ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Июль—август 2009. Том 16, № 4. С. 3–20

УДК 519.178

# ПРИБЛИЖЁННЫЙ АЛГОРИТМ РЕШЕНИЯ МЕТРИЧЕСКОЙ ЗАДАЧИ О ДВУХ КОММИВОЯЖЁРАХ С ОЦЕНКОЙ ТОЧНОСТИ 2 [*)]

*А. А. Агеев, А. В. Пяткин*

**Аннотация.** В задаче $m$-PSP требуется в заданном $n$-вершинном полном неориентированном взвешенном графе найти $m$ непересекающихся гамильтоновых циклов наименьшего суммарного веса. Эта задача была впервые рассмотрена Крарупом в 1974 г.; она имеет применения в задачах дизайна сетей и теории расписаний. Известно, что задача 2-PSP NP-трудна даже в метрическом случае, а в общем случае для неё не существует приближённого алгоритма с точностью, ограниченной константным множителем. Бабурин, Гимади и Коркишко (2004) предложили приближённый алгоритм с точностью $(9/4+\varepsilon)$ для метрической задачи 2-PSP, основанный на решении задачи коммивояжёра. В настоящей статье представлен улучшенный приближённый алгоритм с точностью 2 и временем работы $O(n^2 \log n)$ для метрической задачи 2-PSP. Этот алгоритм использует тот факт, что задача поиска двух непересекающихся остовных деревьев минимального суммарного веса является полиномиально разрешимой.

**Ключевые слова:** приближённый алгоритм, гамильтонов цикл, остовное дерево, задача коммивояжёра.

## Введение

В задаче об $m$ непересекающихся маршрутах комивояжёра ($m$-PSP, от английского Peripatetic Salesman Problem) заданы полный неориентированный граф $G = (V, E)$ с $n$ вершинами и неотрицательная функция весов его рёбер $w : E \to \mathbb{R}_+$. Требуется найти $m$ рёберно не пересекающихся гамильтоновых циклов $C_1, \ldots, C_m \subset E$, минимизируя $\sum\limits_{k=1}^{m} \sum\limits_{e \in C_k} w(e)$.

[*)] Исследование выполнено при финансовой поддержке Российского фонда фундаментальных исследований (проекты 08–01–00370 и 08–01–00516).

© 2009 Агеев А. А., Пяткин А. В.

Задача 1-PSP совпадает с задачей коммивояжёра (TSP). Область применения задачи $m$-PSP включает поиск маршрута сторожей [12], где обычно бывает важно назначить сторожам рёберно не пересекающиеся маршруты, чтобы избежать повторных прохождений и тем самым повысить уровень безопасности. Де Корт [9] показывает возможность применения задачи при дизайне сетей, где нужно определить несколько рёберно не пересекающихся циклов для защиты сети от сбоев. Де Корт также упоминает о применении задачи 2-PSP в теории расписаний, где каждая работа должна быть выполнена дважды на одной машине, но технологические ограничения не допускают повторения одинаковых последовательностей работ.

Де Корт [9] доказал, что задача 2-PSP NP-трудна, построив полиномиальное сведение к ней задачи о гамильтоновом пути. Аналогичными аргументами можно показать NP-трудность задачи $m$-PSP для любого $m > 2$. Де Брей и Волпутфте [5] выделили полиномиально разрешимые случаи задачи 2-PSP. Де Корт [7–9] определил и проанализировал верхние и нижние оценки, которые можно использовать для решения задачи 2-PSP методом ветвей и границ. Дюшен, Лапорт и Семет [6] предложили полиэдральный подход для решения задачи $m$-PSP.

Легко показать (используя сведение из [9] и те же аргументы, что использовались для задачи коммивояжёра), что в случае произвольных весов рёбер задача 2-PSP не может быть решена приближённым алгоритмом с точностью, ограниченной константным множителем. Для случая, когда веса удовлетворяют неравенству треугольника (метрическая задача 2-PSP) А. Е. Бабурин, Э. Х. Гимади и Н. М. Коркишко [2] предложили приближённый алгоритм с точностью $(9/4 + \varepsilon)$.

Как и в задаче коммивояжёра, максимизационная версия задачи 2-PSP даже в случае произвольных весов рёбер допускает решение приближённым алгоритмом с точностью, ограниченной константным множителем. Наилучший известный результат в настоящее время принадлежит А. А. Агееву, А. Е. Бабурину и Э. Х. Гимади [1], которые представили приближённый алгоритм решения этой задачи с точностью $3/4$.

В настоящей статье представлен приближённый алгоритм для решения метрической задачи 2-PSP с точностью 2. Время работы алгоритма ограничено величиной $O(n^2 \log n)$. Известный «фольклорный» приближённый алгоритм с точностью 2 для метрической задачи коммивояжёра использует тот факт, что для заданного остовного дерева $T$ можно легко построить гамильтонов цикл $C$, вес которого не более чем вдвое превосходит вес $T$ (цикл строится из любого обхода дерева $T$). Наш алгоритм

для задачи 2-PSP базируется на дальнейшем развитии этого подхода. Роскинд и Тарьян [11] установили, что в заданном $n$-вершинном рёберно взвешенном полном графе $G$ можно за время $O(n^2 \log n)$ найти два непересекающихся остовных дерева минимального суммарного веса. Первая идея, которая приходит в голову в связи с этим, — попробовать построить обходы этих деревьев, которые произведут два непересекающихся гамильтоновых цикла, что дало бы приближённый алгоритм с точностью 2 для задачи 2-PSP. Однако это не всегда можно сделать: для некоторых пар непересекающихся остовных деревьев таких обходов не существует. Наш алгоритм преодолевает эту трудность, модифицируя исходную пару остовных деревьев с помощью обмена рёбер между ними.

## 1. Алгоритм: общая схема

Поскольку задача 2-PSP не имеет решения для $n < 5$, в дальнейшем предполагается, что $n \geqslant 5$.

Алгоритм начинает работу с нахождения двух непересекающихся остовных деревьев $T_1^*$ и $T_2^*$ минимального суммарного веса (используя алгоритм Роскинда и Тарьяна). Затем он строит первый цикл $C_1$ (фаза 1). Во время фазы 1 пара деревьев $(T_1^*, T_2^*)$ может трансформироваться, и в конце мы можем получить другую пару деревьев $(T_1, T_2)$, удовлетворяющую соотношению $T_1 \cup T_2 = T_1^* \cup T_2^*$. Цикл $C_1$ не пересекается с $T_2$, а его вес не более чем вдвое превышает вес $T_1$.

Второй гамильтонов цикл $C_2$ (не пересекающийся с $C_1$) строится во время фазы 2. В отличие от фазы 1, во время этой фазы дерево $T_2$ не изменяется (никакие обмены рёбрами между $C_1$ и $T_2$ невозможны).

Хотя описание фаз 1 и 2 достаточно сложно, они выполняются за меньшее время, чем фаза 0, так что общее время работы алгоритма можно оценить как $O(n^2 \log n)$.

Ниже общая схема нашего алгоритма представлена формально. Напомним, что граф $G$ называется *внешнепланарным*, если его можно изобразить на плоскости так, чтобы никакие два ребра не пересекались кроме как в вершинах и все вершины графа $G$ лежали на внешней грани.

Алгоритм `Disj_Ham_Cycles`

Фаза 0. Используя алгоритм Роскинда и Тарьяна [11], найти два непересекающихся остовных дерева $T_1^*$ и $T_2^*$ наименьшего суммарного веса.

Фаза 1. Найти гамильтонов цикл $C_1$ и два непересекающихся остовных дерева $T_1$ и $T_2$, удовлетворяющих следующим свойствам:

1) $T_1 \cup T_2 = T_1^* \cup T_2^*$;

2) $T_2 \cap C_1 = \varnothing$;

3) граф $C_1 \cup T_1$ является внешнепланарным с внешней гранью $C_1$.

Фаза 2. Найти такой гамильтонов цикл $C_2$, что

1) $C_1 \cap C_2 = \varnothing$;

2) граф $C_2 \cup T_2$ является внешнепланарным с внешней гранью $C_2$.

Вывести $C_1$ и $C_2$.

Для выполнения фазы 0 используется алгоритм Роскинда и Тарьяна [11], который находит $k$ непересекающихся остовных деревьев минимального суммарного веса за время $O(n^2 \log n + k^2 n^2)$. Таким образом, фаза 0 выполняется за время $O(n^2 \log n)$.

Подробное описание фаз 1 и 2 с необходимым теоретическим обоснованием и оценками трудоёмкости даётся в следующих двух разделах.

Однако общего описания Алгоритма `Disj_Ham_Cycles` достаточно для доказательства оценки его точности.

**Лемма 1.** *Вес допустимого решения задачи 2-PSP, найденного алгоритмом* `Disj_Ham_Cycles`*, не более чем в 2 раза превышает вес оптимального решения.*

Доказательство. Очевидно, $C_1$ и $C_2$ не пересекаются. Покажем, что $w(C_1)+w(C_2) \leqslant 2(w(C_1^*)+w(C_2^*))$, где $C_1^*, C_2^*$ — оптимальное решение задачи 2-PSP. Ясно, что $w(C_1^*) + w(C_2^*) \geqslant w(T_1^*) + w(T_2^*)$. Оценим $w(C_i)$. Пусть $i \in \{1, 2\}$ и $e$ является ребром цикла $C_i$, не принадлежащим дереву $T_i$. Обозначим через $G_i$ внешнепланарный граф $C_i \cup T_i$, а через $F(e)$ — рёбра внутренней грани, содержащей $e$. Из неравенства треугольника следует, что вес ребра $e$ можно оценить сверху через сумму весов рёбер единственного пути в дереве $T_i$, соединяющего концы ребра $e$. Так как $G_i$ внешнепланарный, а $T_i$ — остовное дерево, то каждая грань $G_i$ содержит ровно одно ребро не из $T_i$. Значит, путь в $T_i$, соединяющий концы ребра $e$ состоит в точности из всех рёбер из $F(e) \setminus \{e\}$. Каждая хорда графа $G_i$ принадлежит двум внутренним граням, так что её вес считается дважды. Каждое ребро из $C_i \cap T_i$ принадлежит одной внутренней грани, поэтому его вес считается один раз как для ребра пути и один раз как для ребра $C_i$. Следовательно,

$$w(C_1)+w(C_2) \leqslant 2(w(T_1)+w(T_2)) = 2(w(T_1^*)+w(T_2^*)) \leqslant 2(w(C_1^*)+w(C_2^*)).$$

Лемма 1 доказана.

## 2. Выполнение фазы 1

Общая идея такова: сначала разбить дерево $T_1^*$ на несколько относительно небольших поддеревьев, затем для каждого из них найти подходя-

щие циклы (обмениваясь рёбрами между $T_1^*$ и $T_2^*$, если это необходимо) и, наконец, соорудить из этих циклов искомый цикл $C_1$.

Сначала приведём теоретическое обоснование существования цикла $C_1$ с необходимыми свойствами, а затем представим алгоритм, который следует из этого доказательства.

Пусть $H$ — неориентированный граф. Через $V(H)$ обозначим множество вершин графа $H$. Для подмножества $V' \subseteq V(H)$ обозначим через $H[V']$ подграф графа $H$, индуцированный множеством $V'$.

Рассмотрим два непересекающихся остовных дерева $T_1$ и $T_2$ в исходном графе $G = (V, E)$. Пусть множество $V' \subseteq V$ выбрано так, что $T = T_1[V']$ является остовным деревом подграфа $G[V']$. Тогда $F = T_2[V']$ является лесом в $G[V']$, и $T$ не пересекается с $F$. Будем говорить, что $T$ является *$F$-обходимым*, если в подграфе $G[V']$ существуют остовное дерево $T'$, лес $F'$ и гамильтонов цикл $C$, удовлетворяющие следующим условиям:

1) $T' \cup F' = T \cup F$;

2) $(T_1 \setminus T) \cup T'$ и $(T_2 \setminus F) \cup F'$ являются остовными деревьями в $G$;

3) $F' \cap C = \varnothing$;

4) граф $T' \cup C$ является внешнепланарным с внешней гранью $C$.

Дерево $T$ называется *обходимым*, если оно $F$-обходимо для любого леса $F$, такого что $T$ и $F$ могут быть продолжены до некоторых непересекающихся остовных деревьев $T_1$ и $T_2$ графа $G$.

Основным результатом данного раздела является следующая

**Теорема 1.** *Каждое дерево с не менее чем* 5 *вершинами является обходимым.*

Алгоритмическое доказательство этой теоремы базируется на следующих леммах.

**Лемма 2.** *Пусть $D_1$ и $D_2$ — два обходимых дерева с непересекающимися множествами вершин. Пусть $v_1 \in D_1$ и $v_2 \in D_2$ — произвольные вершины этих деревьев. Обозначим через $D$ дерево, полученное из $D_1$ и $D_2$ отождествлением вершин $v_1$ и $v_2$ в вершину $v$* (*рис.* 1). *Тогда дерево $D$ обходимо.*

Доказательство. Пусть $V = V[D]$, $V_i = V[D_i]$, $i = 1, 2$. Рассмотрим произвольный лес $F$ на множестве вершин $V$, такой что $D$ и $F$ можно продолжить до непересекающихся остовных деревьев $T_1$ и $T_2$ графа $G$. Покажем, что $D$ является $F$-обходимым. Обозначим через $F_1$ ограничение леса $F$ на $V_1$. По условию леммы существуют дерево $D_1'$, лес $F_1'$ и цикл $C_1$, удовлетворяющие условиям 1)–4) из определения обходимых

деревьев. По условию 2) $(F \setminus F_1) \cup F_1'$ является лесом, а $(D \setminus D_1) \cup D_1'$ — остовным деревом в $V[D]$ и они могут быть продолжены до непересекающихся остовных деревьев $T_1' = (T_1 \setminus D_1) \cup D_1'$ и $T_2' = (T_2 \setminus F_1) \cup F_1'$. Обозначим через $F_2$ ограничение $F$ на $V_2$. Поскольку $D_2$ обходимо, существуют дерево $D_2'$, лес $F_2'$ и цикл $C_2$, удовлетворяющие условиям 1)–4) из определения обходимых деревьев, где вместо $T_1$ и $T_2$ стоят соответственно $T_1'$ и $T_2'$. Пусть $F' = (F \setminus (F_1 \cup F_2)) \cup F_1' \cup F_2'$ и $D' = (D \setminus (D_1 \cup D_2)) \cup D_1' \cup D_2'$. Отметим, что $D'$ и $F'$ удовлетворяют условиям 1) и 2). Теперь построим искомый цикл $C$. Обозначим через $x_i$ и $y_i$ соседей вершины $v$ в $C_i$, $i = 1, 2$. Тогда по крайней мере одно из рёбер $x_1x_2, x_1y_2, y_1y_2$ или $y_1x_2$ не лежит в $F'$. Из соображений симметрии можно считать, что $x_1x_2 \notin F$. Пусть $C = (C_1 \cup C_2 \cup x_1x_2) \setminus \{x_1v, x_2v\}$. Тогда $C \cup F' = \varnothing$ и граф $C \cup D'$ внешнепланарный, причём $C$ является его внешней гранью, т. е. $D$ $F$-обходим. Из произвольности леса $F$ следует, что $D$ является обходимым. Лемма 2 доказана.

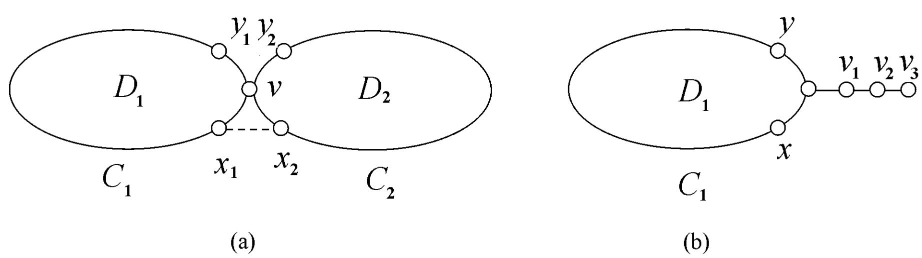


Рис. 1. Шаги индукции в леммах 2 и 3

**Лемма 3.** *Пусть $D_1$ — обходимое дерево, $v \in D_1$, а дерево $D$ получается из $D_1$ добавлением вершин $v_1, v_2, v_3$ и рёбер $vv_1, v_1v_2, v_2v_3$ (рис. 1b). Тогда дерево $D$ обходимо.*

Доказательство. Пусть $V_1 = V(D_1)$ и $V = V(D)$. Обозначим путь $vv_1v_2v_3$ через $P$. Рассмотрим произвольный лес $F$ на множестве вершин $V$ ($D$ и $F$ можно продолжить до непересекающихся остовных деревьев $T_1$ и $T_2$ графа $G$) и обозначим через $F_1$ и $F_2$ его ограничения на $V_1$ и $V(P)$ соответственно. Если $v_3v \notin F$, то $P$ очевидно $F_2$-обходим (достаточно положить $C_2 = P \cup v_3v$), и из тех же соображений, что приведены в лемме 2, $D$ является $F$-обходимым. Допустим, что $v_3v \in F$. Так как $D_1$ обходим, существуют дерево $D_1'$, лес $F_1'$ и цикл $C_1$, удовлетворяющие условиям 1)–4). Пусть $F' = (F \setminus F_1) \cup F_1'$ и $D' = (D \setminus D_1) \cup D_1'$. Обозначим через $x$ и $y$ соседей вершины $v$ в цикле $C_1$. Если $v_3x \notin F$, то положим $C = (C_1 \cup P \cup v_3x) \setminus vx$. Очевидно, $F', D'$ и $C$ удовлетворяют условиям 1)–4). Если $v_3y \notin F$, действуем аналогично. Таким образом, $v_3x, v_3y \in F$.

Тогда по крайней мере одно из рёбер $v_2x, v_2y$ не лежит в $F$ (скажем, $v_2x \notin F$). Если $v_3v_1 \notin F$, то цикл $C$ получается из $C_1$ удалением ребра $vx$ и добавлением пути $vv_1v_3v_2x$. И снова $F', D'$ и $C$ удовлетворяют условиям 1)–4). Наконец, предположим, что $v_3$ смежна с $x, y, v$ и $v_1$ в $F$. Тогда $v_1x \notin F$. Положим $D'' = (D' \cup v_3v) \setminus v_1v$, $F'' = (F' \cup v_1v) \setminus v_3v$ и $C = (C_1 \cup vv_3v_2v_1x) \setminus vx$. Ясно, что $D'', F''$ и $C$ удовлетворяют условиям 1), 3) и 4). Покажем, что выполняется условие 2). Заметим, что вершины $v_3$ и $v$ лежат в разных компонентах связности леса $T_1 \setminus vv_1$; таким образом, $T_1 \cup v_3v \setminus v_1v$ является остовным деревом. Заметим также, что вершины $v, v_1$ и $v_3$ формируют единственный цикл в графе $T_2 \cup v_1v$; следовательно, $T_2 \cup v_1v \setminus v_3v$ также является остовным деревом. Таким образом, $D$ является $F$-обходимым, а из произвольности леса $F$ следует, что $D$ является обходимым. Лемма 3 доказана.

Для дерева $T$, леса $F$ и рёбер $e \in T$ и $f \in F$ назовём *обменом* операцию удаления этих рёбер из $T$ и $F$ и добавления их в $F$ и $T$ соответственно (т. е. полагается $T' = (T \cup f) \setminus e$ и $F' = (F \cup e) \setminus f$). Обмен назовём *корректным*, если $T'$ и $F'$ удовлетворяют условию 2). Следующее очевидное наблюдение помогает проверить корректность всех используемых ниже обменов.

**Наблюдение.** *Обмен является корректным, если ребро $f$ лежит на цикле в графе $F \cup e$, а его концевые вершины лежат в разных компонентах связности леса $T \setminus e$.*

**Утверждение 1.** *Каждое дерево с 5 вершинами обходимо.*

Доказательство. Существуют три неизоморфных дерева с 5 вершинами. Таким образом, возникают три случая. Пусть $F$ — произвольный лес с 5 вершинами.

Случай 1. Дерево $T$ является путём длины 4 (рис. 2а). Если $v_1v_5 \notin F$, то положим $C = T \cup v_1v_5$, и $T$ обходимо (для $T' = T$, $F' = F$). Если $F$ не содержит ни $v_1v_3$, ни $v_3v_5$, то можно применить лемму 2 для деревьев $D_1$и $D_2$, индуцированных множествами $\{v_1, v_2, v_3\}$ и $\{v_3, v_4, v_5\}$ соответственно. Таким образом, можно считать, что $v_1v_3, v_1v_5 \in F$, но $v_3v_5 \notin F$. Если $v_1v_4 \notin F$, то можно в качестве $C$ взять цикл $v_1v_2v_3v_5v_4$, откуда следует, что $T$ обходимо. Наконец, если $v_1v_4 \in F$, то положим $T' = (T \cup v_1v_4) \setminus v_3v_4$, $F' = (F \cup v_3v_4) \setminus v_1v_4$ и $C = v_1v_2v_3v_5v_4$. Очевидно, $T', F'$ и $C$ удовлетворяют условиям 1)–4).

Случай 2. Дерево $T$ такое, как на рис. 2б. Если $v_4v_5 \in F$, то либо $(T_2^* \cup v_3v_4) \setminus v_4v_5$, либо $(T_2^* \cup v_3v_5) \setminus v_4v_5$ является остовным деревом. Так что можно сделать обмен ребра $v_4v_5$ на одно из рёбер $v_3v_4$ или $v_4v_5$,

сводя задачу к предыдущему случаю. Предположим, что $v_4v_5 \notin F$. Если $v_1v_4 \notin F$, то полагаем $C = v_1v_2v_3v_5v_4$; если $v_1v_3 \notin F$, то поддеревья $T \setminus \{v_1, v_2\}$ и $T \setminus \{v_4, v_5\}$ оба являются $F$-обходимыми, и по лемме 2 $T$ обходимо. Таким образом, можно считать, что $v_1v_3, v_1v_4 \in F$. Тогда произведём обмен $v_1v_4$ на $v_3v_4$ и сведём задачу к случаю 1.

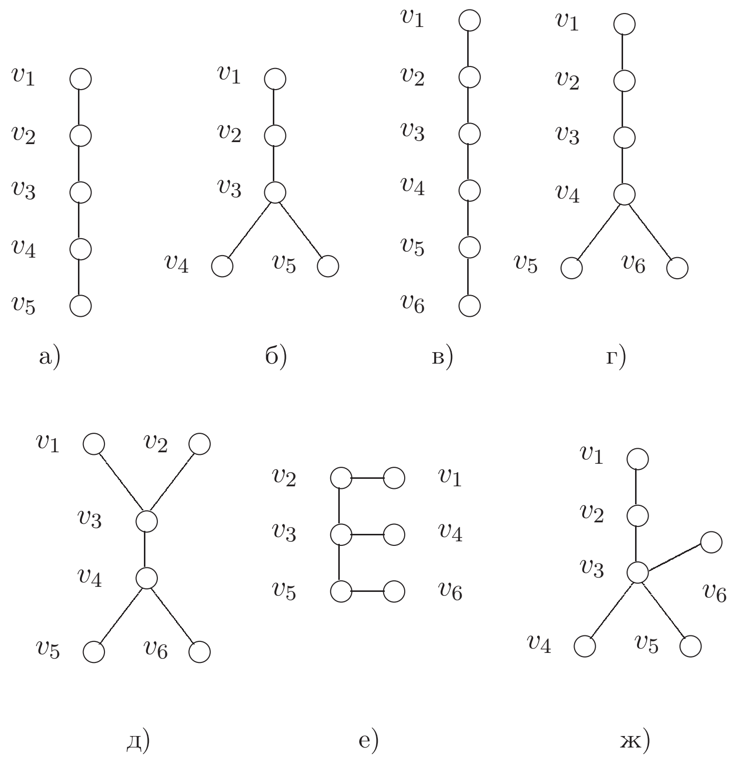


Рис. 2. Деревья с 5 и 6 вершинами

Случай 3. Дерево $T$ является звездой $K_{1,4}$. Если ни одно из рёбер, соединяющих листья звезды, не принадлежит $F$, то $T$ очевидно обходим (в качестве $C$ можно взять любой цикл). Если же какое-то из таких рёбер $e$ лежит в $F$, то его можно обменять на одно из рёбер, соединяющих конец ребра $e$ с центром звезды, сводя задачу к случаю 2. Утверждение 1 доказано.

**Утверждение 2.** *Каждое дерево с* 6 *вершинами обходимо.*

Доказательство. Имеются шесть неизоморфных деревьев с 6 вершинами. Таким образом, возникают шесть случаев. Пусть $F$ — произвольный лес с 6 вершинами.

Случай 1. Дерево $T$ является путём длины 5 (рис. 2в). Если $v_1v_6 \notin F$, то $T, F$ и $C = T \cup v_1v_6$ удовлетворяют условиям 1)–4). Если $F$ не содержит $v_1v_3$, то поддерево $T \setminus \{v_4, v_5, v_6\}$ очевидно $F$-обходимо, и по лемме 3 $T$ является обходимым. Значит, $v_1v_3 \in F$ и, аналогично, $v_4v_6 \in F$. Но тогда $v_1v_4, v_3v_6 \notin F$ и можно положить $C = v_1v_2v_3v_6v_5v_4$.

Случай 2. Дерево $T$ такое, как на рис. 2г. Если $v_5v_6 \in F$, то, как и в утверждении 1, можно обменять ребро $v_5v_6$ на одно из рёбер $v_4v_5$ или $v_4v_6$ и свести задачу к предыдущему случаю. Если $v_5v_6 \notin F$, то поддерево $T \setminus \{v_1, v_2, v_3\}$ является $F$-обходимым, и по лемме 3 $T$ будет обходимо.

Случай 3. Дерево $T$ такое, как на рис. 2д. По крайней мере одно из рёбер, соединяющих $\{v_1, v_2\}$ с $\{v_5, v_6\}$, должно отсутствовать в $F$. Можно предположить, что $v_1v_5 \notin F$. Если либо $v_1v_2 \in F$, либо $v_5v_6 \in F$, то задачу можно свести к случаю 2 точно так же, как случай 2 сводился к случаю 1. В противном случае полагаем $C = v_1v_2v_3v_4v_6v_5$.

Случай 4. Дерево $T$ такое, как на рис. 2е. Рассмотрим два подслучая.

а) Либо $v_1v_4 \notin F$, либо $v_4v_6 \notin F$ (предположим, что имеет место второй вариант). Если $v_1v_3 \notin F$, то оба дерева $T \setminus \{v_1, v_2\}$ и $T \setminus \{v_4, v_5, v_6\}$ являются $F$-обходимыми, и по лемме 2 $T$ также обходимо. Если $v_1v_5 \notin F$, то $T, F$ и $C = v_1v_2v_3v_4v_6v_5$ удовлетворяют условиям 1)–4). Наконец, если $v_1v_3, v_1v_5 \in F$, то можно обменять $v_3v_5$ на $v_1v_5$, приходя к случаю 1.

б) Оба ребра $v_1v_4$ и $v_4v_6$ лежат в $F$. Очевидно, $v_1v_6 \notin F$. Если $v_2v_4 \notin F$ ($v_5v_4 \notin F$), то $T, F$ и $C = v_1v_2v_4v_3v_5v_6$ ($C = v_1v_2v_3v_4v_5v_6$) удовлетворяют условиям 1)–4). Если $v_2v_4, v_5v_4 \in F$, то $v_2v_5 \notin F$. Обменяем $v_5v_6$ на $v_4v_6$ и положим $C = v_1v_2v_5v_3v_4v_6$.

Случай 5. Дерево $T$ такое, как на рис. 2ж. Если множество $\{v_4, v_5, v_6\}$ содержит хотя бы одно ребро из $F$, то эту задачу можно свести к предыдущему случаю точно так же, как случай 2 сводился к случаю 1. В противном случае деревья $T \setminus \{v_1, v_2\}$ и $T \setminus \{v_1, v_2, v_4\}$ являются $F$-обходимыми. Тогда по лемме 2 $v_1v_3 \in F$ и $v_1v_4 \in F$. Теперь можно обменять $v_3v_4$ на $v_1v_4$ и прийти к случаю 2.

Случай 6. Если $T$ является звездой $K_{1,5}$, то задача сводится к случаю 5 в точности тем же способом, который был использован в случае 3 утверждения 1. Утверждение 2 доказано.

Рассмотрим вершину $v$ дерева $T$. Обозначим через $A_1, A_2, \ldots, A_k$ компоненты связности леса $T \setminus v$ и положим $a_i = |A_i|, i = 1, 2, \ldots, k$. Можно считать, что $a_1 \geqslant a_2 \geqslant \ldots \geqslant a_k$. Вершина $v$ называется *центром* дерева $T$, если для неё $a_1$ достигает минимума по всем вершинам дерева $T$. Нетрудно заметить, что если $v$ является центром, то выполняется нера-

венство

$$a_1 \leqslant a_2 + a_3 + \ldots + a_k + 1. \qquad (1)$$

В частности, $a_1 \leqslant \lfloor n/2 \rfloor$.

Говорим, что вершина $v \in V(T)$ является *почкой*, если с ней смежно хотя бы два листа.

Доказательство теоремы 1. Предположим, что утверждение теоремы неверно. Тогда рассмотрим все контрпримеры к ней с минимальным числом вершин $n$. Среди них выберем дерево с наименьшим числом листьев, смежных с почками. Обозначим его через $T$. Из утверждений 1 и 2 вытекает, что $n \geqslant 7$. Пусть $F$ — произвольный лес, непересекающийся с $T$.

**Свойство 1.** *В $T$ нет почек.*

Действительно, если $u$ является почкой, смежной с листьями $v$ и $w$, то дерево $T \setminus \{v, w\}$ содержит по крайней мере 5 вершин. Из минимальности $T$ следует, что дерево $T \setminus \{v, w\}$ обходимо. Если $vw \notin F$, то дерево, индуцированное вершинами $u, v$ и $w$, является $F$-обходимым, и по лемме 2 $T$ также обходимо; противоречие. В противном случае можно обменять ребро $vw$ на $uv$ или $uw$, получив дерево $T'$ с меньшим числом листьев, смежных с почками. По выбору $T$ дерево $T'$ является обходимым, а значит, и $T$ обходимо. Свойство 1 доказано.

Пусть $v$ — центр дерева $T$. По свойству 1 $a_{k-1} \geqslant 2$.

**Свойство 2.** *Не существует такого $I \subset \{1, 2, \ldots, k\}$, что $\sum\limits_{i \in I} a_i \geqslant 4$ и $\sum\limits_{i \notin I} a_i \geqslant 4$.*

Действительно, в противном случае деревья, индуцированные $\bigcup\limits_{i \in I} A_i \cup \{v\}$ и $\bigcup\limits_{i \notin I} A_i \cup \{v\}$, имеют не менее 5 вершин каждое. Из минимальности $T$ оба они обходимы. Тогда по лемме 2 дерево $T$ тоже обходимо. Свойство 2 доказано.

В частности, если $a_1 = 4$, то $a_2 + a_3 + \ldots + a_k \leqslant 3$. Отметим, что из (1) и свойства 2 вытекает, что $a_1 \leqslant 4$.

**Свойство 3.** Если $a_i = 3$ для некоторого $i$, то $n = 7$.

Действительно, по свойству 1 дерево, индуцированное множеством $A_i \cup \{v\}$, должно быть путём длины 3. Если $n > 7$, то $|T \setminus A_i| \geqslant 5$ и из минимальности $T$ вытекает, что дерево $T \setminus A_i$ обходимо. Тогда по лемме 3 $T$ также обходимо; противоречие. Свойство 3 доказано.

Лишь следующие особые деревья удовлетворяют свойствам 1–3.

1. Два дерева с параметрами $a_1 = 4,\ a_2 = 2,\ a_3 = 1$, приведённые на рис. 3г и рис. 3д.

2. Дерево с параметрами $a_1 = 3,\ a_2 = 3$, изображённое на рис. 3а.

3. Дерево с параметрами $a_1 = 3,\ a_2 = 2,\ a_3 = 1$, изображённое на рис. 3б.

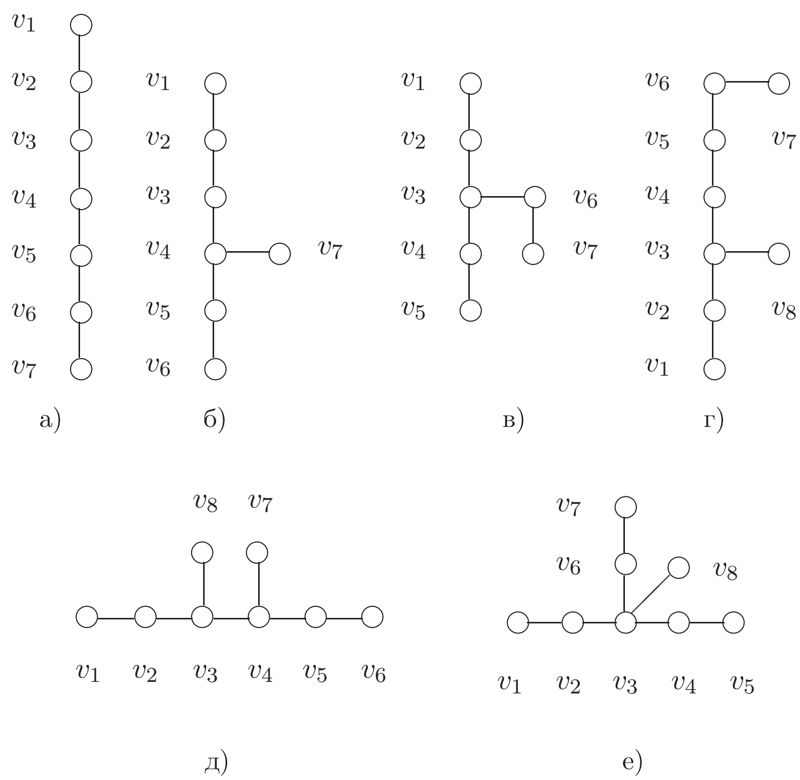


Рис. 3. Особые деревья

4. Дерево с параметрами $a_1 = 2,\ a_2 = 2,\ a_3 = 2$, изображённое на рис. 3в.

5. Дерево с параметрами $a_1 = 2,\ a_2 = 2,\ a_3 = 2,\ a_4 = 1$, изображённое на рис. 3е.

Рассмотрим каждое из этих особых деревьев отдельно (номер каждого случая совпадает с номером дерева на рис. 3).

а) При $T = P_6$ по лемме 3 $v_1v_4, v_4v_7 \in F$. Однако тогда $v_1v_7 \notin F$ и можно положить $C = v_1v_2v_3v_4v_5v_6v_7$.

б) Имеем $v_6v_7 \in F$ по лемме 3 и $v_4v_6 \in F$ по лемме 2 и выбору $T$. Тогда обмениваем $v_4v_7$ на $v_6v_7$, сводя задачу к предыдущему случаю.

в) По лемме 2 и выбору $T$ имеем $v_1v_3, v_3v_5, v_3v_7 \in F$. Следовательно, $v_1v_7, v_5v_7 \notin F$ и по крайней мере одно из рёбер $v_1v_6, v_5v_6$ не лежит в $F$ (скажем, $v_5v_6 \notin F$). Тогда $T, F$ и $C = v_1v_2v_3v_4v_5v_6v_7$ удовлетворяют условиям 1)–4).

г) Из минимальности дерева $T$ вытекает, что поддерево $T \setminus \{v_5, v_6, v_7\}$ обходимо. Тогда по лемме 3 $T$ также обходимо.

д) Из выбора $T$ следует, что деревья $T \setminus \{v_5, v_6\}$ и $T \setminus \{v_5, v_6, v_7\}$ обходимы; следовательно, по лемме 2 имеем $v_4v_6, v_6v_7 \in F$. Тогда обменяем $v_4v_7$ на $v_6v_7$ и сведём задачу к предыдущему случаю.

е) Как и в случае в), имеем $v_1v_3, v_3v_5 \in F$. Но тогда $v_1v_8 \notin F$ или $v_5v_8 \notin F$. В любом случае $T$ обходимо по лемме 2.

Теорема 1 доказана.

Нетрудно видеть, что доказательство теоремы 1 фактически содержит алгоритм построения цикла $C_1$. Формальное описание рекурсивного алгортитма `Phase_1` приведено ниже.

Алгоритм `Phase_1`.

Положим $T_1 := T_1^*$ и $T_2 := T_2^*$.

Шаг 1. Если $T_1$ содержит ровно 5 или 6 вершин или $T_1$ является одним из особых деревьев, изображённых на рис. 3, то цикл $C_1$ находится непосредственно так, как описано в доказательстве утверждений 1, 2 и финальной части доказательства теоремы 1.

В противном случае $T_1$ не удовлетворяет условиям хотя бы одного из свойств 1–3 из доказательства теоремы 1.

Шаг 2. Допустим, что $T_1$ имеет почку $x$ с листьями $u, v$ (т. е. $T_1$ не удовлетворяет условиям свойства 1).

Если $uv \in T_2$, то преобразуем $T_1$ и $T_2$ с помощью обмена, приведённого в доказательстве свойства 1, в деревья $T_1'$ и $T_2'$, где $T_1'$ имеет меньшее чем $T_1$ число листьев, смежных с почками. Положим $T_1 := T_1'$, $T_2 := T_2'$ и перейдём на шаг 1.

Если $u, v \notin T_2$, то применим алгоритм `Phase_1` к поддереву $T_1 \setminus \{u, v\}$, найдём для него цикл $C'$ и построим цикл $C_1$ для $T_1$ из циклов $C'$ и $C'' = xuv$, используя процедуру, описанную в доказательстве леммы 2.

Шаг 3. Найдём в дереве $T_1$ центр $v$ (на самом деле достаточно найти любую вершину, удовлетворяющую (1)).

Шаг 4. Если $n > 7$ и $a_i = 3$ для некоторого $i$ (т. е. $T_1$ не удовлетворяет условиям свойства 3), то применим `Phase_1` к поддереву $T_1 \setminus A_i$ и найдём для него цикл $C'$. Используя процедуру, описанную в лемме 3, построим искомый цикл $C_1$ для $T_1$.

Шаг 5. Найдём такое множество $S \subset \{1, 2, \ldots, k\}$, что $\sum\limits_{i \in S} a_i \geqslant 4$ и $\sum\limits_{i \notin S} a_i \geqslant 4$ (оно существует, так как $T_1$ не удовлетворяет условиям свойства 2). Если $n = 7$, то $S$ можно найти полным перебором. В противном случае либо $a_1 \geqslant 4$ и тогда $S = \{1\}$, либо $a_1 = 2$ и $S = \{1, 2\}$.

Применяя алгоритм `Phase_1` к поддеревьям, индуцированным множествами $\bigcup\limits_{i \in I} A_i \cup \{v\}$ и $\bigcup\limits_{i \notin I} A_i \cup \{v\}$, найдём для них циклы $C'$ и $C''$ и построим их этих циклов искомый цикл $C_1$ для $T_1$, используя процедуру, описанную в лемме 2.

Ясно, что время выполнения фазы 1 ограничено сверху временем выполнения фазы 0.

## 3. Выполнение фазы 2

После выполнения первой фазы имеем гамильтонов цикл $C_1$ и остовное дерево $T_2$, не пересекающееся с $C_1$. Необходимо найти такой цикл $C_2$, не пересекающийся с $C_1$, что граф $C_2 \cup T_2$ является внешнепланарным с внешней гранью $C_2$. Мы используем похожую идею, что и в предыдущем разделе (разбиение $T_2$ на поддеревья), но здесь мы, очевидно, не можем обмениваться рёбрами между $T_2$ и $C_1$. С другой стороны, мы можем использовать тот факт, что запрещённый граф $C_1$ имеет максимальную степень 2.

Под *частичным туром* в данном разделе будем понимать либо гамильтонов цикл, либо остовный граф, каждая компонента связности которого является путём. Пусть $T$ — остовное дерево. Говорим, что $T$ является *проходимым*, если для любого частичного тура $F$, не пересекающегося с $T$, существует такой гамильтонов цикл $C$, что:

1) $F \cap C = \varnothing$;

2) граф $T \cup C$ является внешнепланарным с внешней гранью $C$.

Назовём вершину дерева $T$ *вершиной ветвления*, если все кроме одного из её соседей являются листьями. Поддерево, индуцированное вершиной ветвления и всеми смежными с ней листьями, назовём *ветвью*.

**Лемма 4.** *Пусть $T$ — остовное дерево, а $B$ — его ветвь. Если дерево $T' = T \setminus B$ проходимо, то и $T$ проходимо.*

Доказательство. Пусть $v$ — вершина ветвления в $B$, а $v_1, v_2, \ldots, v_t$ — смежные с ней листья. Рассмотрим произвольный частичный тур $F$. Пусть $C'$ — гамильтонов цикл для $T'$ и $F'$, где через $F'$ обозначено ограничение $F$ на множество вершин дерева $T'$. Пусть $u$ — сосед $v$ в $T$, не являющийся листом. Через $w_1, w_2$ обозначим соседей вершины $u$ в цикле

$C'$ (рис. 4). По определению частичного тура, если $t \geqslant 2$, то по крайней мере одно из рёбер $v_1w_1, v_1w_2, v_tw_1, v_tw_2$ не лежит в $F$. Можно считать, что $v_tw_1 \notin F$. Рассмотрим четыре случая.

1. Если $t \geqslant 4$, то можно так переупорядочить вершины $B \setminus v$, что $v_iv_{i+1} \notin F$ для всех $i = 1, 2, \ldots, t-1$. Тогда ребро $uw_1$ удалим из цикла $C'$, а вместо него вставим путь $uvv_1v_2 \ldots v_tw_1$. Полученный цикл $C$ удовлетворяет условиям 1)–2).

2. Если $t = 3$, то можно предположить, что $v_1v_2, v_2v_3 \in F$ (иначе поступаем так же, как и в предыдущем случае). Тогда $v_2u \notin F$ и $v_1v_3 \notin F$, так как $F$ является частичным туром. Ребро $uw_1$ удалим из $C'$, а путь $uv_2vv_1v_3w_1$ вставим на его место. Полученный цикл $C$ удовлетворяет условиям 1)–2).

3. Если $t = 2$, то можно предположить, что $v_1v_2 \in F$ (иначе поступаем так же, как и в первом случае). Если $uv_1 \notin F$, то ребро $uw_1$ в цикле $C'$ заменим путём $uv_1vv_2w_1$. В противном случае $uv_2, v_1w_1 \notin F$, так как $F$ — частичный тур. Тогда ребро $uw_1$ в цикле $C'$ заменим путём $uv_2vv_1w_1$ и снова получим цикл $C$, удовлетворяющий условиям 1)–2).

4. Если $t = 1$ и $v_1w_i \notin F$ для некоторого $i = 1, 2$ то вставим путь $uvv_1w_i$ вместо ребра $uw_i$ в цикл $C'$. В противном случае имеем $uv_1 \notin F$ и $vw_i \notin F$ для некоторого $i = 1, 2$, и ребро $uw_i$ в цикле $C'$ заменяется на путь $uv_1vw_i$. Лемма 4 доказана.

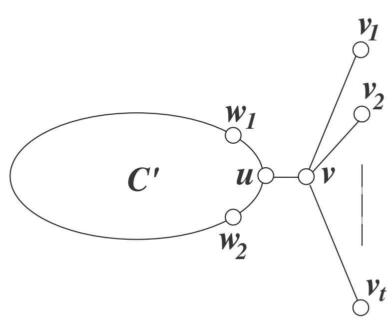


Рис. 4. Шаг индукции в лемме 4

Теперь мы готовы доказать основную теорему настоящего раздела.

**Теорема 2.** *Каждое дерево с не менее чем* 5 *вершинами является проходимым.*

Доказательство. Допустим, что утверждение теоремы неверно, и рассмотрим контрпример $T$ с наименьшим числом вершин $n \geqslant 5$. Пусть $F$ — произвольный частичный тур. Если $T$ является звездой $K_{1,n-1}$, то

поскольку $n-1 \geqslant 4$, её листья можно переупорядочить так, что $v_i v_{i+1} \notin F$ для всех $i = 1, 2, \dots, n-2$. Тогда можно взять $C = vv_1v_2 \dots v_{n-1}$ в качестве искомого цикла ($v$ является центром звезды). Если $T$ не является звездой, то в нём есть вершины ветвления. Если в $T$ есть такая ветвь $B$, что $|T \setminus B| \geqslant 5$, то из минимальности $T$ и леммы 4 вытекает, что $T$ проходимо. Таким образом, для каждой ветви $B$ дерево $T \setminus B$ содержит не более 4 вершин. Непосредственной проверкой нетрудно убедиться, что существует ровно 10 таких деревьев (рис. 5). Рассмотрим каждое из этих деревьев отдельно (номер каждого случая совпадает с номером дерева на рис. 5).

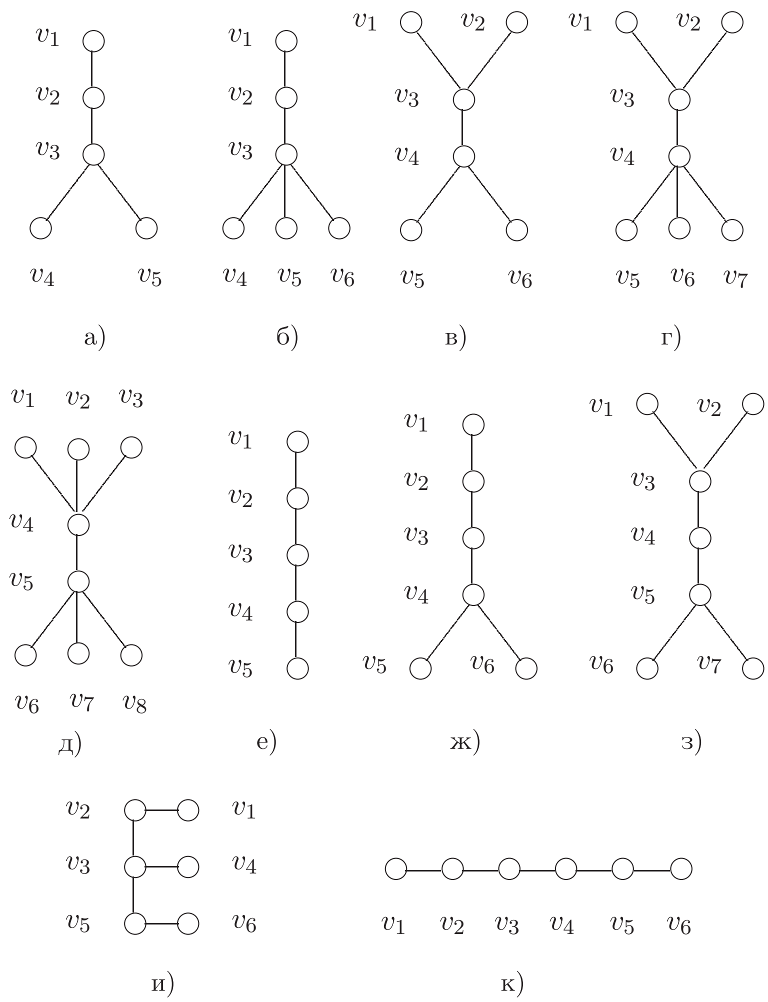


Рис. 5. Деревья без хорошей ветви

а) Можно считать, что $v_4v_5 \in F$ (в противном случае можно добавить ветвь $B = T(\{v_1, v_2\})$ к циклу $C' = v_3v_4v_5$ так же, как в лемме 4). Если $v_1v_5 \in F$, то $v_1v_4, v_2v_5 \notin F$ по определению частичного тура, и можно положить $C = v_1v_4v_3v_5v_2$. Аналогично, $v_1v_4 \notin F$. Если $v_2v_4 \in F$, то $v_2v_5 \notin F$, и можно положить $C = v_2v_5v_3v_4v_1$. В противном случае положим $C = v_1v_2v_4v_3v_5$.

б) Как и в предыдущем случае, можно предположить, что $v_4v_5, v_5v_6 \in F$. Тогда $v_2v_5, v_4v_6 \notin F$ и вершина $v_1$ несмежна в $F$ либо с $v_4$, либо с $v_6$. Тогда положим $C = v_1v_2v_5v_3v_6v_4$ или $C = v_1v_2v_5v_3v_4v_6$ соответственно.

в) Можно считать, что $v_1v_2, v_5v_6 \in F$ (в противном случае можно применить процедуру из леммы 4). Тогда в $F$ может быть не более одного ребра, соединяющего множества $\{v_1, v_2\}$ и $\{v_5, v_6\}$. Таким образом, можно предположить, что $v_1v_5, v_2v_6 \notin F$, и положить $C = v_1v_3v_2v_6v_4v_5$.

г) Можно считать, что $v_1v_2, v_5v_6, v_6v_7 \in F$ и $v_1v_5, v_2v_6, v_5v_7 \notin F$. Тогда цикл $C = v_1v_3v_2v_6v_4v_7v_5$ удовлетворяет условиям 1)–2).

д) Поскольку $v_1v_2, v_2v_3, v_6v_7, v_7v_8 \in F$, можно положить

$$C = v_1v_3v_4v_2v_8v_6v_5v_7.$$

е) Если $v_3v_5 \notin F$, то ветвь $B = T \setminus \{v_3, v_4, v_5\})$ добавляется к циклу $C' = v_3v_4v_5$ так же, как в лемме 4. Значит, $v_3v_5 \in F$. Аналогично $v_1v_3 \in F$. Тогда $v_1v_5 \notin F$, и можно положить $C = v_1v_2v_3v_4v_5$.

ж) Как и в предыдущем случае, $v_1v_3 \in F$ и $F$ должен содержать два ребра из множества $\{v_3v_5, v_3v_6, v_5v_6\}$. Поскольку степень вершины $v_3$ в $F$ не превосходит 2, можно считать, что $v_3v_5, v_5v_6 \in F$. Тогда $v_1v_5, v_3v_6 \notin F$, и можно положить $C = v_1v_2v_3v_6v_4v_5$.

з) Как и в предыдущем случае, можно считать, что $v_1v_2, v_2v_4, v_4v_6$, $v_6v_7 \in F$. Тогда $v_1v_4, v_4v_7, v_2v_6 \notin F$. Поэтому цикл $C = v_1v_3v_2v_6v_5v_7v_4$ удовлетворяет условиям 1)–2).

и) Если $v_1v_4 \notin F$, то ветвь $B = T(\{v_5, v_6\})$ добавляется к циклу $C' = v_1v_2v_3v_4$ так же, как в лемме 4. Значит, $v_1v_4 \in F$. Аналогично $v_4v_6 \in F$. Тогда $v_1v_6, v_4v_5 \notin F$, и можно положить $C = v_1v_2v_3v_4v_5v_6$.

к) Как и в случае е), $v_1v_4, v_3v_6 \in F$. Если $v_1v_6 \notin F$, то $C = v_1v_2v_3v_4v_5v_6$. Иначе, $v_1v_5, v_4v_6 \notin F$, поэтому цикл $C = v_1v_2v_3v_4v_6v_5$ удовлетворяет условиям 1)–2). Теорема 2 доказана.

Доказательство теоремы 2 легко превратить в алгоритм для поиска цикла $C_2$.

Алгоритм `Phase_2`

Положим $T_2 := T_2^*$.

Шаг 1. Если $T_2$ изоморфно звезде $K_{1,n-1}$ или одному из деревьев, показанных на рис. 3, то искомый цикл $C_2$ строится непосредственно, как описано в доказательстве теоремы 2.

Шаг 2. Если в $T_2$ есть такая ветвь $B$, что $|T_2 \setminus B| \geqslant 5$, то, применяя алгоритм `Phase_1` к дереву $T_2' = T_2 \setminus B$, найдём для него цикл $C'$. Используя процедуру из доказательства леммы 4, построим $C_2$ из $C'$.

И снова очевидно, что время работы алгоритма `Phase_2` ограничено сверху временем работы фазы 0.

Таким образом, описание алгоритма `Disj_Ham_Cycles` полностью завершено, и мы доказали следующую теорему.

**Теорема 3.** *Алгоритм* `Disj_Ham_Cycles` *за время* $O(n^2 \log n)$ *находит допустимое решение метрической задачи* 2-PSP, *чей вес не более чем вдвое превосходит вес оптимального решения.*

## ЛИТЕРАТУРА


1. **Агеев А. А., Бабурин А. Е., Гимади Э. Х.** Полиномиальный алгоритм с оценкой точности 3/4 для отыскания двух непересекающихся гамильтоновых циклов максимального веса // Дискрет. анализ и исслед. операций, Сер. 1. — 2006. — Т. 13, № 2. — С. 11–20.
2. **Бабурин А. Е., Гимади Э. Х., Коркишко Н. М.** Приближённые алгоритмы для нахождения двух рёберно непересекающихся гамильтоновых циклов минимального веса // Дискрет. анализ и исслед. операций. Сер. 2. — 2004. — Т. 11, № 1. — С. 11–25.
3. **Сердюков А. И.** Некоторые экстремальные обходы в графах // Управляемые системы. — 1978. — № 17. — С. 76–79.
4. **Christofides N.** Worst-case analysis of a new heuristic for the traveling salesman problem. Technical Report CS–93–13 // Pittsburgh: Carnegie–Mellon Univ., 1976. — 10 p.
5. **De Brey M. J. D., Volgenant A.** Well-solved cases of the 2-peripatetic salesman problem // Optimization. — 1997. — Vol. 39, N 3. — P. 275–293.
6. **Duchenne E., Laporte G., Semet F.** Branch-and-cut algorithms for the undirected $m$-peripatetic salesman problem // Europ. J. Oper. Res. — 2005. — Vol. 162, N 3. — P. 700–712.
7. **De Kort J. B. J. M.** Lower bounds for symmetric $K$-peripatetic salesman problems // Optimization. — 1991. — Vol. 22, N 1. — P. 113–122.
8. **De Kort J. B. J. M.** Upper bounds for the symmetric 2-peripatetic salesman problem // Optimization. — 1992. — Vol. 23, N 4. — P. 357–367.
9. **De Kort J. B. J. M.** A branch and bound algorithm for symmetric 2-peripatetic salesman problems // Europ. J. Oper. Res. — 1993. — Vol. 70. — P. 229–243.

10. **Krarup J.** The peripatetic salesman and some related unsolved problems // Combinatorial programming: methods and applications (Proc. NATO Advanced Study Inst., Versailles, 1974). — Dordrecht: Reidel Press, 1975. — P. 173–178.
11. **Roskind J., Tarjan R. E.** A note on finding minimum-cost edge-disjoint spanning trees // Math. Oper. Res. — 1985. — Vol. 10, N 4. — P. 701–708.
12. **Wolfter C. R., Cordone R.** A heuristic approach to the overnight security service problem // Computers & Oper. Res. — 2003. — Vol. 30. — P. 1269–1287.

*Агеев Александр Александрович*,
e-mail: ageev@math.nsc.ru

*Пяткин Артём Валерьевич*,
e-mail: artem@math.nsc.ru

Статья поступила
21 февраля 2009 г.
Переработанный вариант —
12 мая 2009 г.