

ПРИМЕНЕНИЕ ДОПОЛНЕНИЙ ПАРСОЧЕТАНИЯМИ ДЛЯ РЕШЕНИЯ ЗАДАЧИ MAX TSP

А.В. Панюков, С.А. Тычинин

Изложен подход к приближенному решению задачи коммивояжера на максимум (MAX TSP), основанный на дополнении частичного тура паросочетаниями подграфа открытых вершин. Проведено аналитическое исследование, показавшее, что алгоритм, основанный на непосредственном применении данного подхода, во-первых, имеет вычислительную сложность не более $O(n^3)$, n – число городов, во-вторых, не улучшает гарантированные оценки точности известных алгоритмов. Предложена модификация алгоритма Сердюкова, имеющая оценку вычислительной сложности $O(n^3)$ и наилучшую гарантированную оценку точности. Представлены результаты вычислительного эксперимента, позволяющие выдвинуть гипотезу об асимптотической точности алгоритма для достаточно широкого класса задач.

Ключевые слова: *гамильтонов цикл, задача коммивояжера, приближенный алгоритм, паросочетание, оценки точности, вычислительная сложность, вычислительный эксперимент*

Введение

Пусть $G = (V, E)$ – полный неориентированный граф с множеством вершин $V : |V| = n$ и множеством ребер E . Пусть для каждого ребра $e \in E$ задан его вес $c(e)$. Для множества ребер $E' \subseteq E$ будем обозначать $c(E') = \sum_{e \in E'} c(e)$. Гамильтонов цикл – это цикл, проходящий через каждую вершину графа ровно один раз. Задача коммивояжера на максимум (MAX TSP) заключается в нахождении в графе G гамильтонового цикла максимального веса.

Подмножество ребер некоторого гамильтонова цикла называют частичным туром. Вершины частичного тура, инцидентные двум ребрам, называют закрытыми, остальные вершины называют открытыми. Подмножество $E' \subseteq E$ ребер G называется паросочетанием (совершенным паросочетанием), если каждая вершина $v \in V$ инцидентна не более (соответственно ровно) одному ребру E' . Подмножество $E' \subseteq E$ ребер графа G называется 2-фактором, если каждая вершина $v \in V$ инцидентна ровно двум ребрам в E' . Для отыскания обоих суграфов максимального веса известны алгоритмы с временной сложностью $O(n^3)$ [6, 7].

Будем говорить, что алгоритм имеет оценку точности ρ , если отношение длины найденного гамильтонового тура к длине оптимального маршрута не меньше ρ . Рассматриваемая задача MAX SNP-трудна [4], то есть из существования полиномиальной аппроксимационной схемы (PTAS) для ее решения следует, что $P = NP$. Оценки точности лучших на сегодняшний момент алгоритмов для приближенного решения симметрической и метрической MAX TSP приведены в таблице.

Точность алгоритмов для MAX TSP

Название алгоритма	Оценка точности
Симметрическая TSP	
Алгоритм Сердюкова [1]	3/4
Алгоритм HR1 [9]	25/33 *
Алгоритм 61/81 [5]	61/81 **
Метрическая TSP	
Алгоритм Косточки – Сердюкова [8]	5/6
Алгоритм HR2 [8]	7/8 **
Алгоритм 17/20 [5]	17/20 *

Алгоритмы, оценки точности которых помечены звездочками, являются рандомизированными, а пометка двумя звездочками означает асимптотичность оценки. И в том, и в другом случае оценки точности не являются гарантированными.

В данной работе для решения задачи MAX TSP представлены анонсированные ранее алгоритм [3] и модификация алгоритма Сердюкова [2]. Данные алгоритмы предварительно строят специальные частичные туры, а затем их дополняют до гамильтонового цикла паросочетаниями подграфов открытых вершин. Гарантированные оценки точности для алгоритма [3] – 2/3 для симметрической TSP, и 7/9 для метрической. Оценка точности модификации алгоритма Сердюкова – 3/4 для симметрической TSP, и 5/6 для метрической. Вычислительная сложность всех алгоритмов не превосходит величины $O(n^3)$.

Алгоритм дополнения паросочетаниями

В основе разработанного алгоритма лежит идея дополнения частичного тура до гамильтонового цикла ребрами паросочетаний подграфа на открытых вершинах.

Поясним эту идею. Первым шагом алгоритма является построение 2-фактора максимального веса. После этого производится разрыв полученных циклов удалением ребра минимального веса. Далее выполняется процедура дополнения полученного частичного тура паросочетанием максимального веса на подграфе графа G индуцированным множеством открытых вершин этого частичного тура. В результате будет получен 2-фактор с числом компонент связности, по крайней мере, в 2 раза меньшим, чем у первоначального 2-фактора. Алгоритм выполняет указанную процедуру до тех пор, пока очередной 2-фактор не будет иметь одну компоненту связности. В этом случае полученный цикл будет гамильтоновым.

Дадим иллюстрацию применения данного алгоритма. Пусть даны граф G и его частичный тур перед выполнением процедуры такие, как показано на рис. 1 а. Толстыми линиями на рисунке изображен частичный тур, ребра графа не нарисованы, так как предполагается, что граф полный. Вершины, инцидентные менее чем двум ребрам частичного тура (открыты t вершины), изображены на рисунках без штриховой заливки. Теперь стоит задача дополнить текущий частичный тур до гамильтонового цикла, причем сделать это наиболее оптимальным образом.

Рассмотрим подграф H исходного графа, получаемый при удалении всех закрытых вершин текущего частичного тура. Для каждой цепи частичного тура исходного графа также удалим ребро графа H , соответствующее ребру G , замыкающему эту цепь. В случае, если цепь состоит из одного ребра, соответствующее ребро удаляется из графа H . На рис. 1 б изображен граф H , полученный в результате этой процедуры. Пунктирными линиями изображены отсутствующие в графе H ребра, которые в графе G замыкали цепи частичного тура. На следующем шаге найдем паросочетание максимального веса в графе H (рис. 1

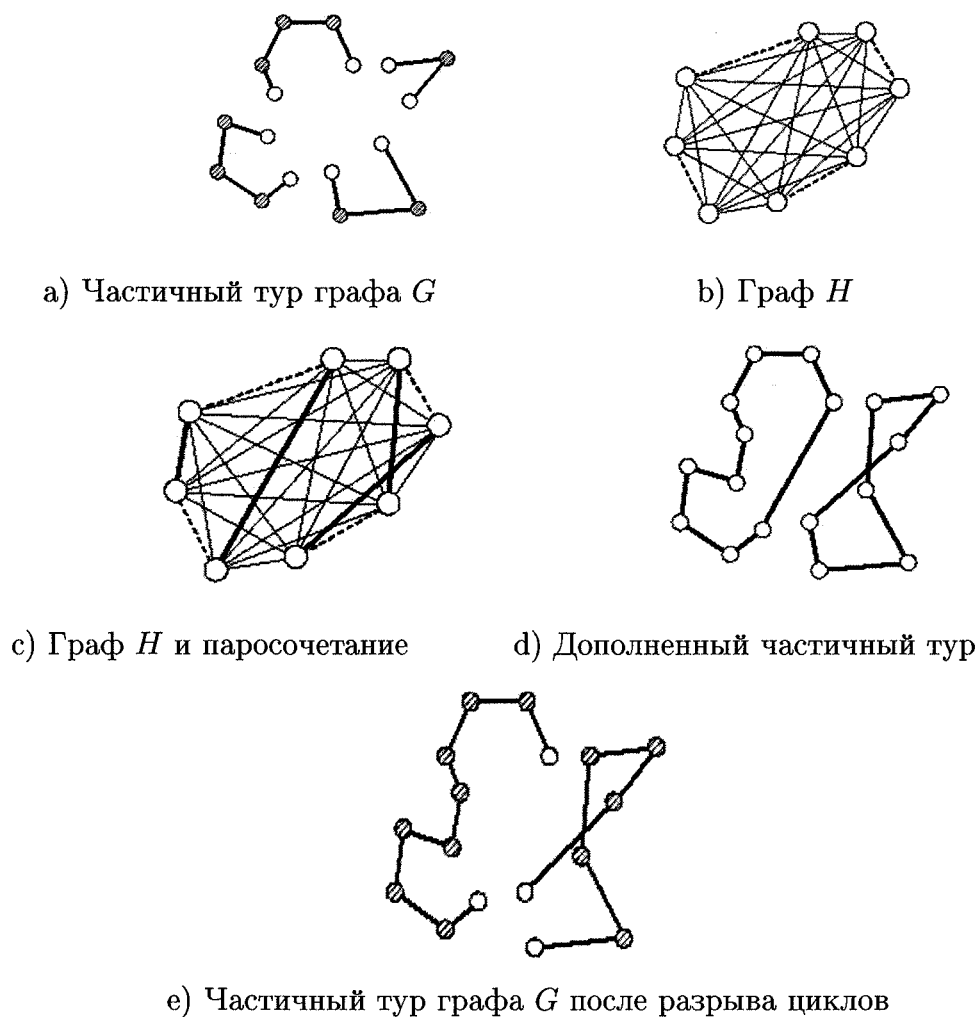


Рис. 1

с). Включим ребра, входящие в паросочетание в частичный тур графа G (рис. 1 d), и, если в частичном туре возникнут циклы длины меньше $|V(G)|$, то удалим из каждого ребро наименьшего веса (рис. 1 e). В результате этой процедуры количество открытых вершин в графе G будет уменьшаться, по крайней мере, на величину, равную количеству компонент связности в частичном туре (следовательно, как минимум вдвое). Если в результате еще остались открытые вершины (частичный тур не является гамильтоновым циклом), то следует выполнить эту процедуру еще раз. Приведем формальное описание алгоритма.

Алгоритм А1 (Алгоритм дополнения паросочетаниями).

Вход: полный неориентированный граф $G = (V, E)$ с весовой функцией c .

Выход: гамильтонов цикл.

Шаг 1. Вычислим 2-фактор максимального веса $C = C_1, \dots, C_s$.

Шаг 2. Если $s = 1$ вернуть C_1 .

Шаг 3. Для каждого C_i : $i = 1, \dots, s$ найти $e_i = \{u_i, v_i\} = \arg \min_{e \in C_i} c(e)$.

Положить $C_i = C_i \setminus \{e_i\}$.

Шаг 4. Пусть множество $S = \bigcup_{i=1, \dots, s} \{u_i, v_i\}$.

Пусть H подграф графа G , образованный вершинами множества S .

Положить $E(H) = E(H) \setminus \left(\bigcup_{i=1, \dots, s} e_i \right)$.

Шаг 5. Найдем в графе H паросочетание максимального веса M .

Шаг 6. Пусть $C = C \cup M$. Обозначить непересекающиеся циклы, образованные ребрами множества C , через C_1, \dots, C_s . Перейти на **Шаг 2**.

Конец алгоритма.

Вычислительная сложность алгоритма

В наихудшем случае 2-фактор будет состоять из наибольшего возможного числа циклов длины 3, и после разрыва получится $\lfloor 2 \cdot n/3 \rfloor$ вершины, которым инцидентно только одно ребро частичного тура (так называемые открытые вершины). В этом случае количество компонент связности частичного тура равно $\lfloor n/3 \rfloor$. В наихудшем случае в результате одной процедуры построения дополнительного графа количество открытых вершин будет уменьшено лишь вдвое. Так как алгоритм заканчивает работу, когда не останется открытых вершин, то количество выполнения шагов 3–5 алгоритма будет не более $\lceil \log_2 \lfloor 2 \cdot n/3 \rfloor \rceil$. Поскольку при каждом последующем выполнении шагов 3–5 размерность задачи понижается как минимум в два раза, то время выполнения этих шагов не будет превосходить величины

$$\sum_{k=1}^{\lceil \log_2 \lfloor 2 \cdot n/3 \rfloor \rceil} O\left(\frac{n}{2^k}\right)^3 = O(n^3).$$

Таким образом, алгоритм **A1** является полиномиальным и при эффективной реализации подзадач нахождения максимальных паросочетаний и 2-фактора имеет вычислительную сложность $O(n^3)$.

Оценка точности алгоритма

Теорема 1. Пусть opt – вес оптимального гамильтонового цикла в неориентированном взвешенном графе G при решении задачи на максимум, а T – цикл, найденный с помощью алгоритма **A1** дополнения подграфами. Тогда выполняются следующее неравенство:

$$c(T) \geq \frac{2}{3} \cdot opt.$$

Данная оценка неулучшаема.

Доказательство. Пусть C – 2-фактор максимального веса, найденный на шаге 1 работы алгоритма. Тогда выполняется неравенство

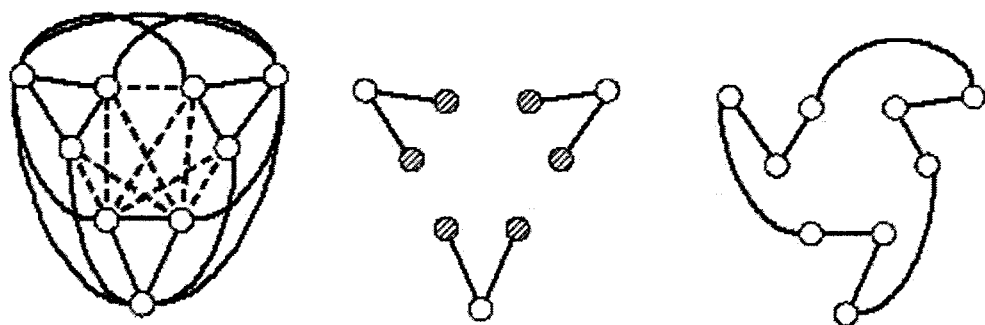
$$c(C) \geq opt,$$

поскольку гамильтонов цикл также является 2-фактором. При разрыве циклов 2-фактора каждый цикл теряет не более трети своего веса, поскольку цикл состоит не менее, чем из трех вершин, и исключается ребро цикла минимального веса. Следовательно,

$$c(T) \geq \frac{2}{3} \cdot opt.$$

Неравенство доказано.

Для доказательства неулучшаемости данной оценки рассмотрим граф, изображенный на рис. 2 а. Штриховыми линиями изображены ребра, имеющие вес 0, остальные ребра имеют вес 1. Найденный 2-фактор будет иметь вес 9, а вес, полученного из него частичного тура, равен 6 (рис. 2 б). Дальнейшее дополнение частичного тура будет производиться ребрами



а) Граф G б) Частичный тур в G в) Оптимальное решение

Рис. 2

веса 0, следовательно, полученный гамильтонов цикл будет иметь вес 6. В то же время в графе G существует очевидный оптимальный гамильтонов цикл веса 9 (рис. 2 в).

Следовательно, использование алгоритма дополнения паросочетаниями для данного графа может привести к нахождению решения аппроксимирующего оптимальное с коэффициентом $2/3$ ($6 = 2/3 \cdot 9$), а, значит, гарантируемая оценка точности $2/3$ неулучшаема. \square

Теорема 2. Пусть opt – вес оптимального гамильтонового цикла в метрическом неориентированном взвешенном графе G при решении задачи на максимум, пусть T – цикл, найденный с помощью алгоритма **A1**. Тогда выполняются следующее неравенство:

$$c(T) > \frac{7}{9} \cdot opt.$$

Доказательство. Пусть (u_i, v_i) – ребро, удаленное из цикла C_i на втором шаге алгоритма. Рассмотрим цикл длины 3, образованный вершинами u_i, v_i и w , где $w \notin C_i$. Из неравенства треугольника следует, что

$$c(u_i, w) + c(w, v_i) \geq c(u_i, v_i). \quad (1)$$

Число вершин в графе H равно $2s$ (s – это число циклов в 2-факторе C). Просуммировав по всем w , получим

$$\sum_{w \in V(H), w \neq u_i, w \neq v_i} (c(u_i, w) + c(w, v_i)) \geq (2s - 2) \cdot c(u_i, v_i). \quad (2)$$

Теперь, просуммировав (2) по всем i , получим:

$$2 \cdot c(H) \geq (2s - 2) \cdot c. \quad (3)$$

где $c(H)$ – сумма весов всех ребер графа H , а $c = \sum_{i=1}^s c(u_i, v_i)$.

Так как граф H получен из полного графа удалением s ребер, число ребер в нем равно:

$$|E(H)| = \frac{2s \cdot (2s - 1)}{2} - s = 2s(s - 1). \quad (4)$$

В паросочетание M графа H входит s ребер, и, следовательно, вес паросочетания будет ограничен снизу следующей величиной:

$$c(M) \geq \frac{c(H)}{(2s - 2)} \geq \frac{(2s - 2) \cdot c}{2(2s - 2)} = \frac{c}{2}. \quad (5)$$

Неравенство (5) получено исходя из того, что граф H является регулярным со степенями вершин $2s - 2$, и, следовательно, множество его ребер может быть разбито на $2s - 2$ попарно непересекающихся паросочетания. Значит, в результате выполнения одной процедуры дополнения вес частичного тура T_1 до разрыва циклов станет равным

$$c(T_1) \geq c(C) - c + \frac{c}{2}. \quad (6)$$

Если образовались циклы, то их разрыв приведет к уменьшению частичного тура на величину не большую, чем $c/4$, так как даже в худшем случае вес удаленных ребер не будет превышать половины веса построенного паросочетания.

После этого вторая процедура дополнения скомпенсирует, по крайней мере, половину веса удаленных ребер. В итоге, если придется выполнить m процедур дополнения, то вес частичного тура будет ограничен снизу следующей величиной:

$$c(T_m) \geq c(C) - c + \frac{c}{2} - \frac{c}{4} + \frac{c}{8} - \dots - \frac{c}{2^{m-1}} + \frac{c}{2^m} = c(C) - \frac{c}{2} - \frac{c}{8} - \dots - \frac{c}{2^m}. \quad (7)$$

В наихудшем случае 2-фактор C состоит из цепей длины 3 и, следовательно,

$$c \leq opt/3.$$

В результате, перейдя в неравенстве (7) к пределу, получим:

$$c(T) \geq opt - \frac{c}{2} - \frac{c}{8} - \dots - \frac{c}{2^m} = opt - \frac{c/2}{1 - 1/4} = opt - \frac{2}{3}c \geq opt - \frac{2}{9}opt = \frac{7}{9}opt. \quad (8)$$

□

Данный алгоритм может быть эффективно использован вместе со многими другими алгоритмами для дополнения частичного тура до гамильтонового. Продемонстрируем данную технику на примере алгоритма Сердюкова. В оригинальном алгоритме Сердюкова способ дополнения частичных туров до гамильтоновых не регламентируется и, как правило, подразумевает использование жадного алгоритма. С целью сохранения целостности изложения приведем предварительно изложение алгоритма Сердюкова с использованием принятых в статье обозначений.

Алгоритм Сердюкова

Для симметрической задачи коммивояжера одним из лучших алгоритмов является алгоритм Сердюкова с оценкой точности $3/4$ [6]. В этом алгоритме используется нахождение максимального паросочетания, и необходимо рассмотреть 2 случая: в графе четное число вершин, либо нечетное.

Алгоритм А2-0 (n – четно, в этом случае может быть найдено максимальное паросочетание, которое будет покрывать все вершины графа).

Вход: полный неориентированный граф $G = (V, E)$ с весовой функцией c . $|V|$ – четно.

Выход: Гамильтонов цикл.

Шаг 1. Вычислить максимальный 2-фактор $C = C_1, \dots, C_s$. Если $s = 1$ вернуть C_1 .

Вычислить максимальное паросочетание M .

Шаг 2. Для каждого цикла C_i , $i = 1, \dots, s$ переместить из C_i в M ребро, которое сохранит M частичным туром.

Шаг 3. Дополнить C до гамильтонова цикла T_1 .

Дополнить M до гамильтонова цикла T_2 .

Вернуть больший по весу тур из T_1 и T_2 .

Конец алгоритма.

Алгоритм А2-1 (n – нечетно, в этом случае для графа не существует совершенно-го паросочетания, поэтому после нахождения максимального 2-фактора и максимального паросочетания необходимо определенным образом изменить эти подграфы и использовать алгоритм, предназначенный для четного размера задачи).

Вход: полный неориентированный граф $G = (V, E)$ с весовой функцией c . $|V|$ – нечетно.

Выход: Гамильтонов цикл.

Шаг 1. Вычислить максимальный 2-фактор $C = C_1, \dots, C_s$. Если $s = 1$ вернуть C_1 .

Вычислить максимальное паросочетание M .

Шаг 2. Пусть v^0 – вершина, не инцидентная никакому ребру из M .

Пусть $v^0 \in C_1$. Найти ребро $e_0 = \{v^0, v'\} = \arg \max_{e \in E \setminus E(C)} c(e)$.

Пусть $e_1, e_2 \in C_1$ – ребра, инцидентные вершине v^0 .

Если $v' \in C_1$, то положить $C_0 = C_1$ и $j = 2$.

Если $v' \in C_2$, то положить $C_0 = C_1 \cup C_2$ и $j = 3$.

В $C_0 \setminus M$ выбрать ребра $e' \in \{e_1, e_2\}$ и e'' , инцидентные вершине v' . В случае $C_0 = C_1$ следует выбрать такое ребро e' , чтобы не образовался цикл через вершину v^0 .

Обозначить $C = e_0 \cup C \setminus \{e', e''\}$, $M = M \cup \{e', e''\}$.

Шаг 3. Для $i = j, \dots, s$: Переместить из C_i в M ребро, которое сохранит M частичным туром.

Шаг 4. Дополнить C и M до гамильтоновых циклов T_1 и T_2 , соответственно.

Вернуть больший по весу тур между T_1 и T_2 .

Конец алгоритма.

Как видно из изложенного в алгоритме Сердюкова, способ дополнения частичных туров до гамильтоновых не регламентируется. Здесь, как правило, подразумевается использование жадного алгоритма. Рассмотрим модификацию алгоритма Сердюкова [2], основанную на дополнении частичных туров до гамильтоновых с использованием паросочетаний подграфов открытых вершин.

Модификация алгоритма Сердюкова

Разработана модификация алгоритма Сердюкова [2], позволяющая получить на каждом графе цикл не меньшего веса, чем с помощью оригинального алгоритма [1].

Изменения по сравнению с алгоритмом Сердюкова следующие:

1. Вычисляется не два, а пять гамильтоновых циклов;
2. Один гамильтонов цикл вычисляется с помощью алгоритма А1;
3. Два цикла получаются дополнением текущих частичных туров C и M до гамильтонового цикла с помощью жадного алгоритма;
4. Оставшиеся два цикла дополнением текущих частичных туров C и M до гамильтонового цикла паросочетаниями подграфов открытых вершин.

Алгоритм А3-0 (n – четно, в этом случае может быть найдено максимальное паросочетание, которое будет покрывать все вершины графа).

Вход: полный неориентированный граф $G = (V, E)$ с весовой функцией c . $|V|$ – четно.

Выход: Гамильтонов цикл.

Шаг 1. Вычислить максимальный 2-фактор $C = C_1, \dots, C_s$. Если $s = 1$ вернуть C_1 .

Вычислить максимальное паросочетание M .

Шаг 2-1. Положить $D = C$. Из каждого цикла 2-фактора D удалить ребро минимального веса.

Шаг 2-2. Для каждого цикла C_i , $i = 1, \dots, s$ переместить из C_i в M ребро, которое сохранит M частичным туром.

Шаг 3-1. Дополнить частичные туры C , M и D паросочетаниями графов открытых вершин до гамильтоновых циклов T_1 , T_2 и T_3 соответственно.

Шаг 3-2. Дополнить алгоритмом координатного подъема частичные туры C и M до гамильтоновых циклов T_4 и T_5 соответственно.

Шаг 4. Вернуть больший по весу тур из T_1 , T_2 , T_3 , T_4 и T_5 .

Конец алгоритма.

Алгоритм АЗ-1 (n – нечетно, в этом случае для графа не существует совершенного паросочетания, по-этому после нахождения максимального 2-фактора и максимального паросочетания необходимо определенным образом изменить эти подграфы и использовать алгоритм, предназначенный для четного размера задачи).

Вход: полный неориентированный граф $G = (V, E)$ с весовой функцией c . $|V|$ – нечетно.

Выход: Гамильтонов цикл.

Шаг 1. Вычислить максимальный 2-фактор $C = C_1, \dots, C_s$. Если $s = 1$ вернуть C_1 .

Вычислить максимальное паросочетание M .

Шаг 2-1. Положить $D = C$. Из каждого цикла 2-фактора D удалить ребро минимального веса.

Шаг 2-2. Пусть v^0 – вершина, не инцидентную никакому ребру из M .

Пусть $v^0 \in C_1$. Найти ребро $e_0 = \{v^0, v'\} = \arg \max_{e \in E \setminus E(C)} c(e)$.

Пусть $e_1, e_2 \in C_1$ – ребра, инцидентные вершине v^0 .

Если $v' \in C_1$, то положить $C_0 = C_1$ и $j = 2$.

Если $v' \in C_2$, то положить $C_0 = C_1 \cup C_2$ и $j = 3$.

В $C_0 \setminus M$ выбрать ребра $e' \in \{e_1, e_2\}$ и e'' , инцидентные вершине v' .

В случае $C_0 = C_1$ следует выбрать такое ребро e' , чтобы не образовался цикл через вершину v^0 .

Обозначить $C = e_0 \cup C \setminus \{e', e''\}$, $M = M \cup \{e', e''\}$.

Шаг 2-3. Для $i = j, \dots, s$: Переместить из C_i в M ребро, которое сохранит M частичным туром.

Шаг 3-1. Дополнить частичные туры C , M и D паросочетаниями графов открытых вершин до гамильтоновых циклов T_1 , T_2 и T_3 соответственно.

Шаг 3-2. Дополнить алгоритмом координатного подъема частичные туры C и M до гамильтоновых циклов T_4 и T_5 соответственно.

Шаг 4. Вернуть больший по весу тур из T_1 , T_2 , T_3 , T_4 и T_5 .

Конец алгоритма.

Результатом работы алгоритма **АЗ** является гамильтонов тур T с длиной

$$c(T) \geq \frac{3}{4} opt.$$

Этот факт объясняется тем, что алгоритм основывается на алгоритме Сердюкова с такой оценкой точности и для любого графа находит решение не хуже исходного алгоритма.

Вычислительный эксперимент

Для оценки эффективности алгоритмов был проведен вычислительный эксперимент. Вычисления производились на полных графах со случайными весами, равномерно распределенными на промежутке $[0, 100]$ и с числом вершин 10, 20, 40, 80 и 160. Для каждого из указанных числа вершин генерировалось 40 случайных графов. В итоге алгоритм дополне-

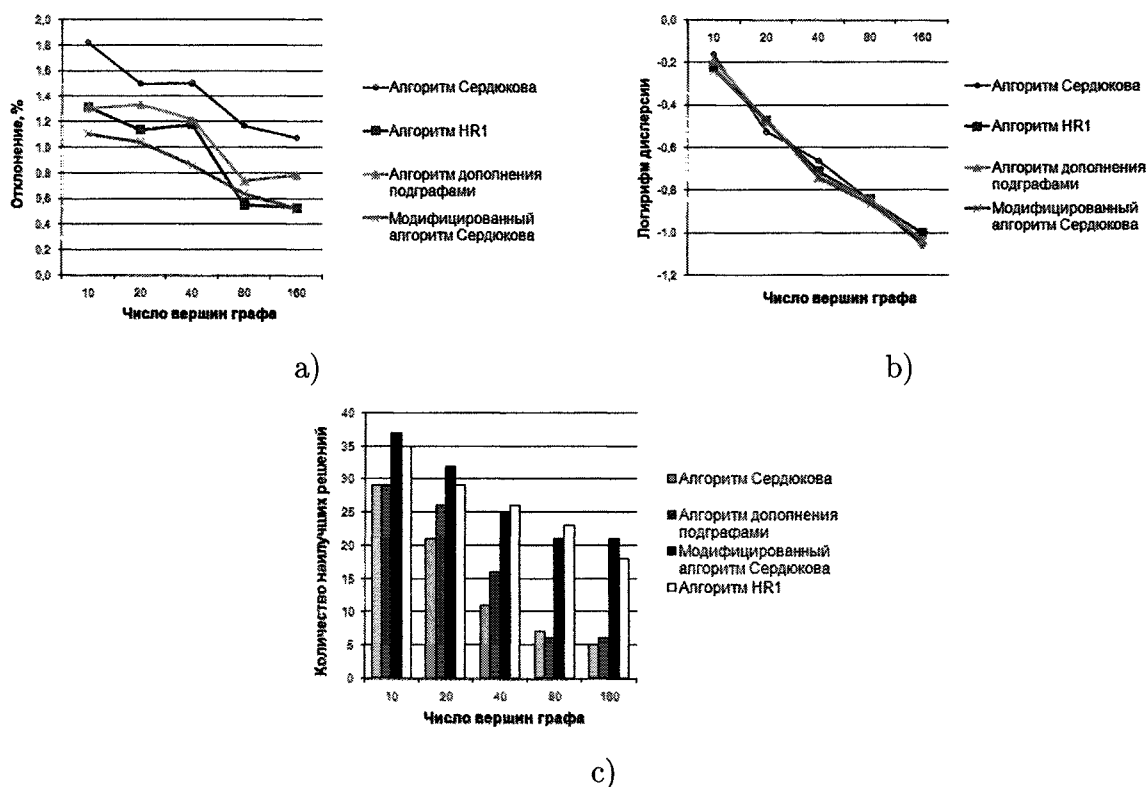


Рис. 3

ния паросочетаниями показали лучшие после алгоритма HR1 результаты, опередив на практике алгоритмы с более высокими теоретическими оценками. Лучшие результаты показал модифицированный алгоритм Сердюкова.

На рис. 3 а приведен график средних отклонений от точного решения для каждого алгоритма. В качестве верхней оценки точного решения брался минимум из веса максимального 2-фактора и удвоенного веса максимального паросочетания. Как видно из рисунка, погрешности всех алгоритмов имеют тенденцию к уменьшению с ростом числа вершин, при этом лучшие результаты, в основном, дает модифицированный алгоритм Сердюкова. Только для 80-вершинных графов ему составляет конкуренцию алгоритм HR1.

На рис. 3 б приведен график логарифма дисперсии отклонений от точного решения. Из графика видно, что дисперсия погрешности модифицированного алгоритма Сердюкова ниже соответствующей дисперсии алгоритма HR1.

Количество решений максимального веса, найденных каждым из исследуемых алгоритмов представлено на рис. 3 с. Легко заметить, что с ростом размерности частоты нахождения максимального решения модифицированным алгоритмом Сердюкова и алгоритмом HR1 примерно равны.

Оценивая результаты проведенного вычислительного эксперимента в целом, можно сделать вывод, что в подавляющем большинстве случаев модифицированный алгоритм Сердюкова дает более надежные результаты, чем остальные алгоритмы. Следовательно, использование данного алгоритма для приближенного решения задачи коммивояжера наиболее оправдано.

Заключение

Предложенный в статье алгоритм приближенного решения задачи коммивояжера на максимум, основанный на дополнении частичного тура паросочетаниями подграфа откры-

тых вершин, имеет вычислительную сложность $O(n^3)$ и оценки точности $2/3$ для симметрической MAX TSP и $7/9$ для метрической. Несмотря на то, что известные гарантированные оценки не улучшены, вычислительный эксперимент показал, что получаемые данным алгоритмом решения являются одними из самых точных среди приближенных алгоритмов. В связи с этим достаточно эффективной оказалась связка разработанного алгоритма с алгоритмом, имеющим наиболее высокую из известных гарантированную оценку точности – алгоритмом Сердюкова. Предложенная в работе модификация алгоритма Сердюкова по результатам вычислительного эксперимента показала наилучшие результаты среди известных полиномиальных алгоритмов для MAX TSP. Представленные результаты вычислительного эксперимента, позволяют выдвинуть гипотезу об асимптотической точности алгоритма **A3** для достаточно широкого класса задач.

Литература

1. Гимади, Э.Х. О некоторых результатах для задачи коммивояжера на максимум / Э.Х. Гимади, А.И. Сердюков // Дискретный анализ и исследование операций. – 2001. – № 1. – С. 22 – 39.
2. Панюков, А.В. Исследование реализаций алгоритма Сердюкова для задачи MAX TSP / А.В. Панюков, С.А. Тычинин // Российская конференция «Дискретная оптимизация и исследование операций»: материалы конф. (Владивосток, 7 – 14 сентября 2007). – Новосибирск, 2007. – С. 132.
3. Тычинин, С.А. Алгоритм дополнения подграфами для решения задачи MAX TSP / С.А. Тычинин // Информационный бюллетень ассоциации математического программирования № 11: Конференция «Математическое программирование и приложения (тезисы докладов)». – Екатеринбург, 2007. – С. 217 – 218.
4. The maximum traveling salesman problem under polyhedral norms / A.I. Barvinok, D.S. Johnson, G. Woeginger, R. Woodroffe // Integer Programming and Combinatorial Optimization. – Berlin, 1999. – P. 195 – 201.
5. Chen, Zh. Improved Deterministic Approximation Algorithms for Max TSP / Zh. Chen, Y. Okamoto, L. Wang // Inform. Process. Lett. – 2005. – № 95. – P. 333 – 342.
6. Gabow, H. An Efficient Implementation of Edmonds' Algorithm for Maximum Matching on Graphs / H. Gabow // J. of the ACM. – 1976. – № 4. – P. 221 – 234.
7. Hartvigsen, D. Extensions of matching theory: PhD Thesis / D. Hartvigsen – Pittsburg, PA: Carnegie Mellon Univ, 1984. – 148 p.
8. Hassin, R. A $7/8$ -approximation algorithm for metric Max TSP / R. Hassin, S. Rubinstein // Inform. Process. Lett. – 2002. – № 81. – P. 247 – 251.
9. Hassin, R. Better approximations for Max TSP / R. Hassin, S. Rubinstein // Inform. Process. Lett. – 2000. – № 75. – P. 181 – 186.

Кафедра экономико-математических методов и статистики,
Южно-Уральский государственный университет
pav@susu.ac.ru

Поступила в редакцию 27 февраля 2008 г.