидными последовательностями, так и каких-либо иных строк, см. [11, 12]);
- для построения практических алгоритмов исследования специально определенных бесконечных грамматических структур;

- для построения эволюционных деревьев, см. [13].

К описанным вариантам применения ЗКВ необходимы следующие комментарии.

Постановку задачи исследования бесконечных грамматических структур см. в [14] и далее в [15], оформление соответствующих задач в виде алгоритмов, в том числе эвристических, – в [15, 16]. При этом процесс сведéния данной задачи к ЗКВ, а также обоснование желательности исследования именно псевдогеометрической версии ЗКВ для данного случая являются предметом дальнейших публикаций.

А для применения задачи коммивояжера в анализе ДНК используется следующий подход. После процесса секвенирования мы получаем набор строк небольшой длины (на практике – обычно от 20 до 100, в зависимости от технологии), в которых есть совпадающие участки. Весом ребра ориентированного графа является степень сходства (где суффиксы относятся к вершинам, из которых выходят дуги, а префиксами – в которые они входят). При этом путем с максимальной стоимостью будет являться наиболее длинная общая надстрока заданной последовательности.

По мнению авторов статьи, данная постановка близка именно к рассматриваемой в данной статье псевдогеометрической версии ЗКВ в связи со следующим обстоятельством. В случае, если бы перекрывающиеся участки ДНК обязательно имели бы одинаковую длину, мы бы получали геометрическую версию ЗКВ. Однако длина перекрывающихся участков может быть произвольной (в некоторых границах).

Подробное исследование именно такой постановки псевдогеометрической версии ЗКВ (т. е. полученных именно такими способами вариантов проблемы) также является предметом дальнейших работ авторов.

3. Геометрический подход – классификации входных данных

Обычно при разработке алгоритмов используется некая идеализированная модель входных данных, чаще всего – с равномерными или нормальными распределениями значений специально выделенных *характеристик* рассматриваемой задачи. Однако на практике

входные данные, как правило, поступают в соответствии с некоторым иным вероятностным распределением – т. е. распределением, отличным от равномерного или нормального. В результате производительность алгоритма на реальных данных – причем как в среднем, так и в худшем случае – рассчитывается неадекватно входным данным.

Возможна и иная ситуация – когда при создании алгоритма учитываются особенности входных данных, характерные для одной предметной области; при этом применение таких алгоритмов для другой предметной области может оказаться крайне неэффективным из-за того, что характерные особенности реальных данных оказываются отличными от тех, которые учитывались при разработке алгоритма.

Многие ученые работают над *теорией*, описывающей алгоритмы для предположительно правильного распределения в каждом конкретном случае, которые достаточно адекватно описывали бы многие практические ситуации. По мнению авторов данной статьи, наибольший прогресс в этом направлении был достигнут в работах Ю. Гуревича ([17] и др.). В некоторых случаях оценка репрезентативности может не являться самоцелью, а служить основой для подходов к оценке эффективности алгоритмов.

По мнению авторов данной статьи, необходимы конкретные модели для каждой рассматриваемой проблемы – т. е. *конкретная интерпретация* подобных подходов, т. н. методы "ad hoc". Для реализации такого подхода нами были определены специальные характеристики, относящиеся к некоторому исходному частному случаю ЗКВ; сто́ит отметить, что они выбирались аналогично характеристикам, использовавшимся в [18] для совершенно иной задачи дискретной оптимизации. Статистические характеристики, используемые нами для ЗКВ, фактически являются существенным развитием т. н. «функции расстояния *до геометрического варианта* ЗКВ» $dist(G, c)$, описанной в [3, Ex. 4.2.3.2]; можно сказать, что эти характеристики отражают случаи невыполнения этого неравенства в заданном частном случае ЗКВ. [3]

[3] Отметим, что *на практике* наиболее интересно применение псевдогеометрического варианта ЗКВ в том случае, когда размерность задачи примерно равна 100. При этом реализация алгоритмов подсчета этих характеристик вряд ли интересна. (Сложность алгоритма, применяемого для подсчета всех этих характеристик, есть $O(n^3)$, где, как и ранее, n – размерность задачи.)

Для их подсчета, аналогично упомянутому примеру из [3], нами рассматривались все возможные пары точек $u, v \in V$ $(u \neq v)$. Для каждой из этих пар и каждой точки $p \in V$ считалось значение

$$\max\Big(0, \frac{c(\{u,v\})}{c(\{u,p\}) + c(\{p,v\})} - 1\Big), \quad p \neq u,\ p \neq v\,.$$

Для набора[4] D таких значений, а также для набора $\mathcal{D}$, составленного из ненулевых элементов набора D, мы рассматривали следующие характеристики:

- $|D|/N$, где N – максимально возможное число ненулевых элементов набора D, т. е. $\frac{n\cdot(n-1)\cdot(n-2)}{2}$;[5]
- $|\mathcal{D}|/N$;
- математическое ожидание D, где D рассматривается как реализация случайной величины;
- математическое ожидание $\mathcal{D}$;
- дисперсию $\mathcal{D}$; [6]
- среднее гармоническое набора $\mathcal{D}$;
- медиану набора $\mathcal{D}$;
- значение специального варианта критерия Ватсона ([19, 20]) для нескольких вариантов наборов $\mathcal{D}$ (т. е. при наличии различных входов).

На основе этих характеристик с помощью персептрона ([21] и др.) принималось решение о классе, для которого был сгенерирован рассматриваемый частный случай ЗКВ.

Конкретное описание вычислительных экспериментов таково. *Предварительно* была создана коллекция этих характеристик для выбранной размерности задачи (в данном случае мы всегда рассматривали 70 городов) и различных значениях σ – а именно, от

[4] Мы пользуемся следующей терминологией. Набор – это совокупность конечного числа объектов, элементы набора могут повторяться.

[5] Как было отмечено выше, мы рассматриваем симметричную ЗКВ.

[6] Для этой и следующих характеристик вряд ли имеет смысл рассматривать их аналоги для набора D.

0.00 до 2.20 с шагом 0.01; каждое из значений характеристики принималось равным среднему арифметическому соответствующих характеристик, полученных для 10 различных случайных генераций частных случаев ЗКВ. Одновременно проводилось самообучение персептрона, классифицирующего рассматриваемый частный случай на основе полученного вектора характеристик – т. е. дающего на выходе предположение об исходном значении σ.

Далее для каждого из следующих значений σ:

$$\sigma = 0.00\,; \quad \sigma = 0.20\,; \quad \sigma = 0.50\,; \quad \sigma = 0.90\,; \quad \sigma = 1.40\,; \quad \sigma = 2.00$$

проводилось по 100 экспериментов. В каждом из них мы генерировали частный случай ЗКВ, после чего *на основе вектора характеристик, полученного для этого частного случая*, выбирали предположение об исходном значении σ двумя способами:

- на основе максимального значения вероятности, выдаваемого ранее обученным персептроном;
- с помощью метода наименьших квадратов.

Кратко результаты экспериментов изложены в следующей таблице 1. В ней первому столбцу (Р) соответствует результаты, полученные с помощью персептрона. В каждой клетке записан отрезок значений σ, в который попадали полученные результаты, а также число случаев (из 100), в которых полученное значение σ отличалось от исходного не более чем на 10%.

Таблица 1.

σ	Р	МНК
0.00	$[0.00, 0.00]$ 100	$[0.00, 0.00]$ 100
0.20	$[0.16, 0.23]$ 83	$[0.16, 0.24]$ 81
0.50	$[0.43, 0.59]$ 81	$[0.42, 0.58]$ 80
0.90	$[0.76, 1.03]$ 79	$[0.76, 1.02]$ 79
1.40	$[1.20, 1.57]$ 78	$[1.22, 1.55]$ 78
2.00	$[1.74, 2.20]$ 79	$[1.76, 2.20]$ 79

Итак, по-видимому, можно сказать, что *оба* варианта классификации дают приемлемые результаты – которые практически не зависят от конкретного способа классификации (с помощью нейросети или с помощью МНК). Важно также то, что оба алгоритма можно считать *комбинацией*: во-первых, *подхода, основанного*

на правилах (rule-based approach) и, во-вторых, *альтернативного ему подхода* (not rule-based approach), для которого в литературе на русском языке еще нет окончательного названия.[7] Подход, основанный на правилах, заключается в выборе характеристик *экспертом*, а альтернативный подход – в апостериорном использовании нейросети или МНК.

Однако как более подробные результаты приведенных выше вычислений, так и возможное улучшение обоих применявшихся нами алгоритмов классификации вряд ли представляет большой интерес: как уже было отмечено, эти вычисления (т. е. классификацию) сто́ит воспринимать как вспомогательную задачу для другой, более важной задачи, рассматриваемой в следующем разделе.

4. Геометрический подход – псевдооптимальное размещение точек

Итак, значительно более важным является не проблема классификации (иными словами – задача восстановления класса для неизвестного нам значения σ по известной матрице частного случая ЗКВ), а псевдовосстановление координат исходного множества точек. Перед описанием применяемого нами алгоритма этого псевдовосстановления рассмотрим значительно более простую («школьную») задачу – восстановление расположения точек в *геометрической* версии ЗКВ по матрице расстояний, заданной с помощью функции $c : E \to \mathbb{N}_0$ (с точностью до преобразований плоскости, сохраняющих расстояния). Очевидно, что эта задача может быть решена с помощью следующего тривиального алгоритма.

Сначала из множества вершин графа V выбираются две произвольные точки, v_1 и v_2. Первую точку (v_1) мы располагаем в начале координат, а вторую (v_1) – в точке $(0, c(v_1, v_2))$. Координаты каждой из следующих точек (пусть это u с координатами (x, y)) можно

[7] На самом деле, окончательного названия нет и в литературе на английском языке: часто используемые слова “not rule-based approach” вряд ли на него «претендуют». Возможными альтернативами подходу, основанному на правилах, *в разных предметных областях* являются т. н. коннекционистский подход (connectionism), структурный подход, а, например, в случае машинного перевода – т. н. статистический перевод.

По поводу последнего сто́ит отметить его использование в сервисе «Яндекс.Перевод», а также появление в последнее время *гибридных* систем перевода, фактически и являющихся *комбинацией* двух вышеупомянутых подходов.

найти, выбрав из множества уже размещенных точек две (пусть v и w) и решив простейшую систему уравнений

$$\begin{cases} (x - x_v)^2 + (y - y_v)^2 = (c(u, v))^2 \\ (x - x_w)^2 + (y - y_w)^2 = (c(u, w))^2 \end{cases} \tag{1}$$

(При этом начиная с четвертой точки необходимы дополнительные вычисления, определяющие, какое именно из двух решений системы (1) необходимо выбрать. Однако мы не будем подробно описывать эти действия, а также не будем рассматривать некоторые ситуации, в которых выбираемые 3 точки оказываются лежащими на одной прямой, и т. п. Мы также не будем рассматривать вопросы, связанные с точностью вычислений для этого алгоритма.)

Очевидно, что при применении этого алгоритма к матрице расстояний $c : E \to \mathbb{N}_0$, заданной *для псевдогеометрического варианта* ЗКВ, мы (вообще говоря) не получаем исходное расположение точек для того геометрического варианта ЗКВ, на основе которого был сгенерирован исходный (т. е. полученный основным алгоритмом в качестве входа) случай псевдогеометрического варианта ЗКВ. Более того, вследствие возможного нарушения в заданной матрице расстояний неравенства треугольника, данный алгоритм может вообще оказаться неприменимым. Однако, как уже было сказано выше, основным предметом данной статьи является предлагаемый авторами алгоритм решения частного случая ЗКВ, заключающийся в применении *модифицированного* алгоритма восстановления расположения точек в случае геометрического варианта ЗКВ. [8]

Итак, несмотря на то, что применение алгоритмов, подобных вышеописанному, в псевдогеометрическом варианте ЗКВ, вообще говоря, невозможно, мы пытаемся *теми же самыми алгоритмами* решить задачу расположения городов, фактически решая минимизационную задачу для специальным образом вычисляемой невязки

$$\sqrt{\frac{2}{n \cdot (n-1)} \cdot \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(c(u_i, u_j) - \widetilde{c}(u_i, u_j)\right)^2}\ , \tag{2}$$

где:

[8] Конечно же, такой описываемый далее алгоритм может быть применен *к любому* частному случаю ЗКВ. Однако в большинстве «случайных» (и отличных от псевдогеотметрических) частных случаях ЗКВ его применение вряд ли целесообразно.

- u_i $(i = 1, \ldots, n)$ – точки; ниже координаты i-й точки будем обозначать (x_i, y_i);
- $c(u_i, u_j)$ – (i, j)-й элемент *исходной* матрицы расстояний;
- $\widetilde{c}(u_i, u_j)$ – (i, j)-й элемент *полученной* матрицы расстояний.

В качестве *эвристического алгоритма для этой задачи*[9] авторами предлагается следующий.

Алгоритм псевдооптимального размещения точек

- *Вход.* Матрица $c : E \to \mathbb{N}_0$ (матрица весов ребер полного взвешенного графа с вершинами $V = \{u_1, \ldots, u_n\}$), некоторое число $N \in \mathbb{N}$.[10]
- *Шаг 1.* $x_1 := 0$; $y_1 := 0$; $x_2 := 0$; $y_2 := c(u_1, u_2)$;
- *Шаг 2.* Для каждой точки $k = 3, \ldots, n$ (будем также обозначать текущую рассматриваемую k-ю точку с помощью u) вычислить координаты (x_k, y_k) путем выполнения описанных далее шагов 3–7.
- *Шаг 3.* $M := \min\left(\frac{(k-1)\cdot(k-2)}{2}, N\right)$;
- *Шаг 4.* Для каждого значения $l = 1, \ldots, M$ выполнить описанные далее шаг 5 и шаг 6.

---

[9] Приведем (с некоторыми сокращениями) отрывок из [2].

> Термин «эвристика» ... используется в различных значениях.
>
> Во-первых, в общем смысле эвристические алгоритмы являются непротиворечивыми алгоритмами для решения оптимизационных задач. Они основаны на какой-нибудь достаточно простой идее поиска во множестве всех допустимых решений – которая, однако, не гарантирует, что какое-нибудь оптимальное решение найдется. ...
>
> А в узком смысле ... эвристика является методом получения непротиворечивого алгоритма, который в типичных частных случаях рассматриваемых оптимизационных проблем действительно дает допустимые решения приемлемого качества за приемлемое время – например, за полиномиальное; но *доказать* этот факт обычно невозможно.

Несмотря на то, что эти два толкования понятия «эвристика» на первый взгляд могут показаться взаимно противоречивыми, мы в разных местах настоящей статьи используем этот термин *в обоих* смыслах.

[10] N – количество выбираемых пар из *уже размещенных* точек. Эти пары выбираются для размещения каждой новой точки, начиная с 3-й.

- *Шаг 5.* Случайно выбрать равномерно распределенные значения $i, j \in \{1, \ldots, k-1\}$, причем $i \neq j$. Будем i-ю точку называть также v, а j-ю – w, и, кроме того, обозначать $c_1 = c(u, v)$, $c_2 = c(u, w)$.

- *Шаг 6.* Если $c_1 + c_2 \leq c(v, w)$ (т. е. для точек u, v и w нарушается неравенство треугольника), то выбираем

$$x_u = \frac{c_1 x_w + c_2 x_v}{c_1 + c_2}, \quad y_u = \frac{c_1 y_w + c_2 y_v}{c_1 + c_2}$$

  (в вырожденной ситуации, т. е. когда одновременно $c_1 = 0$ и $c_2 = 0$, берем точку посередине отрезка $[v, w]$) – т. е. выбираем координаты между i-й и j-й точками пропорционально значениям соответствующих элементов матрицы стоимостей). [11]
  В противном случае (т. е. при $c_1 + c_2 > c(v, w)$):

  – вычисляем 2 варианта пар x_k и y_k согласно системе (1);
  – если $k = 3$, то выбираем произвольную пару координат, иначе выбираем пару по одной случайно выбранной точке среди точек с номерами $1, \ldots, k-1$, отличными от i и j; [12]
  – выбранную пару координат добавляем в коллекцию пар.

- *Шаг 7.* Вычисляем окончательные значения x_k и y_k как средние арифметические значений l соответствующих координат сформированной коллекции. [13]

- *Выход.* Координаты $(x_1, y_1), \ldots, (x_n, y_n)$.

Можно сказать, что в приведенном алгоритме процитированная выше цель – применение достаточно простой идеи поиска во

[11] Отметим еще, что некоторые следствия возможного нарушения треугольника фактически уже рассматривались выше – например, в прошлом разделе.

[12] Расстояние от этой случайно выбранной точки (пусть это p) до «правильного» значения u должно быть ближе к значению $c(u, p)$, чем аналогичное расстояние для «проигнорированного» значения u.

[13] Здесь приведена упрощенная версия алгоритма; на самом деле, мы применяем значительно более сложные алгоритмы усреднения, аналогичные применению описанных в [22] динамических функций риска. Отметим, что применение последних для произвольных задач дискретной оптимизации (а не только при программировании недетерминированных игр) было описано одним из авторов настоящей статьи в [4, 6] и др.

множестве всех допустимых решений – заключается в *последовательном* применении расположения точек $u_1, \ldots, u_n$ путем минимизации значения

$$\sum_{i=1}^{k-2} \sum_{j=i+1}^{k-1} \left(c(u_i, u_j) - \widetilde{c}(u_i, u_j)\right)^2,$$

входящего в формулу (2). После выполнения этой операции (т. е. после размещения всех точек) применяются стандартные методы решения геометрического варианта ЗКВ (например, один из алгоритмов «луковой шелухи», упомянутый выше) – см. некоторые подробности его применения в [23].

5. Некоторые результаты вычислительных экспериментов

Приведем краткие результаты вычислительных экспериментов.

Вычисления производились для размерностей от 20 до 100 с шагом 20. При этом для каждой размерности проводилось не менее 100 случайных генераций псевдогеометрической версии задачи коммивояжера (при этом выбиралось единственное значение $\sigma = 0.2$) – после чего применялись 7 эвристических алгоритмов. [14]

Алгоритмы *для заданного конкретного варианта ЗКВ* применялись следующим образом. Сначала запускался алгоритм имитационной нормализации (имитации отжига [15]) и фиксировалось время его работы; отметим, что среднее время работы этого алгоритма невелико (относительно других рассматриваемых нами эвристических алгоритмов). После этого запускались остальные алгоритмы

[14] Конкретные описания тех эвристических алгоритмов, которые *не* являются предметом настоящей статьи, приведены в [24–26] и др. Другие ссылки, а также более подробные результаты вычислительных экспериментов (для большего количества значений размерности задачи и σ) и программные реализации соответствующих алгоритмов, приведены в нескольких дипломных работах, а также в нескольких магистерских и одной кандидатской диссертации, защищенных под руководством одного из авторов настоящей статьи в 2008-2013 г.г. в Тольяттинском государственном университете. См. также [27] и подробнее [28] – в первую очередь описание гибридного алгоритма. Отметим, что в упомянутых работах приведены не только относительные результаты (как в настоящей статье), но и абсолютные – т. е. приведено среднее время выполнения конкретных задач для конкретных конфигураций вычислительных устройств.

[15] Замечания о терминологии см. в [2, стр. 256].

– причем в тех из них, в которых необходимо временнóе ограничение (т. е. в т. н. anytime-алгоритмах, [4] и др.), максимальное время работы устанавливалось в 20 раз бóльшим, чем время работы алгоритма имитационной нормализации.

Все результаты вычислений сравнивались с результатами использованного нами варианта генетического алгоритма – дающего, как правило, наименее хорошие результаты. А именно, фиксировался результат работы генетического алгоритма (т. е. значение найденного тура для заданного варианта ЗКВ); результат работы каждого из остальных алгоритмов рассматривался в отношении к результату работы генетического алгоритма. Все отношения усреднялись (бралось среднее геометрическое этих отношений) для всех экспериментов рассматриваемой размерности. [16]

Итак, нами применялись варианты следующих эвристических алгоритмов [17] :

- генетического алгоритма (ГА);
- алгоритма имитационной нормализации (ИН);
- алгоритма локального поиска (Лок);
- незавершенного метода ветвей и границ (МВГ);
- гибридного алгоритма – комбинации алгоритмов имитационной нормализации, локального поиска и незавершенного метода ветвей и границ (Гиб);
- муравьиного алгоритма (М);
- геометрического алгоритма (Геом) – являющегося предметом настоящей статьи

(в скобках приведены сокращения, используемые далее в таблице). Результаты вычислительных экспериментов кратко описываются таблицей 2.

[16] Нами также рассматривался вариант усреднения, не учитывающий лучшие 10% и худшие 10% результатов, – однако итоговые значения для всех случаев оказались практически такими же.

[17] Некоторые ссылки на публикации с описаниями алгоритмов были приведены выше.

dim	ГА	ИН	Лок	МВГ	Гиб	М	Геом
20	1	0.94	0.82	0.87	0.80	0.47	0.53
40	1	0.91	0.80	0.87	0.77	0.46	0.57
60	1	0.92	0.79	0.85	0.75	0.47	0.56
80	1	0.90	0.81	0.84	0.75	0.46	0.57
100	1	0.90	0.79	0.83	0.74	0.49	0.58

Таб. 2

Как мы видим, результаты мало зависят от рассматривавшихся размерностей задачи. Некоторое обсуждение результатов см. далее.

6. Заключение

Итак, в данной статье описано применение «геометрического подхода» к частным случаям задачи коммивояжера, не являющимся, вообще говоря, примерами ее геометрической версии. Результаты вычислительных экспериментов оказались лучше всех рассматривавшихся нами подходов – кроме муравьиного алгоритма. Однако время работы муравьиного алгоритма (необходимое для достижения таких результатов) существенно превышает время работы описанного нами «геометрического подхода», и, кроме того, возможное продолжение работ в данном направлении дает надежду на улучшение результатов.

Продолжение работ в данном направлении предполагает решение нескольких различных больших задач. Это, в первую очередь, исследование репрезентативности предлагаемой нами модели входных данных для ЗКВ; предполагается, что это исследование значительно превзойдет примеры, описанные выше в разделе 2, и будет аналогично исследованию репрезентативности недетерминированных конечных автоматов, кратко описанному одним из авторов в [18]. [18] Кроме того, предполагается рассмотреть перенос результатов настоящей статьи на *несимметричную* (причем также псевдогеометрическую) версию ЗКВ. Но еще более важным авторам представляется улучшение алгоритмов, приведенных в настоящей статье.

В качестве возможных вариантов такого улучшения нами в настоящее время рассматриваются следующие. Во-первых, примене-

[18] Некоторые возможные *характеристики* рассматриваемой нами предметной области уже фактически были предложены в настоящей статье.

ние различных эвристик для усреднения координат, выбранным на основе различных пар ранее расположенных точек. Во-вторых, более полное применение *различных* алгоритмов (прежде всего – алгоритмов псевдовосстановления точек) – в зависимости от предположенного (с помощью персептрона) значения σ (которое ранее было использовано при генерации рассматриваемого частного случая ЗКВ). В-третьих, возможное изменение применяемой нами формулы невязки (2). В-четвертых, замена алгоритма «луковой шелухи» на какой-либо другой алгоритм – однако также «геометрический». И, в-пятых, возможное применение в качестве вспомогательных некоторых из алгоритмов, приведенных в таблице 2.

Однако *наиболее перспективным* эвристическим алгоритмом, который авторы предполагают *добавить* к уже реализованным, по-видимому, является следующий. Мы будем рассматривать *несколько* случайно сгенерированных перестановок всего множества точек – и для каждой из этих перестановок применять вышеописанный алгоритм псевдовосстановления их расположения. Далее мы должны для каждой точки выбрать единственный вариант расположения – а *в данной ситуации* любые усреднения, конечно же, бессмысленны. Усреднения (например, просто вычисление среднего арифметического) возможны после решения оптимизационной задачи – заключающейся в повороте сгенерированного множества точек на некоторый угол и смещения на некоторый вектор. В этой задаче минимизируется сумма расстояний между одноименными точками (либо какая-нибудь аналогичная величина). Решать эту задачу целесообразно каким-либо градиентным методом, для которого переменными являются угол поворота и координаты вектора смещения. По-видимому, можно также воспользоваться каким-либо из алгоритмов, предназначенных для обработки изображений.

Мы предполагаем опубликовать результаты дальнейшей работы в статье-продолжении.

Список литературы

[1] *Гэри М., Джонсон М.* Вычислительные машины и труднорешаемые задачи. – Пер. с англ. – М.: Мир. 1982. 416 с.

[2] *Громкович Ю.* Теоретическая информатика. Введение в теорию автоматов, теорию вычислимости, теорию сложности, тео-

рию алгоритмов, рандомизацию, теорию связи и криптографию. – Пер. с нем. – СПб.: Изд-во БХВ-Петербург. 2010. 326 с.

[3] *Hromkovič J.* Algorithmics for Hard Problems. Introduction to Combinatorial Optimization, Randomization, Approximation, and Heuristics. Springer. 2003. 538 p.

[4] *Melnikov B.* Multiheuristic approach to discrete optimization problems // Cybernetics and Systems Analysis. 2006. Vol. 42. No. 3. Sept. P. 335–341.

[5] *Melnikov B., Radionov A., Gumayunov V.* Some special heuristics for discrete optimization problems // Proc. of 8th International Conference on Enterprise Information Systems, ICEIS-2006. P. 360–364.

[6] *Мельников Б., Романов Н.* Еще раз об эвристиках для задачи коммивояжера // Теоретические проблемы информатики и ее приложений. 2001. № 4. С. 81–92.

[7] *Gutin G., Punnen A. (editors)* The Traveling Salesman problem. – Boston, Kluwer Academic Publishers. 856 p.

[8] `http://arxiv.org/ftp/arxiv/papers/1204/1204.2350.pdf`. – Liew Sing. Introducing convex layers to the Traveling Salesman Problem // Preprint: arXiv:1204.2348. – 2012. – Режим доступа – свободный. – [Интернет-ресурс].

[9] *Somhom S., Modares A., Enkawa T.* Competition-based neural network for the multiple travelling salesmen problem with minimax objective // Computers & Operations Research. 1999. Vol. 26. No. 4. Sept. P. 395–407.

[10] *Korostensky C., Gonnet G.* Near optimal multiple sequence alignments using a travelling salesman problem approach // String Processing and Information Retrieval Symposium ŕŕŕ& International Workshop on Groupware. ŕŕŕ1999. P. 105–114.

[11] *Мельников Б., Панин А.* Параллельная реализация мультиэвристического подхода в задаче сравнения генетических последовательностей // Вектор науки Тольяттинского государственного университета. 2012. №. 4. С. 83–86.

[12] *Makarkin S., Melnikov B., Panin A.* On the metaheuristics approach to the problem of genetic sequence comparison and its parallel implementation // Applied Mathematics. 2013. Vol. 4. No 10A. Sept. P. 35–39.

[13] *Korostensky C., Gonnet G.* Using traveling salesman problem algorithms for evolutionary tree construction // Bioinformatics. 2000. Vol. 16. P. 619–627.

[14] *Melnikov B., Kashlakova E.* Some grammatical structures of programming languages as simple bracketed languages // Informatica (Lithuania). 2000. Vol. 11. No 4. P. 441–453.

[15] *Алексеева А., Мельников Б.* Итерации конечных и бесконечных языков и недетерминированные конечные автоматы // Вектор науки Тольяттинского государственного университета. 2011. № 3. С. 30–33.

[16] *Мельников Б.* Алгоритм проверки равенства бесконечных итераций конечных языков // Вестник Московского университета. Серия 15: Вычислительная математика и кибернетика. 1996. № 4. С. 49–53.

[17] *Gurevich Y., Veanes M., Wallace C.* Can abstract state machines be useful in language theory? // Theoretical Computer Science (Developments in Language Theory). 2007. Vol. 376. No. 1–2. Sept. P. 17–29.

[18] *Мельников Б., Пивнева С., Рогова О.* Репрезентативность случайно сгенерированных недетерминированных конечных автоматов с точки зрения соответствующих базисных автоматов // Стохастическая оптимизация в информатике. 2010. Т. 6. № 1. С. 74–82.

[19] *Watson G.* Goodness-of-fit tests on a circle // Biometrika. 1961. Vol. 48. No 1–2. P. 109–114; 1962. Vol. 49. No 1–2. P. 57–63.

[20] *Лагутин М.* Наглядная математическая статистика. – М.: Бином. 2012. 472 с.

[21] *Хайкин С.* Нейронные сети: полный курс. – Пер. с англ. – М.: Вильямс. 2008. 1104 с.

[22] *Melnikov B.* Heuristics in programming of nondeterministic games // Programming and Computer Software. 2001. Vol. 27. No 5. P. 277–288.

[23] *Макаркин С.* Еще об одном подходе к решению псевдогеометрической задачи коммивояжера // Вектор науки Тольяттинского государственного университета. 2012. № 4. С. 79–83.

[24] *Макконнел Дж.* Основы современных алгоритмов. – М.: Техносфера. 2004. 368 с.

[25] *Штовба С.* Муравьиные алгоритмы. – Exponenta Pro. Математика в приложениях. 2004. №. 4.

[26] *Кузюрин Н.,Фомин С.* Эффективные алгоритмы. – М.: МФТИ. 2007. 313 с.

[27] *Мельников Б., Эйрих С.* Подход к комбинированию незавершенного метода ветвей и границ и алгоритма имитационной нормализации // Вестник Воронежского государственного университета. Серия: Системный анализ и информационные технологии. 2010. № 1. С. 35–38.

[28] *Эйрих С.* Применение имитационной нормализации в гибридных алгоритмах. – Дисс. . . . канд. физ.-мат. наук: 05.13.18, Тольяттинский государственный университет, Д 212.264.03, защищена 21.02.2012. 124 с.