Санкт-Петербургский государственный университет

БЕЛОВА Татьяна Сергеевна

Выпускная квалификационная работа

Сложность приближения задач реберной модификации до графов без порожденного фиксированного подграфа

Уровень образования: магистратура
Направление: 01.04.01 «Математика»
Основная образовательная программа: ВМ.5832.2019
«Современная математика»

Научный руководитель:
доцент,
Санкт-Петербургский государственный университет,
Факультет математики и компьютерных наук,
к.ф.-м.н. Близнец И. А.

Рецензент:
профессор,
Джорджтаунский университет,
к.ф.-м.н. Головнёв А. Г.

Санкт-Петербург
2021г

Содержание

Введение

В общем виде задача модификации графов звучит следующим образом. Есть набор разрешенных модификаций графа, например, удаление вершин или ребер. На вход подается граф G. Требуется для графа G совершить минимальное количество разрешенных модификаций так, чтобы получившийся граф обладал определенными свойствами. Многие известные задачи на графах являются задачами модификации графов. Например, в задаче Vertex Cover требуется удалить минимальное количество вершин так, чтобы в графе не осталось ребер, в задаче Odd Cycle Transversal требуется удалить минимальное количество вершин (ребер) так, чтобы оставшийся граф был двудольным, а в задаче Feedback Vertex Set требуется удалить минимальное количество вершин так, чтобы в графе не осталось циклов.

Большинство естественных задач модификации графов являются NP-трудными. В работе [13] было показано, что если свойство, которым должен обладать граф после модификаций, является нетривиальным (то есть выполняется для бесконечного множества графов и не выполняется тоже для бесконечного множества графов), а также является наследуемым (то есть из выполнения этого свойства для графа G следует выполнение этого свойства для любого индуцированного подграфа G), а единственной разрешенной модификацией является удаление вершин, то такие задачи являются NP-трудными. Однако для задач модификации ребер все обстоит куда сложнее. Еще в ранних работах [15, 7], посвященных задачам модификации ребер, было замечено, что несмотря на то, мы можем получать результаты для конкретных классов задач, получить обобщение для всех задач модификации ребер может быть сложно.

Особый интерес задачи модификации графов представляют в области параметризованных алгоритмов, поскольку из практических соображений количество необходимых модификаций обычно мало по сравнению с размером графа, а тогда разумно рассматривать количество модификаций в качестве параметра. Впервые рассматривать задачи модификации графов с точки зрения параметризованных алгоритмов было предложено в работе [10].

Типичными представителями задач модификации графов являются задачи H-free Edge Deletion, H-free Edge Completion и H-free Edge Editing для всевозможных графов H. В задаче H-free Edge Deletion дан граф G и требуется удалить наименьшее количество ребер из G так, чтобы в получившемся графе не нашлось индуцированного подграфа, изоморфного графу H. В задаче H-free Edge Completion требуется добавить наименьшее количество ребер в G так, чтобы в получившемся графе не нашлось индуцированного подграфа, изоморфного графу H. В задаче H-free Edge Editing разрешено и удалять, и добавлять ребра, и требуется совершить наименьшее количество модификаций ребер G, чтобы в получившемся графе не нашлось индуцированного подграфа, изоморфного графу H.

Используя стандартный метод расщепления, в работе [4] было показано, что задачи H-free Edge Deletion (Completion, Editing) решаются за время $c^k \cdot poly(n)$, где k — ограничение на количество допустимых модификаций, n — размер графа, а c зависит только от графа H, который фиксирован для каждой задачи. При этом в общем случае не приходится надеяться на то, что задачи H-free Edge Deletion (Completion, Editing) решаются за время, существенно меньшее чем $c^k \cdot poly(n)$, поскольку в работе [2] было показано, что для всех графов H, содержащих хотя бы два ребра (антиребра), задачи H-free Edge Deletion (Completion) являются NP-трудными и не решаются за время $2^{o(k)} \cdot poly(n)$, если выполняется гипотеза ETH, а для всех графов H, содержащих хотя бы три вершины, задачи H-free Edge Editing являются NP-трудными и не решаются за время $2^{o(k)} \cdot poly(n)$, если выполняется гипотеза ETH.

Задачи H-free Edge Deletion (Completion, Editing) также рассматриваются с точки зрения сложности кернелизации, а именно существования полиномиальных ядер. Другими словами, интересен вопрос, существует ли полиномиальный по времени алгоритм, который по инстансу задачи H-free Edge Deletion (Completion, Editing) может выдать другой инстанс этой же задачи, размер которого ограничен некоторым полиномом от параметра k. В отличие от задач H-free Vertex Deletion, в которых вместо ребер из графа удаляются вершины, и для которых с помощью простого сведения к задаче $|V(H)|$-Hitting Set можно получить полиномиальное ядро ([12, 1]), ситуация с задачами H-free Edge Deletion (Completion, Editing) обстоит куда сложнее. Это связано с тем, что удаление и добавление ребер в граф может создавать в графе новые подграфы, изоморфные H. В работе [12] впервые был представлен граф H, для которого задачи H-free Edge Deletion и H-free Edge Editing не допускают полиномиального ядра, если $NP \nsubseteq coNP/poly$. Позже в работе [9] было показано, что для всех H, являющихся циклами на хотя бы четырех вершинах или путями на хотя бы семи вершинах, задачи H-free Edge Deletion не допускают полиномиального ядра при $NP \nsubseteq coNP/poly$. Этот результат был усилен в работах [5, 6]. В них удалось показать, что если H — это путь или цикл, то задачи H-free Edge Deletion (Completion, Editing) не допускают полиномиального ядра при $NP \nsubseteq coNP/poly$ тогда и только тогда, когда H содержит хотя бы четыре ребра. Также в работах [5, 6] было показано, что если H — это трехсвязный граф и $NP \nsubseteq coNP/poly$, то задачи H-free Edge Deletion и H-free Edge Editing не допускают полиномиального ядра, тогда и только тогда, когда H не является полным графом, а задача H-free Edge Completion не допускает полиномиального ядра тогда и только тогда, когда в H отсутствует хотя бы два ребра. В работе [14] была предпринята попытка классифицировать все графы H на хотя бы пяти вершинах в зависимости от существования для соответствующих им задач полиномиального ядра. Была сформулирована гипотеза о том, что если $NP \nsubseteq coNP/poly$, то для графа H на хотя бы пяти вершинах задача H-free Edge Editing допускает полиномиальное ядро тогда и только тогда, когда H является пустым или полным графом, а задача H-free Edge Deletion допускает полиномиальное ядро тогда и только тогда, когда H либо является полным графом, либо содержит не более одного ребра. Несмотря на то, что сами гипотезы доказаны не были, в работе [14] показали, что для доказательства этих гипотез достаточно показать отсутствие полиномиального ядра только для конечного множества графов.

Еще одним направлением исследования задач H-free Edge Deletion (Completion, Editing) является изучение вопроса сложности приближения этих задач. Как было замечено ранее, удаление и добавление ребер в граф может вызывать появление новых индуцированных подграфов, изоморфных H. По этой причине искать точное или приближенное решение жадным способом становится маловероятным. Также существует связь между существованием полиномиального ядра и $poly(OPT)$-приближения для задачи, поскольку часто можно построить $poly(OPT)$-приближение, имея полиномиальное ядро, и наоборот. Несмотря на то, что эта связь не является формальной, и существуют задачи [8], которые ведут себя иначе, вполне вероятно, что в задачах H-free Edge Deletion (Completion, Editing) эта связь присутствует и результаты, полученные для сложности кернелизации, будут полезны и для случая сложности приближения. Так, в работе [3] было показано, что для H, являющихся трехсвязными графами с хотя бы двумя антиребрами, а также для H, являющихся циклами на хотя бы четырех вершинах или путями на хотя бы пяти вершинах, задачи H-free Edge Deletion и H-free Edge Completion не допускают $poly(OPT)$-приближения, если $P \neq NP$. Этот результат повторяет результат для сложности кернелизации, полученный в работах [5, 6]. Однако при этом, в отличие от случая кернелизации, в случае с $poly(OPT)$-приближением наличие в H хотя бы двух антиребер оказывается существенным для задачи H-free Edge Deletion. В той же работе [3] было показано, что задачи $(K_n - e)$-free Edge Deletion являются Min Horn Deletion-полными при всех $n \geq 5$. Min Horn Deletion-полные задачи представляют собой отдельный класс задач, про сложность приближения которых пока известно достаточно мало. В работе [11] было показано, что такие задачи лежат в классе poly-APX, но не допускают $2^{\log^{1-\varepsilon} n}$-приближения ни для какого $\varepsilon > 0$, если $P \neq NP$. Более подробно о Min Horn Deletion-полных задачах мы поговорим в разделе 3, посвященном этим задачам.

Результаты, представленные в работе. В этой работе мы попробуем построить полную классификацию графов на хотя бы пяти вершинах в зависимости от сложности приближения соответствующих им задач H-free Edge Deletion (Completion, Editing). Сформулируем гипотезы о том, как выглядит интересующая нас классификация для каждой из задач H-free Edge Deletion, H-free Edge Completion и H-free Edge Editing.

Гипотеза 1. *Пусть в графе H хотя бы пять вершин. Тогда:*
(i) $H = \overline{K_n}$ и $H = \overline{K_n - e}$: для H-free Edge Deletion есть полиномиальное решение.
(ii) $H = K_n$: для H-free Edge Deletion есть константное приближение.
(iii) $H = K_n - e$: H-free Edge Deletion является Min Horn Deletion-полной.
(iv) Остальные H: $P \neq NP \Rightarrow$ H-free Edge Deletion не допускает $poly(OPT)$-приближения.

Гипотеза 2. *Пусть в графе H хотя бы пять вершин. Тогда:*
(i) $H = K_n$ и $H = K_n - e$: для H-free Edge Completion есть полиномиальное решение.
(ii) $H = \overline{K_n}$: для H-free Edge Completion есть константное приближение.
(iii) $H = \overline{K_n - e}$: H-free Edge Completion является Min Horn Deletion-полной.
(iv) Остальные H: $P \neq NP \Rightarrow$ H-free Edge Completion не допускает $poly(OPT)$-приближения.

Гипотеза 3. *Пусть в графе H хотя бы пять вершин. Тогда:*
(i) $H = K_n$ и $H = \overline{K_n}$: для H-free Edge Editing есть константное приближение.
(ii) $H = K_n - e$ и $H = \overline{K_n - e}$: H-free Edge Editing является Min Horn Deletion-полной.
(iii) Остальные H: $P \neq NP \Rightarrow$ H-free Edge Editing не допускает $poly(OPT)$-приближения.

По сути данные гипотезы повторяют гипотезы для случая кернелизации за исключением того, что имеют дополнительный пункт о Min Horn Deletion-полных задачах. Таким образом, с одной стороны, мы верим, что сложность приближения и сложность кернелизации для задач H-free Edge Deletion (Completion, Editing) ведут себя очень похоже, а с другой, в случае со сложностью приближения мы наблюдаем немного более интересную картину.

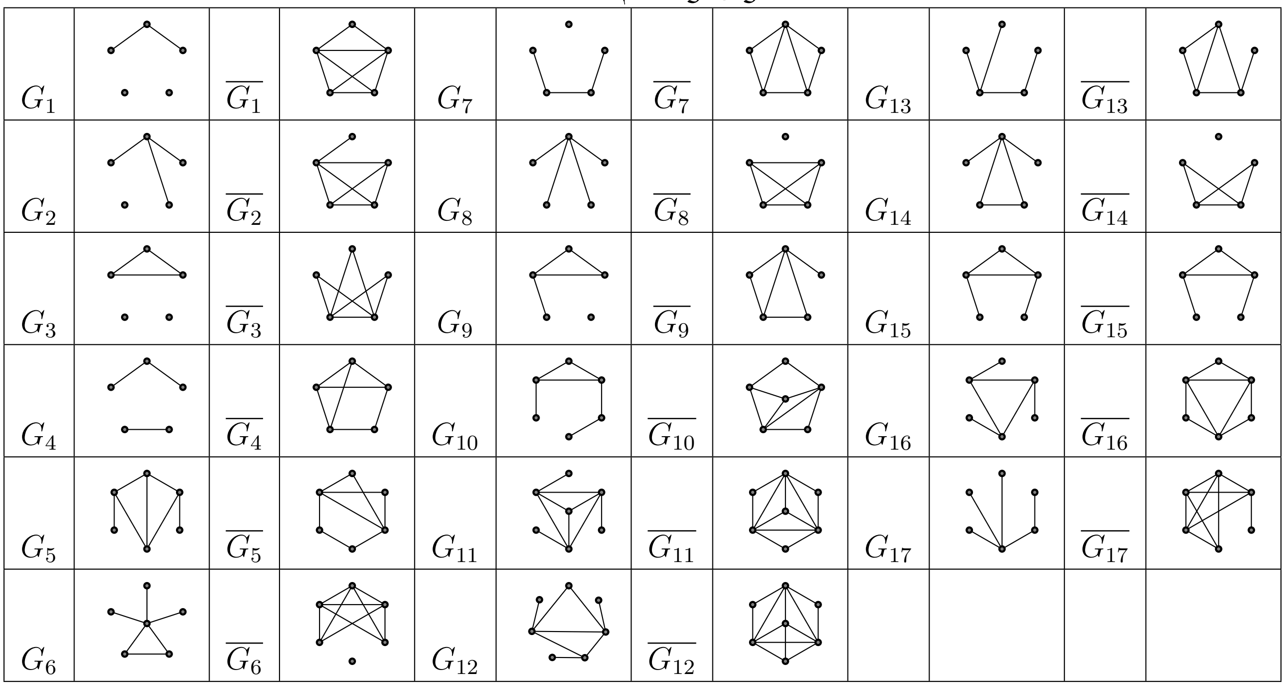
Для каждой из гипотез 1, 2, 3 мы докажем все пункты, кроме последнего, а для последнего научимся ограничивать множество рассматриваемых графов конечным множеством, а именно получим следующий результат.

Теорема 1. *Существует множество $\mathcal{G}$ из семнадцати графов такое, что:*
(i) $\forall H \in \mathcal{G} \cup \overline{\mathcal{G}}$ задача H-free Edge Deletion (Completion) не допускает $poly(OPT)$-приближения $\Leftrightarrow$ Гипотеза для H-free Edge Deletion (Completion) верна;
(ii) $\forall H \in \mathcal{G}$ задача H-free Edge Editing не допускает $poly(OPT)$-приближения $\Leftrightarrow$ Гипотеза для H-free Edge Editing верна.

Таким образом, для доказательства гипотез будет достаточно доказывать отсутствие $poly(OPT)$-приближения только для конечного набора графов. Если же какая-то из гипотез неверна, то контрпример к ней достаточно искать среди графов из этого набора. Множество графов $\mathcal{G} \cup \overline{\mathcal{G}}$ приведено в таблице 1.

Данная работа целиком посвящена доказательству теоремы 1. Повествование будет устроено следующим образом. В разделе 1 мы введем основные определения и обозначения, а также базовые леммы, которыми мы будем активно пользоваться в последующих разделах. В

Таблица 1: $\mathcal{G} \cup \overline{\mathcal{G}}$



разделе 2 мы рассмотрим графы H, для которых задачи H-free Edge Deletion (Completion, Editing) решаются за полиномиальное время или же допускают константное приближение, и тем самым докажем первые два пункта гипотез 1 и 2 и первый пункт гипотезы 3. В разделе 3 мы опишем понятие Min Horn Deletion-полных задач и докажем пункт (iii) гипотез 1 и 2, а также пункт (ii) гипотезы 3. В разделе 4 мы опишем некоторые классы графов H, для которых задачи H-free Edge Deletion (Completion, Editing) не допускают $poly(OPT)$-приближения при $P \neq NP$. В разделе 5 мы покажем, как ограничить множество таких графов H, что для доказательства гипотез 1, 2, 3 достаточно показать отсутствие $poly(OPT)$-приближения для задач H-free Edge Deletion (Completion, Editing) при $P \neq NP$, множеством графов определенной структуры. После чего, в разделах 6 и 7, мы покажем, как ограничить это множество графов конечным множеством. В разделе 8 мы еще сильнее уменьшим это множество и тем самым докажем теорему 1.

1 Определения и обозначения

Пусть G — граф. $V(G)$ обозначает множество вершин G, $E(G)$ — множество ребер G, $\overline{E}(G)$ — множество антиребер G, то есть множество пар вершин, не соединенных ребром. Мы рассматриваем только простые графы, то есть графы без петель и кратных ребер. $\overline{G}$ — дополнение графа G, то есть такой граф, что $V(\overline{G}) = V(G)$, $E(\overline{G}) = \overline{E}(G)$. Для фиксированного графа G введем следующие обозначения. ℓ — минимальная степень в G, h — максимальная степень в G, V_ℓ — множество вершин G минимальной степени, V_h — множество вершин G максимальной степени, $V_m = V(G) \setminus V_\ell \setminus V_h$. $N(v)$, или *окрестность* v, — это множество соседей вершины v в G, а $N[v]$, или *замкнутая окрестность* v, — это $N(v) \cup \{v\}$. $G[V']$ — это граф, индуцированный множеством вершин $V' \subseteq V(G)$ в G. Для $v \in V(G)$ $G - v$ обозначает граф, полученный из G удалением вершины v и инцидентных ей ребер. Для $V' \subseteq V(G)$ $G - V'$ обозначает граф, полученный из G удалением всех вершин из V' и инцидентных им ребер. Для $e \in E(G)$ $G - e$ обозначает граф, полученный из G удалением ребра e. Для $E' \subseteq E(G)$ $G - E'$ обозначает граф, полученный из G удалением всех ребер из E'. Для $E' \subseteq \overline{E}(G)$ $G + E'$ обозначает граф, полученный из G добавлением всех ребер из E'. Пусть F — множество пар вершин из G, тогда $G \triangle F$ обозначает такой граф, что $V(G \triangle F) = V(G)$, $E(G \triangle F) = \{(u,v) \mid ((u,v) \in E(G) \wedge (u,v) \notin F) \vee ((u,v) \notin E(G) \wedge (u,v) \in F)\}$. В частности, если $F \subseteq E(G)$, то $G \triangle F = G - F$, а если $F \subseteq \overline{E}(G)$, то $G \triangle F = G + F$. Если граф G' такой, что $V(G') \subseteq V(G)$, а F — множество пар вершин из $V(G)$, то $F \cap G' = F \cap (E(G') \cup \overline{E}(G'))$.

Граф G — *k-связный*, если он содержит хотя бы k вершин и $G - V'$ — связный для любого $V' \subseteq V(G)$ такого, что $|V'| \leq k - 1$. Будем называть множество $S \subseteq V(G)$ k-разделителем для графа G, если $|S| = k$ и $G - S$ несвязный. K_n — полный граф на n вершинах, $K_{n,m}$ — полный двудольный граф с долями размера n и m, C_n — цикл на n вершинах, P_n — путь на n вершинах. $G_1 \cup G_2$ — это такой граф, что $V(G_1 \cup G_2) = V(G_1) \cup V(G_2)$, $E(G_1 \cup G_2) = E(G_1) \cup E(G_2)$. $G_1 \boxtimes G_2$ — это такой граф, что $V(G_1 \boxtimes G_2) = V(G_1) \cup V(G_2)$ и $E(G_1 \boxtimes G_2) = E(G_1) \cup E(G_2) \cup \{(x,y) \mid x \in V(G_1), y \in V(G_2)\}$. tG — это объединение t копий графа G. Для $t \geq 3$ J_t — это граф, полученный из $K_2 \boxtimes tK_1$ и C_4 отождествлением ребра между вершинами максимальной степени в $K_2 \boxtimes tK_1$ с ребром из C_4. Для $t \geq 3$ Q_t — это граф, полученный из $K_{2,t}$ добавлением пути длины три между вершинами максимальной степени. Для $t_1, t_2 \geq 0$ T_{t_1,t_2} — это дерево с двумя смежными вершинами u и v такими, что $|N(u) \setminus \{v\}| = t_1$, $|N(v) \setminus \{u\}| = t_2$, и $N(v) \cup N(u) \setminus \{u,v\}$ — это вершины, степени один.

Если $\mathcal{X}$ — множество графов, то $\overline{\mathcal{X}}$ — множество дополнений этих графов. По аналогии со статьей [14] определим множества графов $\mathcal{H}$, $\mathcal{A}$, $\mathcal{D}$, $\mathcal{B}$, $\mathcal{S}$, $\mathcal{F}$ и соответствующие им множества $\overline{\mathcal{H}}$, $\overline{\mathcal{A}}$, $\overline{\mathcal{D}}$, $\overline{\mathcal{B}}$, $\overline{\mathcal{S}}$ и $\overline{\mathcal{F}}$. В таблицах 2, 3, 4 и 5 изображены графы, образующие множества $\mathcal{H}$, $\overline{\mathcal{H}}$, $\mathcal{A}$, $\overline{\mathcal{A}}$, $\mathcal{D}$, $\overline{\mathcal{D}}$, $\mathcal{B}$, $\overline{\mathcal{B}}$, $\mathcal{S}$ и $\overline{\mathcal{S}}$. $\mathcal{F}$ — это объединение множеств графов $\mathcal{F}_i$, описанных в таблице 6. Обозначим $\mathcal{W} = \mathcal{H} \cup \overline{\mathcal{H}} \cup \mathcal{A} \cup \overline{\mathcal{A}} \cup \mathcal{D} \cup \overline{\mathcal{D}} \cup \mathcal{B} \cup \overline{\mathcal{B}} \cup \mathcal{S} \cup \overline{\mathcal{S}} \cup \mathcal{F} \cup \overline{\mathcal{F}}$. Заметим, что $\overline{\mathcal{W}} = \mathcal{W}$.

В задаче H-free Edge Deletion требуется по данному графу G и целому числу k определить, можно ли удалить из G не более k ребер так, чтобы получившийся граф не содержал индуцированного подграфа, изоморфного H. Будем говорить, что F — *решение* для инстанса (G,k) задачи H-free Edge Deletion, если $F \subseteq E(G)$, $|F| \leq k$ и $G - F$ не содержит индуцированного H. В задаче H-free Edge Completion требуется по данному графу G и целому числу k определить, можно ли добавить в G не более k ребер так, чтобы получившийся граф не содержал индуцированного подграфа, изоморфного H. Будем говорить, что F — *решение* для инстанса (G,k) задачи H-free Edge Completion, если $F \subseteq \overline{E}(G)$, $|F| \leq k$ и $G + F$ не содержит индуцированного H. В задаче H-free Edge Editing требуется по данному графу G и целому числу k определить, можно ли изменить (добавить или удалить) не более k ребер так, чтобы получившийся граф не содержал индуцированного подграфа, изоморфного H. Будем

Таблица 2: $\mathcal{H} \cup \overline{\mathcal{H}}$

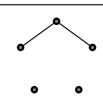
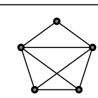
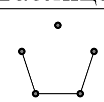
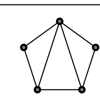
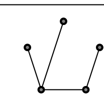
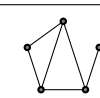
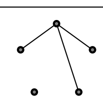
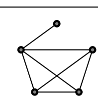
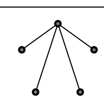
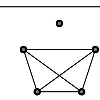
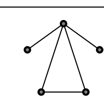
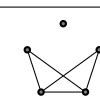
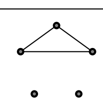
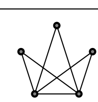
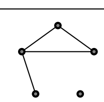
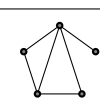
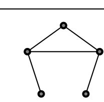
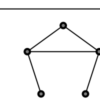
H_1		$\overline{H_1}$		H_4		$\overline{H_4}$		H_7		$\overline{H_7}$	
H_2		$\overline{H_2}$		H_5		$\overline{H_5}$		H_8		$\overline{H_8}$	
H_3		$\overline{H_3}$		H_6		$\overline{H_6}$		H_9		$\overline{H_9}$	

Таблица 3: $\mathcal{A} \cup \overline{\mathcal{A}}$

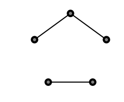
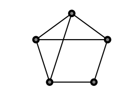
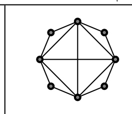
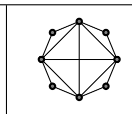
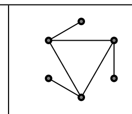
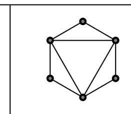
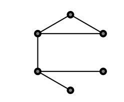
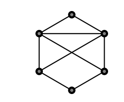
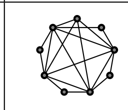
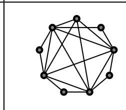
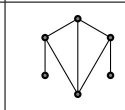
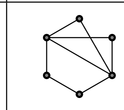
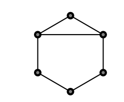
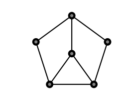
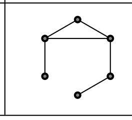
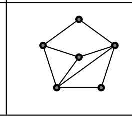
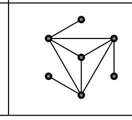
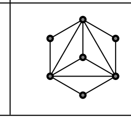
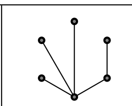
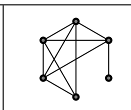
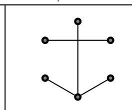
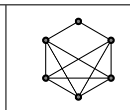
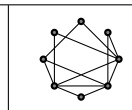
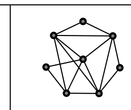
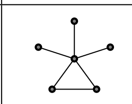
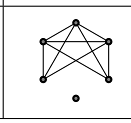
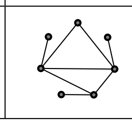
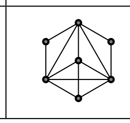
A_1		$\overline{A_1}$		A_4		$\overline{A_4}$		A_7		$\overline{A_7}$	
A_2		$\overline{A_2}$		A_5		$\overline{A_5}$		A_8		$\overline{A_8}$	
A_3		$\overline{A_3}$		A_6		$\overline{A_6}$		A_9		$\overline{A_9}$	

Таблица 4: $\mathcal{D} \cup \overline{\mathcal{D}} \cup \mathcal{B} \cup \overline{\mathcal{B}}$

D_1		$\overline{D_1}$		B_1		$\overline{B_1}$		B_3		$\overline{B_3}$	
D_2		$\overline{D_2}$		B_2		$\overline{B_2}$					

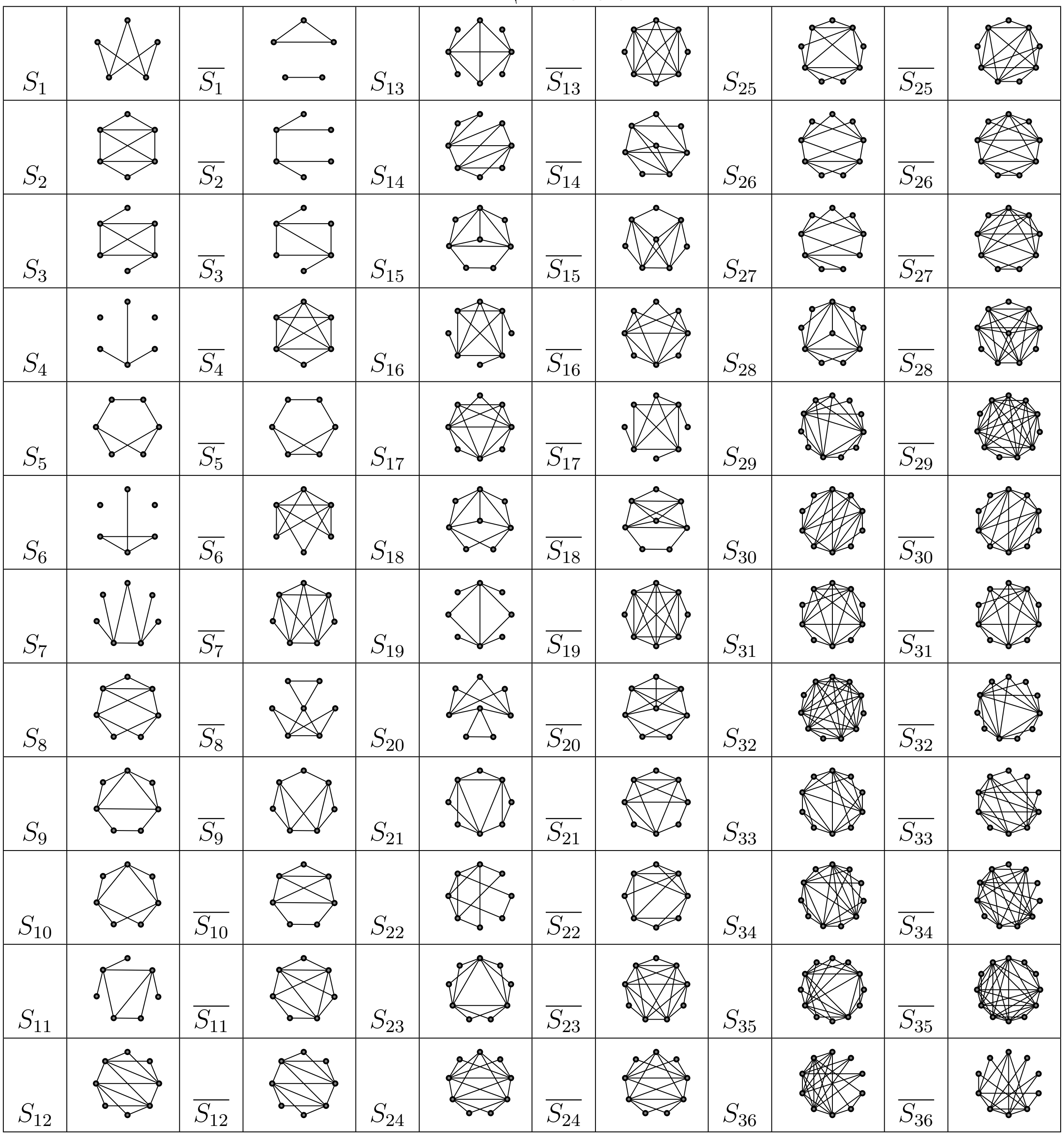
говорить, что F — *решение* для инстанса (G, k) задачи H-free Edge Editing, если F — множество пар вершин из G, $|F| \leq k$ и $G \triangle F$ не содержит индуцированного H. В оптимизационной версии задачи H-free Edge Deletion (Completion, Editing) на вход подается только граф G и требуется определить наименьшее k, при котором существует решение F для инстанса (G, k). Мы будем использовать обе версии этих задач, из контекста будет понятно, какую версию мы рассматриваем. Заметим, что задачи H-free Edge Deletion и $\overline{H}$-free Edge Completion эквивалентны, а также задачи H-free Edge Editing и $\overline{H}$-free Edge Editing эквивалентны, а именно верна следующая лемма.

Лемма 1. *(i) F — решение для инстанса (G, k) задачи H-free Edge Deletion $\Leftrightarrow$ F — решение для инстанса $(\overline{G}, k)$ задачи $\overline{H}$-free Edge Completion. (ii) F — решение для инстанса (G, k) задачи H-free Edge Editing $\Leftrightarrow$ F — решение для инстанса $(\overline{G}, k)$ задачи $\overline{H}$-free Edge Editing.*

В задаче Sandwich H-free Edge Deletion дан граф G и множество D *запрещенных* ребер. Ребра из $E(G) \setminus D$ называются *разрешенными*. Требуется определить, существует ли такое множество $F \subseteq E(G) \setminus D$, что $G - F$ не содержит H в качестве индуцированного подграфа. Такое F будем называть решением для инстанса (G, D) задачи Sandwich H-free Edge Deletion. В задаче Sandwich H-free Edge Completion дан граф G и множество D *запрещенных* антиребер. Антиребра из $\overline{E}(G) \setminus D$ называются *разрешенными*. Требуется определить, существует ли такое множество $F \subseteq \overline{E}(G) \setminus D$, что $G + F$ не содержит H в качестве индуцированного подграфа. Такое F будем называть решением для инстанса (G, D) задачи Sandwich H-free Edge Completion. В этих задачах размер решения F не важен, интересно только его существование. Заметим, что задачи Sandwich H-free Edge Deletion и Sandwich $\overline{H}$-free Edge Completion эквивалентны, а именно верна следующая лемма.

Лемма 2. *F — решение для инстанса (G, D) задачи Sandwich H-free Edge Deletion $\Leftrightarrow$ F — решение для инстанса $(\overline{G}, D)$ задачи Sandwich $\overline{H}$-free Edge Completion.*

Таблица 5: $\mathcal{S} \cup \overline{\mathcal{S}}$



Пусть $f : \mathbb{N} \to \mathbb{N}$ — неубывающая функция. $f(OPT)$-*приближением для инстанса* I некоторой минимизационной задачи будем называть такое решение F для инстанса I, что $|F| \leq f(OPT(I)) \cdot OPT(I)$, где $OPT(I)$ — размер оптимального решения для I. Алгоритм для минимизационной задачи называется $f(OPT)$-*аппроксимационным алгоритмом*, если для любого инстанса I этой задачи он находит $f(OPT)$-приближение для I. Будем говорить, что минимизационная задача *допускает* $f(OPT)$-*приближение*, если для нее существует $f(OPT)$-аппроксимационный алгоритм, работающий за полиномиальное время. Из леммы 1 получаем следующую лемму.

Таблица 6: $\mathcal{F} \cup \overline{\mathcal{F}}$

$\mathcal{F}_1$	$K_{2,t}$	$\overline{\mathcal{F}_1}$	$K_t \cup K_2$	$t \geq 4$
$\mathcal{F}_2$	$K_{1,t}$	$\overline{\mathcal{F}_2}$	$K_t \cup K_1$	$t \geq 5$
$\mathcal{F}_3$	$K_2 \boxtimes tK_1$	$\overline{\mathcal{F}_3}$	$K_t \cup 2K_1$	$t \geq 4$
$\mathcal{F}_4$	$T_{t,1}$	$\overline{\mathcal{F}_4}$	$\overline{T_{t,1}}$	$t \geq 4$
$\mathcal{F}_5$	$\overline{(K_t - e) \cup 2K_1}$	$\overline{\mathcal{F}_5}$	$(K_t - e) \cup 2K_1$	$t \geq 4$
$\mathcal{F}_6$	$\overline{(K_t - e) \cup K_2}$	$\overline{\mathcal{F}_6}$	$(K_t - e) \cup K_2$	$t \geq 4$
$\mathcal{F}_7$	$K_{1,t} \cup K_2$	$\overline{\mathcal{F}_7}$	$\overline{K_{1,t} \cup K_2}$	$t \geq 4$
$\mathcal{F}_8$	$\overline{(K_t - e) \cup K_1}$	$\overline{\mathcal{F}_8}$	$(K_t - e) \cup K_1$	$t \geq 6$
$\mathcal{F}_9$	J_t	$\overline{\mathcal{F}_9}$	$\overline{J_t}$	$t \geq 3$
$\mathcal{F}_{10}$	Q_t	$\overline{\mathcal{F}_{10}}$	$\overline{Q_t}$	$t \geq 3$

Лемма 3. *(i) H-free Edge Deletion допускает $f(OPT)$-приближение $\Leftrightarrow$ $\overline{H}$-free Edge Completion допускает $f(OPT)$-приближение. (ii) H-free Edge Editing допускает $f(OPT)$-приближение $\Leftrightarrow$ $\overline{H}$-free Edge Editing допускает $f(OPT)$-приближение.*

Далее мы будем часто пользоваться следующей леммой.

Лемма 4. *Пусть P_1 и P_2 — задачи, параметризованные размером решения. Пусть задача P_2 допускает $q(OPT)$-приближение для некоторого неубывающего полинома q, а p — такой полином, что $p(x) = q(x) \cdot x$. Пусть существуют такие полиномиальные алгоритмы A, S_1 и S_2, что для любого $k \geq 0$ и любого I' выполняется следующее: (i) F' — решение для инстанса (I', k) задачи $P_1 \Rightarrow S_1(A(I', k), I', F')$ — решение для инстанса $(A(I', k), k)$ задачи P_2; (ii) F — решение для инстанса $(A(I', k), p(k))$ задачи $P_2 \Rightarrow S_2(A(I', k), I', F)$ — решение для инстанса $(I', p(k))$ задачи P_1. Также пусть для каждого инстанса I' задачи P_1 размер оптимального решения $OPT(I')$ ограничен некоторым полиномом от $|I'|$. Тогда существует $q(OPT)$-приближение для задачи P_1.*

Доказательство. Пусть дан инстанс I' задачи P_1. Для каждого k построим инстанс $I = A(I', k)$ задачи P_2 и найдем $q(OPT)$-приближение F для I. Если $|F| \leq p(k)$, то найдем $F' = S_2(I, I', F)$, и из всех найденных F' выберем минимальное по размеру решение F_{min}. Покажем, что F_{min} является $q(OPT)$-приближением для I'. Заметим, что в какой-то момент мы переберем $k = OPT(I')$. Пусть F'_{OPT} — оптимальное решение для I'. Так как $S_1(I, I', F'_{OPT})$ выдает решение для I размера не больше чем k, то $OPT(I) \leq k$. Тогда мы найдем решение F для I размера не больше чем $q(OPT(I)) \cdot OPT(I) \leq q(k) \cdot k = p(k)$, а тогда $F_{min} \leq p(k) = q(OPT(I')) \cdot OPT(I')$. ☐

2 Простые случаи

В этом разделе мы рассмотрим графы H, для которых задачи H-free Edge Deletion (Completion, Editing) решаются за полиномиальное время или допускают константное приближение. В результате мы докажем пункты (i) и (ii) гипотез 1 и 2, а также пункт (i) гипотезы 3.

Лемма 5. *Задача $\overline{K_n}$-free Edge Deletion допускает полиномиальное решение.*

Доказательство. Пусть дан инстанс (G, k) задачи $\overline{K_n}$-free Edge Deletion. Если в G нет индуцированного подграфа, изоморфного $\overline{K_n}$, то $F = \emptyset$ является решением для (G, k). Если же в G есть индуцированный $\overline{K_n}$, то от него невозможно избавиться удалением ребер из G, поэтому для такого инстанса решения не существует. $\square$

Лемма 6. *Задача $\overline{K_n - e}$-free Edge Deletion допускает полиномиальное решение.*

Доказательство. Пусть дан инстанс (G, k) задачи $\overline{K_n - e}$-free Edge Deletion. Если в G нет индуцированного подграфа, изоморфного $\overline{K_n - e}$, то $F = \emptyset$ является решением для (G, k). Если же в G есть индуцированный подграф, изоморфный $\overline{K_n - e}$, то, чтобы от него избавиться, нам необходимо удалить конкретное ребро e', соответствующее ребру e в $\overline{K_n - e}$. Тогда решением для (G, k) будет решение для $(G - e', k - 1)$, объединенное с ребром e'. Таким образом, мы получили полиномиальный алгоритм, решающий $\overline{K_n - e}$-free Edge Deletion. $\square$

Лемма 7. *Задача K_n-free Edge Deletion допускает $\binom{n}{2}$-приближение.*

Доказательство. Пока в графе остается индуцированный подграф, изоморфный K_n, будем находить его и удалять все ребра в нем. Заметим, что из каждого индуцированного подграфа, изоморфного K_n, необходимо удалить хотя бы одно ребро, поэтому оптимальный ответ должен пересекать все клики, ребра которых мы удалили. Значит, мы удалили не более $\binom{n}{2} \cdot OPT(G)$, то есть получили константное приближение. $\square$

Лемма 8. *Задача K_n-free Edge Editing допускает $\binom{n}{2}$-приближение.*

Доказательство. Пусть дан инстанс $I_e = (G, k)$ задачи K_n-free Edge Editing, и F — решение для этого инстанса. Заметим, что $F' = F \cap E(G)$ является решением для инстанса $I_d = (G, k)$ задачи K_n-free Edge Deletion, поскольку $F' \subseteq E(G)$ и $G \triangle F'$ получается из $G \triangle F$ удалением ребер, а при такой операции не мог образоваться индуцированный K_n. Тогда $OPT(I_d) \leq OPT(I_e)$. Воспользуемся приближением для K_n-free Edge Deletion из леммы 7 и получим решение F' для I_d. Заметим, что $|F'| \leq \binom{n}{2} \cdot OPT(I_d) \leq \binom{n}{2} \cdot OPT(I_e)$, и что F' также является решением для I_e. Таким образом, получили $\binom{n}{2}$-приближение для I_e. $\square$

Воспользовавшись леммой 3, получаем следующие следствия.

Следствие 1. *Задача K_n-free Edge Completion допускает полиномиальное решение.*

Следствие 2. *Задача $(K_n - e)$-free Edge Completion допускает полиномиальное решение.*

Следствие 3. *Задача $\overline{K_n}$-free Edge Completion допускает $\binom{n}{2}$-приближение.*

Следствие 4. *Задача $\overline{K_n}$-free Edge Editing допускает $\binom{n}{2}$-приближение.*

3 Min Horn Deletion-полные задачи

В этом разделе мы рассмотрим графы H, для которых задачи H-free Edge Deletion (Completion, Editing) являются Min Horn Deletion-полными. В результате мы докажем пункт (iii) гипотез 1 и 2, а также пункт (ii) гипотезы 3.

Для начала опишем, какие задачи называются Min Horn Deletion-полными, и введем необходимые определения. В работе [11] была предпринята попытка классифицировать все задачи удовлетворения булевых ограничений в зависимости от сложности их аппроксимации. Для некоторых задач не удалось установить сложность аппроксимации, но зато удалось разбить все такие задачи на классы, в которых все задачи эквивалентны друг другу относительно A-сведений, которые мы определим ниже. Тогда для каждого такого класса можно выбрать любую задачу в качестве представителя. Одним из таких представителей выбрана задача Min Horn Deletion. В задаче Min Horn Deletion на вход подается формула φ в КНФ, каждый клоз которой содержит либо один произвольный литерал, либо три литерала, из которых ровно один отрицательный. Требуется найти значения переменных, при которых количество невыполненных клозов в φ минимально.

Будем говорить, что задача P *A-сводится* к задаче Q, или, что то же самое, существует *A-сведение* от P к Q, если существуют два полиномиальных алгоритма F и G, а также константа α такие, что выполняются следующие свойства:
(i) Пусть I — инстанс задачи P. Тогда $F(I)$ — инстанс задачи Q.
(ii) Пусть I — инстанс задачи P, а S — решение для $F(I)$. Тогда $G(I, S)$ — решение для I.
(iii) Пусть I — инстанс задачи P, и $r \geq 1$. Тогда если S является r-приближением для $F(I)$, то $G(I, S)$ является (αr)-приближением для I.

Заметим также, что A-сведения являются транзитивными, то есть если P A-сводится к Q, а Q A-сводится к R, то P A-сводится к R.

Пусть даны две задачи P и Q. Будем говорить, что P является *Q-полной относительно A-сведений*, или же просто *Q-полной*, если существуют A-сведения от P к Q и от Q к P. В работе [3] было показано, что задача $(K_n - e)$-free Edge Deletion является Min Horn Deletion-полной.

Лемма 9 (Теорема 3 в [3])**.** *Для любого $n \geq 5$ задача $(K_n - e)$-free Edge Deletion является Min Horn Deletion-полной относительно A-сведений.*

Пользуясь леммой 1, строим A-сведения с $\alpha = 1$ между $(K_n - e)$-free Edge Deletion и $\overline{K_n - e}$-free Edge Completion и получаем следующее следствие.

Следствие 5. *Для любого $n \geq 5$ задача $\overline{K_n - e}$-free Edge Completion является Min Horn Deletion-полной относительно A-сведений.*

Таким образом, для гипотез 1 и 2 пункт (iii) верен. Для того чтобы показать пункт (ii) гипотезы 3, нам необходимо получить аналогичный результат для $(K_n - e)$-free Edge Editing. Для доказательства леммы 9 сначала показывалось существование A-сведения от Min Horn Deletion-полной задачи к $(K_n - e)$-free Edge Deletion, а потом существование A-сведения от $(K_n - e)$-free Edge Deletion к Min Horn Deletion-полной задаче MinOnes($\mathcal{F}$) для некоторого множества ограничений $\mathcal{F}$.

В задаче MinOnes($\mathcal{F}$) на вход подается множество переменных X и набор булевых ограничений, каждое из которых берется из множества $\mathcal{F}$ и применяется к некоторому набору переменных из X. Требуется найти выполняющий набор для этих ограничений, содержащий минимальное возможное количество единиц. $\mathcal{F}$ для задачи фиксировано и не является частью входа. Для доказательства леммы 9 в качестве $\mathcal{F}$ в работе [3] использовалось множество $\{f_n, g_n\}$, где $f_n(x_1, x_2, \ldots, x_{\binom{n}{2}}) = 0$ тогда и только тогда, когда ровно одна из переменных принимает значение 1, $g_n(x_1, x_2, \ldots, x_{\binom{n}{2}-1}) = 0$ тогда и только тогда, когда все переменные принимают значение 0. В работе было показано, что для такого $\mathcal{F}$ задача MinOnes($\mathcal{F}$) является Min Horn Deletion-полной. Из результатов, полученных в работе [11], следует, что MinOnes($\mathcal{F}$)

является Min Horn Deletion-полной тогда и только тогда, когда Weighted MinOnes($\mathcal{F}$) является Min Horn Deletion-полной, где Weighted MinOnes($\mathcal{F}$) — это MinOnes($\mathcal{F}$), в котором каждой переменной назначен целый неотрицательный вес и требуется найти выполняющий набор, для которого суммарный вес переменных, которым присвоено значение 1, минимален. Таким образом, мы получили следующую лемму.

Лемма 10. *Для определенного выше $\mathcal{F}$ задача Weighted MinOnes($\mathcal{F}$) — Min Horn Deletion-полная относительно A-сведений.*

Нам понадобится этот факт для доказательства следующей леммы. Введем задачу Weighted H-free Edge Deletion, в которой по заданному графу G с целыми неотрицательными весами на ребрах требуется удалить ребра минимального суммарного веса так, чтобы в оставшемся графе не оказалось индуцированного графа H.

Лемма 11. *Для любого $n \geq 5$ задача Weighted $(K_n - e)$-free Edge Deletion является Min Horn Deletion-полной относительно A-сведений.*

Доказательство. Для доказательства Min Horn Deletion-трудности применим тривиальное сведение от задачи $(K_n - e)$-free Edge Deletion, сделав веса всех ребер равными 1. Такое сведение является A-сведением с $\alpha = 1$.

Для доказательства Min Horn Deletion-полноты построим A-сведение от Weighted $(K_n - e)$-free Edge Deletion к Weighted MinOnes($\mathcal{F}$), где $\mathcal{F} = \{f_n, g_n\}$ определено выше. Пусть дан инстанс G задачи Weighted $(K_n - e)$-free Edge Deletion. Для каждого ребра G введем уникальную переменную. Назначение этой переменной единицы будет соответствовать удалению ребра. Для каждого индуцированного $K_n - e$ в G добавим ограничение $g_n(x_{i_1}, \ldots, x_{i_{\binom{n}{2}-1}})$, где $x_{i_1}, \ldots, x_{i_{\binom{n}{2}-1}}$ — переменные, соответствующие ребрам этого подграфа. Такие ограничения соответствуют тому, что из каждого индуцированного $K_n - e$ должно быть удалено хотя бы одно ребро. Для каждого индуцированного K_n в G добавим ограничение $f_n(x_{i_1}, \ldots, x_{i_{\binom{n}{2}}})$, где $x_{i_1}, \ldots, x_{i_{\binom{n}{2}}}$ — переменные, соответствующие ребрам этого подграфа. Такие ограничения соответствуют тому, что из каждого индуцированного K_n может быть удалено либо ноль ребер, либо хотя бы два. Полученный инстанс Weighted MinOnes($\mathcal{F}$) назовем I.

Заметим, что $K_n - e$ мог получиться после удаления ребер только из индуцированных подграфов K_n и $K_n - e$. Значит, получившихся ограничений достаточно. Пусть $F \subseteq E(G)$. Пусть $\{x_i = 1 \Leftrightarrow e_i \in F\}$ — означивание переменных для I. Тогда F — решение для G тогда и только тогда, когда $\{x_i\}$ — выполняющий набор для I, и $w(F) = w(\{x_i\})$, где w — это вес решения. Получили A-сведение с $\alpha = 1$.

Таким образом, существует A-сведение от Min Horn Deletion-полной задачи к Weighted $(K_n - e)$-free Edge Deletion, и существует A-сведение от Weighted $(K_n - e)$-free Edge Deletion к Min Horn Deletion-полной задаче. Значит, Weighted $(K_n - e)$-free Edge Deletion является Min Horn Deletion-полной относительно A-сведений. □

Теперь мы можем перейти к доказательству того, задачи $(K_n - e)$-free Edge Editing — Min Horn Deletion-полные для $n \geq 5$.

Лемма 12. *Для $n \geq 5$ существует A-сведение от задачи $(K_n - e)$-free Edge Deletion к задаче $(K_n - e)$-free Edge Editing.*

Доказательство. Пусть G' — инстанс $(K_n - e)$-free Edge Deletion. Пусть $m = |E(G')|$, а C — это $K_n - e$, из которого удалено одно произвольное ребро.

Для каждого антиребра e' графа G' сделаем следующее. Создадим $m^2 + 1$ копию C и для каждой копии отождествим одно из ее антиребер с e'. Получившийся граф назовем G. Пусть F' — решение для G'. Покажем, что F' также является решением для инстанса G задачи $(K_n - e)$-free Edge Editing. Предположим, что это не так и в $G \triangle F'$ нашелся индуцированный $K_n - e$. Так как F' — решение для G', то индуцированный $K_n - e$ не может целиком находиться внутри $G' \triangle F'$. Целиком внутри некоторой копии C он лежать тоже не может, поскольку там меньше ребер, чем в $K_n - e$. Значит, $K_n - e$ пересекает какую-то копию C. Но эта копия отделена от остального графа 2-разделителем, а $K_n - e$ — трехсвязный при $n \geq 5$, значит, он не мог пересечь C. Таким образом, F' является решением для G.

Пусть теперь F — решение для G, и $|F| \leq m^2$. Покажем, что $F \cap E(G')$ является решением для G'. Пусть в $G' - (F \cap E(G'))$ нашелся индуцированный $K_n - e$. Если $F \cap \overline{E}(G') = \emptyset$, то $G' - (F \cap E(G'))$ — индуцированный подграф $G \triangle F$, а тогда $K_n - e$ в нем нет. Значит, $F \cap \overline{E}(G') \neq \emptyset$. Рассмотрим $e' \in F \cap \overline{E}(G')$. К антиребру e' в G подвешена $m^2 + 1$ копия C. Все эти копии не имеют общих ребер и антиребер, кроме e'. Так как $|F| \leq m^2$, то хотя бы для одной копии C ни одно из ее ребер и антиребер, кроме e', не принадлежит F. Но тогда при добавлении ребра e' C превращается в $K_n - e$, и в $G - F$ появится индуцированный $K_n - e$. Противоречие, значит, $F \cap E(G')$ — решение для G'.

Таким образом, если $F \leq m^2$, то мы можем перенести решение с $\alpha = 1$. Заметим, что $F' = E(G')$ является решением для G', а тогда и для G. Значит, $OPT(G) \leq m$. Тогда если $|F| > m^2$, то коэффициент приближения $r > m$. Если G' не содержит индуцированного $K_n - e$, то $OPT(G') = \emptyset$. Иначе выводим $F' := E(G')$, что является не более чем m-приближением для G', а соответственно, и r-приближением. Таким образом, мы получили A-сведение с $\alpha = 1$. $\square$

Лемма 13. *Для $n \geq 5$ существует A-сведение от задачи $(K_n - e)$-free Edge Editing к задаче Weighted $(K_n - e)$-free Edge Deletion.*

Доказательство. Для начала заметим следующее. Пусть мы хотим изменить минимальное возможное количество ребер в G' так, чтобы в нем не осталось $K_n - e$. Вместо этого можно добавить в граф все недостающие ребра из $\overline{E}(G')$ и удалять ребра из получившегося графа, при этом платить единицу штрафа за каждое оставленное ребро из $\overline{E}(G')$ и за каждое удаленное ребро из $E(G')$. Тогда минимальный возможный штраф, который нужно заплатить, чтобы в графе не осталось индуцированного $K_n - e$, будет равен минимальному количеству ребер, которое нужно изменить, чтобы в G' не осталось индуцированного $K_n - e$.

Давайте теперь построим инстанс задачи Weighted $(K_n - e)$-free Edge Deletion, который бы реализовывал эту идею со штрафованием оставленных ребер из $\overline{E}(G')$. Добавим в G' все отсутствующие ребра и для каждого добавленного ребра e' добавим в граф копию $K_n - e$, отождествив одно из ее ребер с e'. Назовем такую копию $C_{e'}$. Добавленным ребрам из $\overline{E}(G')$ назначим вес 0, всем остальным — вес 1. Полученный граф с весами назовем G. Пусть F' — решение для инстанса G' задачи $(K_n - e)$-free Edge Editing. Покажем, что его можно преобразовать в решение F для инстанса G задачи Weighted $(K_n - e)$-free Edge Deletion за полиномиальное время, при этом $|F'| = w(F)$, где $w(F)$ — вес решения F. Каждому $e' \in \overline{E}(G')$ сопоставим ребро $q \in E(C_{e'}) \setminus \{e'\}$. Пусть $F = (F' \cap E(G')) \cup (\overline{E}(G') \setminus F') \cup \{q_{e'} \mid e' \in F' \cap \overline{E}(G')\}$, другими словами, в F содержатся все антиребра графа $G' \triangle F'$, а для каждого антиребра G', попавшего в F', в F добавляется некоторое ребро из копии $K_n - e$, соответствующей этому антиребру.

Во-первых, заметим, что $|F'| = w(F)$. И правда, $w(F) = w(F' \cap E(G')) + w(\overline{E}(G') \setminus F') + w(\{q_{e'} \mid e' \in F' \cap \overline{E}(G')\}) = |F' \cap E(G')| + 0 + |F' \cap \overline{E}(G')| = |F'|$. Во-вторых, покажем, что $G - F$ не содержит индуцированных $K_n - e$. Заметим, что подграф $G - F$, индуцированный вершинами $V(G')$, совпадает с $G' \triangle F'$, поэтому $K_n - e$ не может лежать целиком внутри этого подграфа. Также $K_n - e$ не может лежать целиком внутри какого-то $C_{e'} - F$, так как для каждого $e' \in F'$ из $C_{e'}$ удаляется $q_{e'}$, а для каждого $e' \notin F'$ из $C_{e'}$ удаляется e', в любом случае получается, что в $C_{e'} - F$ на одно ребро меньше, чем в $K_n - e$. Значит, $K_n - e$ пересекает некоторое $C_{e'} - F$. Но так быть не может, поскольку $C_{e'} - F$ отделяется от остального графа двумя вершинами, а $K_n - e$ — трехсвязный. Таким образом, F — решение для G, и $|F'| = w(F)$.

Обратно, пусть F — решение для G. Покажем, что его можно преобразовать в решение F' для G' за полиномиальное время, и при этом $|F'| \leq w(F)$. Пусть $F' = (F \cap E(G')) \cup (\overline{F} \cap \overline{E}(G'))$. Заметим, что подграф $G - F$, индуцированный вершинами $V(G')$, совпадает с $G' \triangle F'$. Таким образом, в $G' \triangle F'$ нет индуцированного $K_n - e$. Осталось проверить, что $|F'| \leq w(F)$. $|F'| = |F \cap E(G')| + |\overline{F} \cap \overline{E}(G')| = w(F \cap E(G')) + |\overline{F} \cap \overline{E}(G')|$. Заметим, что для каждого ребра $e' \in \overline{F} \cap \overline{E}(G')$ множество F должно содержать хотя бы одно ребро из $E(C_{e'}) \setminus \{e'\}$, так как иначе в $G - F$ образуется индуцированный $K_n - e$. При этом для разных e' эти ребра различные, так как $C_{e'}$ не пересекаются по ребрам, и у всех этих ребер вес 1. Таким образом, мы показали, что $|\overline{F} \cap \overline{E}(G')| \leq w(F \setminus E(G'))$, а тогда $|F'| \leq w(F \cap E(G')) + w(F \setminus E(G')) = w(F)$.

Таким образом, мы построили A-сведение от $(K_n - e)$-free Edge Editing к Weighted $(K_n - e)$-free Edge Deletion с $\alpha = 1$. □

Объединяя леммы 12 и 13, получаем результат для задачи $(K_n - e)$-free Edge Editing. Пользуясь леммой 1, получаем результат для задачи $\overline{K_n - e}$-free Edge Editing.

Следствие 6. *Для любого $n \geq 5$ задача $(K_n - e)$-free Edge Editing является Min Horn Deletion-полной относительно A-сведений.*

Следствие 7. *Для любого $n \geq 5$ задача $\overline{K_n - e}$-free Edge Editing является Min Horn Deletion-полной относительно A-сведений.*

4 Задачи, не допускающие $poly(OPT)$-приближения при $P \neq NP$

В этом разделе мы рассмотрим некоторые классы графов H, для которых задачи H-free Edge Deletion (Completion, Editing) не допускают $poly(OPT)$-приближения при $P \neq NP$. При доказательстве отсутствия $poly(OPT)$-приближения для задач H-free Edge Deletion (Completion), мы будем активно использовать технику, введенную в работе [3]. Также мы покажем, что эту технику можно обобщить для доказательства отсутствия $poly(OPT)$-приближения для задач H-free Edge Editing.

Сначала мы будем доказывать NP-трудность задачи Sandwich H-free Edge Deletion или Sandwich H-free Edge Completion, строя полиномиальное сведение от задачи 3-SAT, являющейся одной из известнейших NP-трудных задач. В задаче 3-SAT на вход подается формула φ в КНФ, каждый клоз которой состоит из ровно трех литералов. Требуется определить, существует ли выполняющий набор для φ.

Для каждой из задач Sandwich H-free Edge Deletion и Sandwich H-free Edge Completion введем три типа гаджетов: Variable, Clause и Connector гаджеты. Сначала опишем эти гаджеты для задачи Sandwich H-free Edge Deletion. Variable гаджет — это копия H с двумя добавленными ребрами e_x и $e_{\neg x}$. Добавленные ребра являются разрешенными, а остальные — запрещенными. Удаление ребра e_x будет соответствовать тому, что $x = 1$, а удаление $e_{\neg x}$

— тому, что $x = 0$. Clause гаджет — это копия H, три ребра e_{ℓ_1}, e_{ℓ_2} и e_{ℓ_3} в которой являются разрешенными, а остальные — запрещенными. Удаление ребра e_{ℓ_i} будет соответствовать тому, что литерал ℓ_i выполнил клоз $(\ell_1 \vee \ell_2 \vee \ell_3)$. Connector гаджет — это копия H с одним добавленным ребром e_{in}, в которой e_{in} и еще одно ребро e_{out}, не имеющее общих концов с e_{in}, являются разрешенными, а остальные — запрещенными. С помощью Connector гаджетов Clause гаджеты будут соединяться с Variable гаджетами, чтобы ребра, соответствующие переменным и их литералам, удалялись согласованно.

Похожим образом определим эти гаджеты для задачи Sandwich H-free Edge Completion. Variable гаджет — это копия H с двумя удаленными ребрами e_x и $e_{\neg x}$. Полученные антиребра являются разрешенными, остальные — запрещенными. Добавление ребра e_x будет соответствовать тому, что $x = 1$, а добавление $e_{\neg x}$ — тому, что $x = 0$. Clause гаджет состоит из двух частей C_1 и C_2, имеющих общее антиребро. C_1 — это копия H, два антиребра e_{ℓ_1} и $e_{\ell_2 \vee \ell_3}$ которой являются разрешенными, а остальные — запрещенными. C_2 — это копия H с одним удаленным ребром. Полученное антиребро отождествляется с $e_{\ell_2 \vee \ell_3}$ в C_1. Антиребро $e_{\ell_2 \vee \ell_3}$ и еще два антиребра e_{ℓ_2} и e_{ℓ_3} являются разрешенными, остальные — запрещенными. Добавление ребер будет соответствовать тому, что соответствующие литералы или дизъюнкции литералов равны единице. Connector гаджет — это H с одним удаленным ребром e_{in}, в котором e_{in} и еще одно антиребро e_{out}, не имеющее общих концов с e_{in}, являются разрешенными, а остальные — запрещенными.

Для полноты изложения опишем конструкцию, представленную в работе [3].

Конструкция 1. *По инстансу φ задачи 3-SAT и множеству $S = \{Variable, Clause, Connector\}$ гаджетов для задачи Sandwich H-free Edge Deletion (Completion) построим граф G, состоящий из этих гаджетов. Для каждой переменной введем свою копию Variable гаджета, для каждого клоза — свою копию Clause гаджета. Для каждого клоза $c = (\ell_1 \vee \ell_2 \vee \ell_3)$ и каждого литерала $\ell \in c$ введем цепочку Connector гаджетов $G_1^{\ell,c}, G_2^{\ell,c}, \ldots, G_{p+2}^{\ell,c}$, где $p = |V(H)|$. Для каждого $1 \leq i \leq p+1$ ребро (антиребро) e_{out} гаджета $G_i^{\ell,c}$ отождествляется с ребром (антиребром) e_{in} гаджета $G_{i+1}^{\ell,c}$. Ребро (антиребро) e_{out} гаджета $G_{p+2}^{\ell,c}$ отождествляется с ребром (антиребром) e_ℓ Variable гаджета для переменной, соответствующей литералу ℓ. Ребро (антиребро) e_{in} гаджета $G_1^{\ell,c}$ отождествляется с ребром (антиребром) e_ℓ Clause гаджета для клоза c. Полученный граф назовем G, множество запрещенных ребер (антиребер) — D.*

В работе [3] показано, что по решению F для инстанса (G, D) задачи Sandwich H-free Edge Deletion (Completion) мы можем за полиномиальное время восстановить выполняющий набор для формулы φ. При этом доказательство работает для произвольного графа H. Обратное же верно только для графов H, обладающих определенными свойствами. В работе [3] показано, что по выполняющему набору для φ можно построить решение для (G, D) для трехсвязных H, содержащих хотя бы два антиребра. Нам же потребуется чуть более общее утверждение. Для этого рассмотрим еще одну конструкцию.

Конструкция 2. *Пусть даны инстанс φ задачи 3-SAT, выполняющий набор τ для φ и множество $S = \{$Variable, Clause, Connector$\}$ гаджетов для задачи Sandwich H-free Edge Deletion (Completion). Применим конструкцию 1 к формуле φ и гаджетам $\{$Variable, Clause, Connector$\}$, получим (G, D) — инстанс задачи Sandwich H-free Edge Deletion (Completion). Пусть $F \subseteq E(G)$ $(F \subseteq \overline{E}(G))$ содержит все ребра (антиребра) e_ℓ из Variable и Clause гаджетов, где $\tau(\ell) = 1$. Также F содержит все ребра (антиребра) e_{in} и e_{out} из Connector гаджетов $G_i^{\ell,c}$, где $\tau(\ell) = 1$. Результатом конструкции будет граф $G \triangle F$.*

Будем говорить, что H *пересекает* множество {Variable, Clause, Connector} гаджетов, если существуют инстанс φ задачи 3-SAT и выполняющий набор τ для φ, что если $\widetilde{G}$ — результат применения конструкции 2 к φ, τ и {Variable, Clause, Connector}, то в $\widetilde{G}$ найдется индуцированный подграф, изоморфный H и содержащий внутренние вершины хотя бы двух разных гаджетов.

Лемма 14. *Пусть дан граф H, а также Variable, Clause и Connector гаджеты для задачи Sandwich H-free Edge Deletion (Completion), и пусть H не пересекает эти гаджеты. Тогда задача Sandwich H-free Edge Deletion (Completion) — NP-трудная.*

Доказательство. Пусть φ — инстанс задачи 3-SAT. Применим конструкцию 1 к гаджетам и формуле φ, получим (G, D) — инстанс задачи Sandwich H-free Edge Deletion (Completion). Если существует выполняющий набор τ для φ, то по нему построим F так, как было описано в конструкции 2. Так как H не пересекает гаджеты по условию, а целиком внутри гаджета он не может содержаться по построению F, то $G - F$ не содержит индуцированного H, а значит, F — решение для (G, D). Обратно, если F — решение для (G, D), то при любом H существует выполняющий набор для φ, как обсуждалось выше. Таким образом, мы получили полиномиальное сведение 3-SAT к Sandwich H-free Edge Deletion (Completion), а значит, задача Sandwich H-free Edge Deletion (Completion) — NP-трудная. □

Из леммы 2 получаем следующее следствие.

Следствие 8. *Sandwich H-free Edge Deletion является NP-трудной $\Leftrightarrow$ Sandwich $\overline{H}$-free Edge Completion является NP-трудной.*

Введем понятия еще двух гаджетов: Deletion Enforcer и Completion Enforcer. Deletion Enforcer для графа H — это H с одним добавленным ребром, а Completion Enforcer — это H с одним удаленным ребром. Добавленное (удаленное) ребро назовем *выделенным.* С помощью этих двух гаджетов мы сможем показывать отсутствие $poly(OPT)$-приближения для H-free Edge Deletion (Completion, Editing) при $P \neq NP$. Для этого введем еще две конструкции.

Конструкция 3. *Даны графы H и G', а также множество $F \subseteq E(G')$ и целое число $p \geq 0$. Пусть X — Deletion Enforcer для H. Для каждого ребра $e \in F$ построим $p+1$ копию X и для каждой копии отождествим ее выделенное ребро с e. Полученный граф назовем G.*

Конструкция 4. *Даны графы H и G', а также множество $F \subseteq \overline{E}(G')$ и целое число $p \geq 0$. Пусть X — Completion Enforcer для H. Для каждого антиребра $e' \in F$ построим $p+1$ копию X и для каждой копии отождествим ее выделенное антиребро с e'. Полученный граф назовем G.*

Будем говорить, что H *пересекает* Deletion (Completion) Enforcer X, если существует такой граф G, что если объединить X и G, отождествив выделенное ребро (антиребро) X с некоторым ребром (антиребром) G, то в полученном графе найдется индуцированный H, содержащий вершину из $V(X) \setminus V(G)$ и вершину из $V(G) \setminus V(X)$.

Лемма 15. *Пусть даны граф H и Deletion Enforcer X для H. Пусть H не пересекает X. Тогда (i) Sandwich H-free Edge Deletion — NP-трудная $\Rightarrow$ H-free Edge Deletion не допускает $poly(OPT)$-приближения при $P \neq NP$; (ii) H-free Edge Completion не допускает $poly(OPT)$-приближения $\Rightarrow$ H-free Edge Editing не допускает $poly(OPT)$-приближения.*

Доказательство. Первый пункт доказывается в работе [3] для трехсвязных H с хотя бы одним антиребром. Это доказательство естественным образом обобщается на интересующие нас H. Для полноты изложения приведем это доказательство. Пусть H-free Edge Deletion допускает $q(OPT)$-приближение для некоторого полинома q. Введем полином $p(x) = q(x) \cdot x$. Пусть дан инстанс (G', D) задачи Sandwich H-free Edge Deletion. Пусть $k = |D|$. Применим конструкцию 3 к $(H, G', D, p(k))$, получим граф G. Пусть F' — решение для (G', D). Заметим, что F' также является решением для инстанса (G, k) задачи H-free Edge Deletion, так как граф H не может целиком находиться ни внутри $G' - F'$, ни Deletion Enforcer'а, и пересекать Deletion Enforcer он тоже не может. Раз для H-free Edge Deletion есть $q(OPT)$-приближение, значит, мы можем найти решение F для $(G, p(k))$ за полином. Покажем, что $F \cap E(G')$ является решением для (G', D). Так как в $G - F$ нет индуцированного H, то и в $G' - (F \cap E(G'))$ нет индуцированного H. Значит, осталось проверить только то, что $F \cap D = \emptyset$. Пусть это не так и $e \in F \cap D$. Тогда после удаления e из G появляется $p(k) + 1$ копия графа H. Причем ребра этих копий не пересекаются. Тогда, так как $|F| \leq p(k)$, найдется копия, ни одно из ребер которой не содержится в F. Но тогда в G найдется H, противоречие. Таким образом, если H-free Edge Deletion допускает $q(OPT)$-приближение, то мы можем решить Sandwich H-free Edge Deletion за полиномиальное время, что невозможно, если $P \neq NP$. Значит, при $P \neq NP$ задача H-free Edge Deletion не допускает $poly(OPT)$-приближения.

Для доказательства второго пункта проведем похожие рассуждения. Пусть H-free Edge Completion допускает $q(OPT)$-приближение для некоторого полинома q. Введем полином $p(x) = q(x) \cdot x$. Пусть (G', k) — инстанс задачи H-free Edge Completion. Применим конструкцию 3 к $(H, G', E(G'), p(k))$, получим граф G. Пусть F' — решение для (G', k). Покажем, что F' также является решением для инстанса (G, k) задачи H-free Edge Editing. Если в $G \triangle F'$ есть индуцированный H, то он не мог попасть в $G' \triangle F'$, так как F' — решение для (G', k). Значит, H содержит вершину из какого-то Deletion Enforcer'а. По условию H не пересекает Deletion Enforcer. Тогда H целиком лежит внутри $X \triangle F'$ для некоторой копии X. Но это невозможно, так как F' не содержит ребер из Deletion Enforcer'ов, а тогда $X \triangle F' = X$, а H не является индуцированным подграфом X. Таким образом, F' — решение для (G, k). Обратно, пусть F — решение для инстанса $(G, p(k))$ задачи H-free Edge Editing. Покажем, что $F \cap \overline{E}(G')$ — решение для инстанса $(G', p(k))$ задачи H-free Edge Completion. Понятно, что $G' \triangle (F \cap G')$ не содержит индуцированного H, так как G' является подграфом G. Значит, осталось только показать, что $F \cap E(G') = \emptyset$. Пусть это не так и $e \in F \cap E(G')$. Тогда после удаления e из G появляется $p(k)+1$ копия графа H. Причем эти копии пересекаются только по ребру e. Тогда, так как $|F| \leq p(k)$, найдется копия, ни одно из ребер или антиребер которой не содержится в F. Но тогда в $G \triangle F$ найдется индуцированный H, противоречие. По лемме 4 получаем, что если для задачи H-free Edge Editing существует $poly(OPT)$-приближение, то и для задачи H-free Edge Completion оно существует. □

Аналогичным образом, используя конструкцию 4, можно доказать и следующую лемму.

Лемма 16. *Пусть дан граф H и Completion Enforcer X для H. Пусть H не пересекает X. Тогда (i) Sandwich H-free Edge Completion — NP-трудная $\Rightarrow$ H-free Edge Completion не допускает $poly(OPT)$-приближения при $P \neq NP$; (ii) H-free Edge Deletion не допускает $poly(OPT)$-приближения $\Rightarrow$ H-free Edge Editing не допускает $poly(OPT)$-приближения.*

В работе [3] было доказано отсутствие $poly(OPT)$-приближений для задач H-free Edge Deletion и H-free Edge Completion для трехсвязных графов H с хотя бы двумя антиребрами, а также для графов H, являющихся циклами на хотя бы четырех вершинах или путями на хотя бы пяти вершинах. Комбинируя эти результаты с леммой 3, получаем следующие две леммы.

Лемма 17. *Пусть H — это трехсвязный граф с хотя бы двумя антиребрами. Тогда задачи H-free Edge Deletion (Completion) и $\overline{H}$-free Edge Deletion (Completion) не допускают $poly(OPT)$-приближения при $P \neq NP$.*

Лемма 18. *Пусть H — это цикл на хотя бы четырех вершинах или путь на хотя бы пяти вершинах. Тогда задачи H-free Edge Deletion (Completion) и $\overline{H}$-free Edge Deletion (Completion) не допускают $poly(OPT)$-приближения при $P \neq NP$.*

Докажем аналогичные результаты для задач H-free Edge Editing.

Лемма 19. *Пусть H — это трехсвязный граф с хотя бы двумя антиребрами. Тогда задачи H-free Edge Editing и $\overline{H}$-free Edge Editing не допускают $poly(OPT)$-приближения при $P \neq NP$.*

Доказательство. По лемме 17 H-free Edge Deletion не допускает $poly(OPT)$-приближения. Возьмем копию H и удалим из нее произвольное ребро. Получим Completion Enforcer X для H. Так как H — трехсвязный граф, то он не пересекает X. Тогда по лемме 16 у задачи H-free Edge Editing нет $poly(OPT)$-приближения при $P \neq NP$. По лемме 3 у задачи $\overline{H}$-free Edge Editing тоже нет $poly(OPT)$-приближения при $P \neq NP$. □

Лемма 20. *Пусть H — это цикл на хотя бы четырех вершинах или путь на хотя бы пяти вершинах. Тогда задачи H-free Edge Editing и $\overline{H}$-free Edge Editing не допускают $poly(OPT)$-приближения при $P \neq NP$.*

Доказательство. Пусть H — это цикл на хотя бы четырех вершинах. По лемме 18 задача H-free Edge Completion не допускает $poly(OPT)$-приближения при $P \neq NP$. Пусть X — это H с одним добавленным ребром. Заметим, что X — это Deletion Enforcer для H, и H не может пересечь X, так как у циклов нет 2-разделителей, являющихся ребрами. Таким образом, по лемме 15 мы получаем, что задача H-free Edge Editing не допускает $poly(OPT)$-приближения при $P \neq NP$. По лемме 3 задача $\overline{H}$-free Edge Editing тоже не допускает $poly(OPT)$-приближения при $P \neq NP$.

Для $H = P_{\geq 6}$ воспользуемся леммой 17, так как $\overline{H}$ — это трехсвязный граф с хотя бы двумя антиребрами. Остался только случай $H = P_5$. По лемме 18 задача $\overline{H}$-free Edge Deletion не допускает $poly(OPT)$-приближения при $P \neq NP$. Пусть X — это $\overline{H}$ с удаленным ребром между двумя вершинами степени два. Тогда X — это Completion Enforcer для $\overline{H}$. Предположим, что $\overline{H}$ пересекает X. Но тогда у $\overline{H}$ либо должна быть точка сочленения, либо должен существовать 2-разделитель, по одну сторону от которого будет индуцированный P_4, но это не так. Таким образом, $\overline{H}$ не пересекает X, и по лемме 16 мы получаем, что задача $\overline{H}$-free Edge Editing не допускает $poly(OPT)$-приближения при $P \neq NP$. По лемме 3 задача H-free Edge Editing тоже не допускает $poly(OPT)$-приближения при $P \neq NP$. □

Лемма 21. *Пусть H — регулярный граф, при этом не пустой и не полный. Тогда задачи H-free Edge Deletion, H-free Edge Completion и H-free Edge Editing не допускают $poly(OPT)$-приближения при $P \neq NP$.*

Доказательство. В теореме 3.7 в работе [14] доказывается, что если граф H регулярный, но не пустой и не полный, то H или $\overline{H}$ — это трехсвязный граф с хотя бы двумя антиребрами или цикл на хотя бы четырех вершинах. Но тогда по леммам 17, 18, 19 и 20 задачи H-free Edge Deletion, H-free Edge Completion и H-free Edge Editing не допускают $poly(OPT)$-приближения при $P \neq NP$. □

5 Общий случай

Для каждой из гипотез 1, 2 и 3 мы доказали все пункты, кроме последнего. В этом разделе мы покажем, что для завершения доказательства достаточно доказать отсутствие $poly(OPT)$-приближения для задач H-free Edge Deletion (Completion, Editing) только для определенного множества графов H. Для этого мы введем алгоритм, который по графу H будет получать граф H' такой, что если существует $poly(OPT)$-приближение для задачи H-free Edge Deletion (Completion, Editing), то существует $poly(OPT)$-приближение и для задачи H'-free Edge Deletion (Completion, Editing). Нам понадобится следующая конструкция, введенная в работе [2].

Конструкция 5. *Даны (G', p, H, V'), где G', H — графы, $V' \subseteq V(H)$, $p \in \mathbb{N}$. Для каждой инъективной функции $f : V' \to V(G')$ делаем следующее. Добавляем множества вершин $V_1, \ldots, V_{p+1}$ и вводим биекции $g_1, \ldots, g_{p+1}$ такие, что $g_i : V(H) \to (f(V') \cup V_i)$ и $\forall v \in V'$ $g(v) = f(v)$. Для каждого V_i добавляем ребра: $E_i = \{(u, v) \mid u \in (f(V') \cup V_i), v_i \in V_i, (g_i^{-1}(u), g_i^{-1}(v)) \in E(H)\}$. Получившийся граф назовем G.*

Неформально говоря, при построении этой конструкции мы рассмотрели все потенциально возможные способы найти индуцированный $H[V']$ в $G' \triangle F$ для произвольного F и для каждого такого способа добавили $p + 1$ надстройку, достраивающую H.

Лемма 22. *Пусть G — это результат применения конструкции 5 к (G', p, H, V') и $H' = H[V']$. Пусть F — решение для инстанса (G, p) задачи H-free Edge Deletion (Completion, Editing). Тогда $F \cap G'$ является решением для инстанса (G', p) задачи H'-free Edge Deletion (Completion, Editing).*

Доказательство. Предположим, что это не так и $F \cap G'$ не является решением для (G', p). Тогда $(G' \triangle (F \cap G'))[\widetilde{V}] = H'$ для некоторого $\widetilde{V} \subseteq V(G')$. Тогда среди надстроек $V_1, \ldots, V_{p+1}$, соответствующих $\widetilde{V}$ и биекции между V' и $\widetilde{V}$, найдется надстройка V_i, ребра которой не содержатся в F. Но тогда $(G \triangle F)[\widetilde{V} \cup V_i] = H$, и мы получаем противоречие с тем, что F — решение для (G, p). □

Для доказательства некоторых из дальнейших утверждений нам потребуется еще и обратное утверждение, а именно что если F' — решение для инстанса (G', k) задачи H'-free Edge Deletion (Editing), то F' является решением для инстанса (G, k) задачи H-free Edge Deletion (Editing). Однако это утверждение выполняется не для всех H и H', поэтому его придется доказывать в каждом случае отдельно.

Лемма 23. *Пусть H — нерегулярный граф, и пусть $H' = H - V_\ell$. Тогда если для H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, то и для H'-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение.*

Доказательство. Пусть задача H-free Edge Deletion (Completion, Editing) допускает $q(OPT)$-приближение для некоторого полинома q. Введем полином $p(x) = q(x) \cdot x$. Пусть дан инстанс (G', k) задачи H'-free Edge Deletion (Completion, Editing). Применим конструкцию 5 к $(G', p(k), H, V(H'))$, получим граф G.

Пусть F' — решение для (G', k). Покажем, что F' также является решением для инстанса (G, k) задачи H-free Edge Deletion (Completion, Editing). Предположим, что это не так, тогда в $G \triangle F'$ нашелся индуцированный H. Рассмотрим вершины H, попавшие в $V(G) \setminus V(G')$.

Степень любой вершины из $V(G) \setminus V(G')$ равна ℓ в $G \triangle F'$, а значит, вершины H со степенью больше чем ℓ не могли попасть в $V(G) \setminus V(G')$. Но тогда H' целиком лежит в $G' \triangle F'$, получаем противоречие с тем, что F' — решение для (G', k).

Обратно, пусть F — решение для $(G, p(k))$ задачи H-free Edge Deletion (Completion, Editing). Тогда по лемме 22 $F \cap G'$ является решением для $(G', p(k))$ задачи H'-free Edge Deletion (Completion, Editing). Тогда по лемме 4 мы получаем, что из существования $poly(OPT)$-приближения для H-free Edge Deletion (Completion, Editing) следует существование $poly(OPT)$-приближения для H'-free Edge Deletion (Completion, Editing). □

Лемма 24. *Пусть H — нерегулярный граф, и пусть $H' = H - V_h$. Тогда если для H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, то и для H'-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение.*

Доказательство. Пусть для задачи H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение. Тогда по лемме 3 для задачи $\overline{H}$-free Edge Completion (Deletion, Editing) тоже существует $poly(OPT)$-приближение, и по лемме 23 для задачи $(\overline{H} - V_\ell(\overline{H}))$-free Edge Completion (Deletion, Editing) существует $poly(OPT)$-приближение. $\overline{H} - V_\ell(\overline{H}) = \overline{H - V_h}$, поэтому для задачи $\overline{H - V_h}$-free Edge Completion (Deletion, Editing) существует $poly(OPT)$-приближение, а тогда по лемме 3 и для задачи $(H - V_h)$-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение. □

Наш алгоритм мы будем строить по образу алгоритма из статьи [14], посвященной существованию полиномиальных ядер для задач H-free Edge Deletion (Completion, Editing). В статье вводились следующие множества.

$$\begin{aligned}
\mathcal{X}_D &= \{C_\ell, \overline{C_\ell} \mid \ell \geq 4\} \\
&\quad \cup \{P_\ell, \overline{P_\ell} \mid \ell \geq 5\} \\
&\quad \cup \{H \mid H \text{ — регулярный, но не пустой и не полный}\} \\
&\quad \cup \{H \mid H \text{ — трехсвязный, но не полный}\} \\
&\quad \cup \{H \mid \overline{H}\text{— трехсвязный с хотя бы двумя антиребрами}\} \\
\mathcal{X}_E &= \mathcal{X}_D \cup \{\overline{K_n - e} \mid n \geq 5\} \\
\mathcal{Y}_E &= \{K_n, \overline{K_n} \mid n \geq 1\} \\
&\quad \cup \{H \mid H \neq C_4, \overline{H} \neq C_4, |V(H)| \leq 4\} \\
\mathcal{Y}_D &= \mathcal{Y}_E \cup \{\overline{K_n - e} \mid n \geq 5\}
\end{aligned}$$

В $\mathcal{X}_D$ и $\mathcal{X}_E$ попадали графы, для которых задачи H-free Edge Deletion и H-free Edge Editing соответственно не допускали полиномиального ядра, а в $\mathcal{Y}_D$ и $\mathcal{Y}_E$ попадали графы на менее чем пяти вершинах (кроме C_4) и графы, для которых соответствующие задачи допускали полиномиальное ядро. По аналогии построим множества $\mathcal{X}$ и $\mathcal{Y}$, подходящие для наших целей.

$$
\begin{aligned}
\mathcal{X} = &\{C_\ell, \overline{C_\ell} \mid \ell \geq 4\} \\
&\cup \{P_\ell, \overline{P_\ell} \mid \ell \geq 5\} \\
&\cup \{H \mid H \text{ — регулярный, но не пустой и не полный}\} \\
&\cup \{H \mid H \text{ — трехсвязный с хотя бы двумя антиребрами}\} \\
&\cup \{H \mid \overline{H}\text{— трехсвязный с хотя бы двумя антиребрами}\} \\
\mathcal{Y} = &\{K_n, \overline{K_n} \mid n \geq 1\} \\
&\cup \{K_n - e, \overline{K_n - e} \mid n \geq 5\} \\
&\cup \{H \mid H \neq C_4, \overline{H} \neq C_4, |V(H)| \leq 4\}
\end{aligned}
$$

Заметим, что $\mathcal{X} = \mathcal{X}_D \setminus \{K_n - e \mid n \geq 5\}$, и $\mathcal{Y} = \mathcal{Y}_D \cup \{K_n - e \mid n \geq 5\}$. Также заметим, что $\mathcal{X}_D \cup \mathcal{Y}_D = \mathcal{X}_E \cup \mathcal{Y}_E = \mathcal{X} \cup \mathcal{Y} = \overline{\mathcal{X}} \cup \overline{\mathcal{Y}}$. Для того чтобы завершить доказательство гипотез 1, 2 и 3, нам осталось доказать только последние их пункты, а именно достаточно доказать, что для графов $H \notin \mathcal{Y}$ задачи H-free Edge Deletion (Completion, Editing) не допускают $poly(OPT)$-приближений. Давайте покажем, что достаточно доказать отсутствие $poly(OPT)$-приближений не для всех графов $H \notin \mathcal{Y}$, а только для какого-то их подмножества. Для этого рассмотрим следующий алгоритм.

Churn(H) :

Шаг 1: Если H регулярный, то вернуть H.

Шаг 2: Если $H - V_\ell \notin \mathcal{Y}$, то вернуть Churn$(H - V_\ell)$.

Шаг 3: Если $H - V_h \notin \mathcal{Y}$, то вернуть Churn$(H - V_h)$.

Шаг 4: Вернуть H.

Лемма 25. *Если H-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение, то и Churn(H)-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение.*

Доказательство. Churn(H) получается из H конечной последовательностью удалений V_ℓ или V_h. Проведем индукцию по длине s этой последовательности. При $s = 0$ получаем, что Churn$(H) = H$ и утверждение очевидно. Пусть $s \geq 1$. Пусть H' — это $H - V_\ell$, если $H - V_\ell \notin \mathcal{Y}$, и $H - V_h$ иначе. Тогда по леммам 23 и 24 если H-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение, то H'-free Edge Deletion (Completion, Editing) тоже допускает $poly(OPT)$-приближение. А тогда по индукционному предположению Churn(H')-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение. Так как Churn$(H) =$ Churn(H'), то Churn(H)-free Edge Deletion (Completion, Editing) тоже допускает $poly(OPT)$-приближение. □

Таким образом, если существует граф $H \notin \mathcal{Y}$ такой, что для задачи H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, то для задачи Churn(H)-free Edge Deletion (Completion, Editing) тоже существует $poly(OPT)$ приближение, поэтому достаточно доказывать отсутствие $poly(OPT)$-приближения только для графов, равных Churn(H) для некоторого $H \notin \mathcal{Y}$. Таким образом, нам интересно понять, какие графы образуют множество $\{\text{Churn}(H) \mid H \notin \mathcal{Y}\}$. Во-первых, это могут быть регулярные графы. Заметим, что Churn(H) запускается на не пустом и не полном графе H, при этом в ходе алгоритма мы никогда не переходим к пустому или полному графу, так как такой граф лежит в $\mathcal{Y}$. Таким образом, если мы получили регулярный граф на выходе алгоритма, то этот граф не пустой и не полный, а значит, лежит в $\mathcal{X}$. Если мы получили граф H, не являющийся регулярным, то мы знаем, что $H - V_\ell \in \mathcal{Y}$ и $H - V_h \in \mathcal{Y}$.

Пусть $\mathcal{W}$ — множество графов, определенное в разделе 1. В работе [14] в леммах 3.8 — 3.23 показали следующий факт.

Лемма 26. *Пусть $H \notin \mathcal{X} \cup \mathcal{Y}$ и пусть $H - V_\ell \in \mathcal{Y}_D$ и $H - V_h \in \mathcal{Y}_D$. Тогда $H \in \mathcal{W}$.*

Осталось понять, что происходит с графами H, для которых $\{H - V_\ell, H - V_h\} = \{K_n - e, S\}$, где $n \geq 5$ и $S \in \mathcal{Y}$. Введем обозначения $\{V_1, V_2\} = \{V_\ell, V_h\}$ такие, что $H - V_1 = K_n - e$, а $H - V_2 \in \mathcal{Y}$. Пусть H такой, что $H - V_1 = K_n - e$, а $H - V_2 \in \mathcal{Y}_E \cup \{K_m - e \mid m \geq 5\}$. Рассмотрим $\overline{H}$. Для него $\overline{H} - V_1 = \overline{K_n - e} \in \mathcal{Y}_D$, $\overline{H} - V_2 \in \overline{\mathcal{Y}_E} \cup \{\overline{K_m - e} \mid m \geq 5\} = \mathcal{Y}_D$. При этом V_1 и V_2 — это множества вершин наименьшей и наибольшей степени в $\overline{H}$. Тогда по лемме 26 если $\overline{H} \notin \mathcal{X} \cup \mathcal{Y}$, то $\overline{H} \in \mathcal{W}$. Тогда если $H \notin \overline{\mathcal{X}} \cup \overline{\mathcal{Y}}$, то $H \in \overline{\mathcal{W}}$, что эквивалентно тому, что если $H \notin \mathcal{X} \cup \mathcal{Y}$, то $H \in \mathcal{W}$, поскольку $\overline{\mathcal{X}} \cup \overline{\mathcal{Y}} = \mathcal{X} \cup \mathcal{Y}$, и $\overline{\mathcal{W}} = \mathcal{W}$. Таким образом, остался только случай, когда $H - V_1 = K_n - e$, а $H - V_2 = \overline{K_m - e}$, где $n, m \geq 5$. Назовем множество таких графов H множеством $\mathcal{U}$. Таким образом, получаем следующую лемму.

Лемма 27. *Пусть $H \notin \mathcal{X} \cup \mathcal{Y}$ и пусть $H - V_\ell \in \mathcal{Y}$ и $H - V_h \in \mathcal{Y}$. Тогда $H \in \mathcal{W} \cup \mathcal{U}$.*

Следствие 9. *Пусть $H \notin \mathcal{Y}$. Тогда Churn$(H) \in \mathcal{X} \cup \mathcal{W} \cup \mathcal{U}$.*

Назовем множество графов $\mathcal{L}$ *хорошим для Deletion (Completion, Editing) задач*, если для него выполнено следующее свойство. Пусть $P \neq NP$. Пусть в H хотя бы пять вершин, $H \notin \mathcal{Y}$ и задача H-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение. Тогда найдется такой граф $H' \in \mathcal{L}$, что задача H'-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение и $|V(H')| \leq |V(H)|$. Назовем множество $\mathcal{L}$ *хорошим*, если оно является хорошим для Deletion, Completion и Editing задач одновременно.

Лемма 28. *Пусть $\mathcal{L}$ — хорошее множество для Deletion (Completion, Editing) задач. Тогда если для всех графов $H \in \mathcal{L}$ задача H-free Edge Deletion (Completion, Editing) не допускает $poly(OPT)$-приближения при $P \neq NP$, то и для всех графов $H \notin \mathcal{Y}$ задача H-free Edge Deletion (Completion, Editing) не допускает $poly(OPT)$-приближения при $P \neq NP$.*

Доказательство. Пусть $P \neq NP$ и граф $H \notin \mathcal{Y}$ такой, что задача H-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение. Если в H хотя бы пять вершин, то по определению хорошего множества для Deletion (Completion, Editing) задач в $\mathcal{L}$ найдется граф H' такой, что задача H-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение, а по условию это не так. Значит, в H не более четырех вершин. Тогда H — это C_4 или $\overline{C_4}$. Но по леммам 18 и 20 задачи C_4-free Edge Deletion (Completion, Editing) и $\overline{C_4}$-free Edge Deletion (Completion, Editing) не допускают $poly(OPT)$-приближения при $P \neq NP$. □

Лемма 29. *Множество $\mathcal{W} \cup \mathcal{U}$ — хорошее.*

Доказательство. Пусть $P \neq NP$. Пусть в H хотя бы пять вершин, $H \notin \mathcal{Y}$ и задача H-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение. По лемме 25 Churn(H) тоже допускает $poly(OPT)$-приближение. По следствию 9 Churn$(H) \in \mathcal{X} \cup \mathcal{W} \cup \mathcal{U}$. Задачи, соответствующие графам из $\mathcal{X}$, не допускают $poly(OPT)$-приближения при $P \neq NP$, а значит, интересующий нас граф $H' = $ Churn(H) находится в $\mathcal{W} \cup \mathcal{U}$. Осталось только заметить, что Churn(H) не увеличивает количество вершин в графе. □

Суть хорошего множества в том, что вместо того, чтобы доказывать последний пункт гипотез для всех $H \notin \mathcal{Y}$, мы можем доказывать его только для графов из этого множества. На данный момент мы смогли ограничиться множеством $\mathcal{W} \cup \mathcal{U}$. Далее мы будем убирать графы из этого множества и в итоге получим конечное множество графов, для которых достаточно проверить гипотезы.

Лемма 30. *Пусть $\mathcal{L}$ — хорошее множество для Deletion (Completion, Editing) задач, и пусть $H \in \mathcal{L}$ такой, что для задачи H-free Edge Deletion (Completion, Editing) не существует $poly(OPT)$-приближения при $P \neq NP$. Тогда $\mathcal{L} \setminus \{H\}$ — хорошее множество для Deletion (Completion, Editing) задач.*

Доказательство. Пусть это не так и множество $\mathcal{L} \setminus \{H\}$ не является хорошим для Deletion (Completion, Editing) задач. Значит, $P \neq NP$ и нашелся граф $G \notin \mathcal{Y}$ с хотя бы пятью вершинами такой, что для задачи G-free Edge Deletion (Completion, Editing) есть $poly(OPT)$-приближение, при этом для всех графов из $\mathcal{L} \setminus \{H\}$ с количеством вершин, не превосходящим $|V(G)|$, $poly(OPT)$-приближения для соответствующих им задач нет. Тогда, так как $\mathcal{L}$ хорошее для Deletion (Completion, Editing) задач, для задачи H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, но по условию это не так. □

Лемма 31. *Пусть $\mathcal{L}$ — хорошее множество для Deletion (Completion, Editing) задач. Пусть $H \in \mathcal{L}$, а H' — такой граф, что если для H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, то и для H'-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение. При этом $H' \notin \mathcal{Y}$ и $|V(H')| < |V(H)|$. Тогда множество $\mathcal{L} \setminus \{H\}$ — хорошее для задач H-free Edge Deletion (Completion, Editing).*

Доказательство. Пусть это не так и множество $\mathcal{L} \setminus \{H\}$ не является хорошим для Deletion (Completion, Editing) задач. Значит, $P \neq NP$ и нашелся граф $G \notin \mathcal{Y}$ с хотя бы пятью вершинами такой, что для задачи G-free Edge Deletion (Completion, Editing) есть $poly(OPT)$-приближение, при этом для всех графов из $\mathcal{L} \setminus \{H\}$ с количеством вершин, не превосходящим $|V(G)|$, $poly(OPT)$-приближения для соответствующих им задач нет. Тогда так как $\mathcal{L}$ — хорошее для Deletion (Completion, Editing) задач, то $V(H) \leq V(G)$ и для задачи H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, а тогда и для задачи H'-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение. Тогда H' не равен C_4 или $\overline{C_4}$, поскольку по леммам 18 и 20 для задач C_4-free Edge Deletion (Completion, Editing) и $\overline{C_4}$-free Edge Deletion (Completion, Editing) не существует $poly(OPT)$-приближения при $P \neq NP$. Тогда в H' хотя бы пять вершин. Тогда должен существовать граф $H'' \in \mathcal{L}$ такой, что $|V(H'')| \leq |V(H')| < |V(H)|$ и для задачи H''-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение. Так как в H'' строго меньше вершин, чем в H, то $H'' \in \mathcal{L} \setminus \{H\}$. Противоречие, так как мы предполагали, что для всех графов из $\mathcal{L} \setminus \{H\}$ с количеством вершин, не превосходящим $|V(G)|$, у соответствующих им задач нет $poly(OPT)$-приближения. □

Назовем граф H *неинтересным для Deletion (Completion, Editing) задач*, если либо для задачи H-free Edge Deletion (Completion, Editing) не существует $poly(OPT)$-приближения при $P \neq NP$, либо существует граф $H' \notin \mathcal{Y}$ такой, что $|V(H')| < |V(H)|$ и из существования $poly(OPT)$-приближения для задачи H-free Edge Deletion (Completion, Editing) следует существование $poly(OPT)$-приближения для задачи H'-free Edge Deletion (Completion, Editing). Назовем граф H *неинтересным*, если он является неинтересным для Deletion, Completion и Editing задач одновременно. И лемм 30 и 31 получаем следующее следствие.

Следствие 10. *Пусть $\mathcal{L}$ — хорошее множество, а H — неинтересный граф. Тогда множество $\mathcal{L} \setminus \{H\}$ — хорошее.*

Таким образом, мы показали, что неинтересные графы всегда можно убрать из хорошего множества, оставив это множество хорошим. Далее мы будем показывать для разных графов из $\mathcal{W} \cup \mathcal{U}$, что они являются неинтересными, и таким образом получим хорошее множество, состоящее из конечного числа графов. Отметим также следующую лемму.

Лемма 32. *H — неинтересный $\Leftrightarrow$ $\overline{H}$ — неинтересный.*

Доказательство. Покажем, что если H — неинтересный для Deletion (Completion, Editing) задач, то $\overline{H}$ — неинтересный для Completion (Deletion, Editing) задач. Тогда мы получим, что $\overline{H}$ — неинтересный. Пусть задача H-free Edge Deletion (Completion, Editing) не допускает $poly(OPT)$-приближения при $P \neq NP$. Тогда по лемме 3 задача $\overline{H}$-free Edge Completion (Deletion, Editing) тоже не допускает $poly(OPT)$-приближения при $P \neq NP$. Пусть существует такой граф $H' \notin \mathcal{Y}$, что $|V(H')| < |V(H)|$ и из существования $poly(OPT)$-приближения для задачи H-free Edge Deletion (Completion, Editing) следует существование $poly(OPT)$-приближения для задачи H'-free Edge Deletion (Completion, Editing). Если задача $\overline{H}$-free Edge Completion (Deletion, Editing) допускает $poly(OPT)$-приближение, то по лемме 3 задача H-free Edge Deletion (Completion, Editing) тоже допускает $poly(OPT)$-приближение, а тогда и H'-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение, и снова по лемме 3 задача $\overline{H'}$-free Edge Completion (Deletion, Editing) допускает $poly(OPT)$-приближение. При этом $\overline{H'} \notin \mathcal{Y}$ и $|V(\overline{H'})| < |V(\overline{H})|$. Таким образом, $\overline{H}$ — неинтересный. □

6 Избавляемся от графов из множества $\mathcal{U}$

В этом разделе мы покажем, что все графы из множества $\mathcal{U}$ являются неинтересными и их можно убрать из рассмотрения. Нам понадобятся некоторые утверждения для графов вида $K_n - e$ с *галочками*, то есть с вершинами степени два, оба соседа которых лежат в $K_n - e$. Назовем *вершиной* галочки вершину степени два, соседи которой лежат в $K_n - e$, а *основанием* галочки — соседей вершины этой галочки.

Лемма 33. *Пусть H — это $K_n - e$ хотя бы с тремя галочками для некоторого $n \geq 5$. Пусть при этом существует пара галочек с непересекающимися основаниями. Тогда задачи Sandwich H-free Edge Deletion и Sandwich H-free Edge Completion — NP-трудные.*

Доказательство. Рассмотрим две галочки A и B с непересекающимися основаниями и еще одну произвольную галочку C. Определим e_1 как антиребро между вершинами галочек A и B. Если основание C совпадает с основанием A, то определим e_2 как антиребро между вершинами галочек B и C. Иначе e_2 — это антиребро между вершинами галочек A и C. Будем использовать введенные антиребра e_1 и e_2 для построения Variable, Clause и Connector гаджетов для задач Sandwich H-free Edge Deletion и Sandwich H-free Edge Completion.

Рассмотрим произвольный 2-разделитель (u, v) графа H. Заметим, что он является основанием галочки, а тогда по одну сторону от него в H лежит галочка, а по другую — граф $K_n - e$. Предположим, что H пересекает гаджеты. Тогда найдется граф G, полученный применением конструкции 2 к некоторой выполнимой формуле φ, выполняющему набору τ для φ и {Variable, Clause, Connector}, множество вершин $V' \subseteq V(G)$ такое, что индуцированный подграф $G[V']$ изоморфен H, а также ребро или антиребро q графа G такое, что (i) q

принадлежит хотя бы двум гаджетам, а следовательно, соответствует в них разрешенным ребрам (антиребрам); (ii) q является 2-разделителем в $G[V']$, соответствующим некоторому 2-разделителю (u, v) в H. Так как $K_n - e$ — трехсвязный, то он целиком лежит в одном каком-то гаджете, при этом одно из его ребер (антиребер) попадает в q, то есть является разрешенным. Поэтому будем строить гаджеты так, чтобы это не выполнялось.

Для начала построим гаджеты для задачи Sandwich H-free Edge Deletion. В качестве Variable гаджета возьмем копию H с добавленными e_1 и e_2, при этом ребра e_1 и e_2 будут разрешенными, а остальные — запрещенными. Покажем, что вне зависимости от того, какое из разрешенных ребер будет удалено из данного гаджета, в полученном графе не найдется индуцированного подграфа $K_n - e$, содержащего обе вершины какого-либо разрешенного ребра. Для этого заметим, что у концов e_1 нет общих соседей, а у концов e_2 не более одного общего соседа, но при этом у любой пары вершин в $K_n - e$ есть хотя бы два общих соседа. В качестве Clause гаджета возьмем копию H, выберем любые три ребра на галочках и сделаем их разрешенными, а остальные — запрещенными. Заметим, что для каждого разрешенного ребра степень одного из его концов не превосходит двух при любом способе удаления разрешенных ребер, а в $K_n - e$ у любой вершины степень хотя бы три. В качестве Connector гаджета возьмем копию H, добавим в нее ребро e_1 и сделаем e_1 разрешенным. Также сделаем разрешенным еще одно ребро из галочки C. Остальные ребра будут запрещенными. По тем же причинам, что и для Variable и Clause гаджетов, в Connector гаджете не найдется индуцированного $K_n - e$, содержащего обе вершины какого-либо разрешенного ребра при любом способе удаления разрешенных ребер. Таким образом, мы построили такие гаджеты для задачи Sandwich H-free Edge Deletion, что H не пересекает эти гаджеты. Тогда по лемме 14 задача Sandwich H-free Edge Deletion — NP-трудная.

Теперь построим гаджеты для задачи Sandwich H-free Edge Completion. В качестве Variable гаджета возьмем копию H, удалим в ней два произвольных ребра из галочек и сделаем получившиеся антиребра разрешенными, а остальные — запрещенными. Clause гаджет будет состоять из двух частей C_1 и C_2. C_1 — это копия H, в которой антиребра e_1 и e_2 являются разрешенными, а остальные — запрещенными. C_2 — это копия C_1, в которой также удалено и разрешено произвольное ребро какой-то галочки. В качестве Connector гаджета мы возьмем копию H с одним удаленными ребром e_{in}, лежащем в галочке C, и антиребром $e_{out} = e_1$. Антиребра e_{in} и e_{out} будут разрешенными, остальные — запрещенными. Заметим, что для каждого построенного гаджета и для каждого разрешенного антиребра e' этого гаджета либо при любом способе добавления разрешенных антиребер в гаджете найдется не более одного общего соседа для концов e', либо при любом способе добавления разрешенных антиребер один из концов e' будет иметь степень не более двух в этом гаджете. Выше мы показали, что в этом случае в гаджете не может появиться индуцированного $K_n - e$, содержащего оба конца e'. Таким образом, мы построили такие гаджеты для задачи Sandwich H-free Edge Completion, что H не пересекает эти гаджеты. Тогда по лемме 14 задача Sandwich H-free Edge Completion — NP-трудная. □

Лемма 34. *Пусть H — это $K_n - e$ с галочками для некоторого $n \geq 5$. Пусть при этом существует пара галочек с непересекающимися основаниями. Тогда существует такой Deletion Enforcer X, что H не пересекает X.*

Доказательство. Рассмотрим произвольный 2-разделитель для H. По обе стороны от разделителя окажется вершина, смежная с обеими вершинами этого разделителя. Тогда, если H пересекает Deletion Enforcer, то в этом Deletion Enforcer'е должна найтись вершина, смежная с обоими концами выделенного ребра. Рассмотрим Deletion Enforcer X — копию H с добавленным выделенным ребром, соединяющим вершины галочек с непересекающимися основаниями. У концов выделенного ребра в X нет общих соседей. Таким образом, H не пересекает X. $\square$

Лемма 35. *Пусть H — это $K_n - e$ с галочками для некоторого $n \geq 5$. Тогда существует такой Completion Enforcer X, что H не пересекает X.*

Доказательство. Заметим, что любой 2-разделитель H — это основание какой-то галочки. Тогда, если в H у всех галочек основание — это ребро, то нам подойдет любой Completion Enforcer. Пусть есть галочка A, основание которой — антиребро. Для H верно, что для любого 2-разделителя по обе стороны от разделителя найдется вершина, соединенная с обеими вершинами этого разделителя. Тогда если H пересекает Completion Enforcer, то в этом Completion Enforcer'е должна найтись вершина, смежная с обоими концами выделенного антиребра. В качестве Completion Enforcer'а X возьмем копию H и удалим в нем одно из ребер галочки A. Тогда у концов выделенного антиребра в X нет общих соседей. Таким образом, H не пересекает X. $\square$

Пользуясь леммами 15 и 16, получаем следующие следствия.

Следствие 11. *Пусть H — это $K_n - e$ с галочками для некоторого $n \geq 5$. Пусть при этом существует пара галочек с непересекающимися основаниями. Тогда если задача Sandwich H-free Edge Deletion (Completion) NP-трудная, то для задачи H-free Edge Deletion (Completion) не существует $poly(OPT)$-приближения при $P \neq NP$.*

Следствие 12. *Пусть H — это $K_n - e$ с галочками для некоторого $n \geq 5$. Тогда если для задачи H-free Edge Deletion не существует $poly(OPT)$-приближения, то и для задачи H-free Edge Editing не существует $poly(OPT)$-приближения.*

Теперь мы готовы перейти к доказательству того, что графы из множества $\mathcal{U}$ — неинтересные. Для графа $H \in \mathcal{U}$ выполняется $H - V_1 = K_n - e$ и $H - V_2 = \overline{K_m - q}$, где $n, m \geq 5$. Рассмотрим несколько случаев в зависимости от расположения ребра q и антиребра e в H.

Лемма 36. *Пусть $H \in \mathcal{U}$ и $q \in H[V_\ell]$. Тогда H — неинтересный.*

Доказательство. В $\overline{K_m - q}$ хотя бы три изолированных вершины, а значит, в $H - V(q)$ хотя бы три антиребра, поэтому $H - V(q)$ не полный и не равен $K_t - e$ ни для какого t. Также $H - V(q)$ содержит больше одного ребра и хотя бы пять вершин, так как содержит $K_n - e$ в качестве индуцированного подграфа. Значит, $H - V(q) \notin \mathcal{Y}$. Введем $H' = H - V(q)$. Покажем, что если есть $poly(OPT)$-приближение для задачи H-free Edge Deletion (Completion, Editing), то существует $poly(OPT)$-приближение и для задачи H'-free Edge Deletion (Completion, Editing). Тогда мы получим, что граф H является неинтересным.

Пусть задача H–free Edge Deletion (Completion, Editing) допускает $q(OPT)$-приближение для некоторого полинома q. Введем полином $p(x) = q(x) \cdot x$. Пусть дан инстанс (G', k) задачи H'-free Edge Deletion (Completion, Editing). Применим конструкцию 5 к $(G', p(k), H, V(H'))$. Получим граф G. Пусть F' — решение для (G', k). Покажем, что F' также является решением для инстанса (G, k) задачи H-free Edge Deletion (Completion, Editing). Предположим, что это не так. Тогда в $G \triangle F'$ найдется индуцированный H. Так как F' — решение для (G', k), то H не может целиком лежать в $G' \triangle F'$. Значит, в H содержатся вершины из $V(G) \setminus V(G')$. Рассмотрим вершину $\widetilde{u} \in V(G) \setminus V(G')$. По построению существует ребро $\widetilde{q} = (\widetilde{u}, \widetilde{v})$ такое, что $\widetilde{v} \in V(G) \setminus V(G')$. Степень $\widetilde{u}$ и $\widetilde{v}$ в $G \triangle F'$ равна ℓ, так как в F' лежат только ребра из G'. Это значит, что если $\widetilde{u}$ входит в H, то $\widetilde{v}$ тоже входит в H, поскольку иначе максимальная возможная степень $\widetilde{u}$ в H была бы $\ell - 1$. Но тогда мы нашли две вершины степени ℓ в H, соединенные ребром. Это может быть только ребро q в H. Таким образом, в вершины $V(G) \setminus V(G')$ может попасть только конец ребра q, а тогда в $G' \triangle F'$ содержится H'. Получили противоречие с тем, что F' — решение для (G', k). Значит, в $G \triangle F'$ нет индуцированного H, и F' — решение для (G, k).

По лемме 22, если F — это решение для инстанса $(G, p(k))$ задачи H-free Edge Deletion (Completion, Editing), то $F \cap G'$ — решение для инстанса $(G', p(k))$ задачи H'-free Edge Deletion (Completion, Editing). Тогда мы можем воспользоваться леммой 4 и получить, что если для H-free Edge Deletion (Completion, Editing) есть $poly(OPT)$-приближение, то и для H'-free Edge Deletion (Completion, Editing) есть $poly(OPT)$-приближение. □

Лемма 37. *Пусть $H \in \mathcal{U}$ и $q \in H[V_h]$. Тогда H — неинтересный.*

Доказательство. В этом случае $K_n - e = H[V_\ell \cup V_m]$. Каждая вершина $K_n - e$ имеет степень хотя бы $n-2$. Тогда $\ell \geq n-2 \geq 3$, а $h \geq l+1 \geq 4$. Таким образом, H состоит из графа $K_n - e$ и еще $|V_h|$ вершин, каждая из которых соединена с хотя бы $h-1 \geq 3$ вершинами из $K_n - e$. Тогда H — трехсвязный с хотя бы двумя антиребрами, то есть H-free Edge Deletion (Completion, Editing) не допускает $poly(OPT)$-приближения при $P \neq NP$, а тогда H — неинтересный. □

Применяя лемму 32, получаем следующие следствия.

Следствие 13. *Пусть $H \in \mathcal{U}$ и $e \in H[V_h]$. Тогда H — неинтересный.*

Следствие 14. *Пусть $H \in \mathcal{U}$ и $e \in H[V_\ell]$. Тогда H — неинтересный.*

Таким образом, мы получаем, что графы $H \in \mathcal{U}$, для которых оба конца ребра q лежат в V_1 или оба конца антиребра e лежат в V_2, являются неинтересными. Далее будем рассматривать только такие $H \in \mathcal{U}$, для которых q и e содержат хотя бы по одному концу в V_m. Заметим, что $|V_m| \leq 2$, так как $H[V_m]$ является пересечением графов $K_n - e$ и $\overline{K_m - q}$. Значит, q и e не могут одновременно целиком находиться внутри $H[V_m]$.

Лемма 38. *Пусть $H \in \mathcal{U}$. Пусть один конец ребра q лежит в V_m, а второй — в V_1. Тогда H — неинтересный.*

Доказательство. Пусть $q = (v_1, v_m)$, где $v_1 \in V_1$, $v_m \in V_m$. Пусть d — это степень вершин из V_1 ($d = \ell$ или $d = h$). Так как $v_1 \in V_1$ и v_1 инцидентна ребру q, то $d \geq 1$. Если $d = 1$, то H — это $K_n - e$, в котором к некоторым вершинам подвешены вершины степени один, или *листья*. В этом случае V_1 — это V_ℓ, а V_2 — это V_h. Заметим, что степень вершины v_m хотя бы $n-2+1 = n-1$, так как ее степень внутри $K_n - e$ хотя бы $n-2$, и из нее идет ребро q в V_1. Но тогда, чтобы у вершины v_m степень была меньше h, из каждой вершины V_2 должен идти хотя бы один лист. Так как $n \geq 5$, а $|V_m| \leq 2$, то $|V_2| \geq 3$. Тогда в индуцированном $K_n - e$ есть хотя бы четыре вершины, а именно $V_2 \cup \{v_m\}$, из которых идут листья. Таким образом, в H хотя бы четыре листа. Рассмотрим граф $\overline{H}$. Он содержит $K_{\geq 4}$, индуцированный листьями H, а из всех остальных вершин есть хотя бы по три ребра в $K_{\geq 4}$. Получаем, что $\overline{H}$ — трехсвязный с хотя бы двумя антиребрами. Если $d \geq 3$, то H — это $K_n - e$ и еще несколько вершин, каждая из которых соединена хотя бы с тремя вершинами из $K_n - e$. Тогда H — трехсвязный с хотя бы двумя антиребрами. Таким образом, если $d \neq 2$, то по леммам 17 и 19 для задачи H-free

Edge Deletion (Completion, Editing) не существует $poly(OPT)$-приближения, если $P \neq NP$, а тогда H — неинтересный.

Далее мы будем рассматривать только такие H, для которых $d = 2 = \ell$. В этом случае, так как $m \geq 5$, а $|V_m| \leq 2$, то $|V_1| \geq 3$ и H — это $K_n - e$ с хотя бы тремя галочками. Так как степень вершины v_m хотя бы $n-1$, то $h \geq n$, а тогда каждая вершина V_2 должна быть покрыта хотя бы одной галочкой. Пусть основания любой пары галочек пересекаются. Тогда они все пересекаются с основанием галочки v_1. Так как один лист этой галочки лежит в V_m, то для остальных галочек он недоступен, так как внутри $\overline{K_m - q}$ нет ребер, кроме q. Тогда основания всех остальных галочек содержат второй лист галочки v_1. Назовем эту вершину u. Так как в V_2 хотя бы три вершины, то u покрывает хотя бы на две галочки больше, чем остальные вершины V_2. При этом степень u в $K_n - e$ может быть меньше степеней остальных вершин V_2 в $K_n - e$ максимум на 1. Это значит, что степень u в H не может равняться h. Получили противоречие, значит, найдутся две галочки с непересекающимися основаниями.

Мы хотим показать, что задача H-free Edge Deletion (Completion, Editing) не допускает $poly(OPT)$-приближения, если $P \neq NP$. Так как в H хотя бы три галочки и при этом есть две галочки с непересекающимися основаниями, то по лемме 33 задачи Sandwich H-free Edge Deletion и Sandwich H-free Edge Completion — NP-трудные. По следствию 11 получаем, что задачи H-free Edge Deletion и H-free Edge Completion не допускают $poly(OPT)$-приближения при $P \neq NP$. По следствию 12 получаем, что для задачи H-free Edge Editing тоже не существует $poly(OPT)$-приближения при $P \neq NP$. Таким образом, H — неинтересный. □

Пользуясь леммой 32, получаем следующее следствие.

Следствие 15. *Пусть $H \in \mathcal{U}$. Пусть один конец антиребра e лежит в V_m, а второй — в V_2. Тогда H — неинтересный.*

Мы разобрали все случаи расположения ребра q и антиребра e в $H \in \mathcal{U}$ и показали, что все графы из множества $\mathcal{U}$ являются неинтересными. Тогда, пользуясь следствием 10, мы получаем следующую лемму.

Лемма 39. *$\mathcal{W}$ — хорошее множество.*

7 Избавляемся от графов из множества $\mathcal{F} \cup \overline{\mathcal{F}} \cup \mathcal{S} \cup \overline{\mathcal{S}}$

В этом разделе мы покажем, что все графы из множества $\mathcal{F} \cup \overline{\mathcal{F}} \cup \mathcal{S} \cup \overline{\mathcal{S}}$ являются неинтересными. Доказательства будут во многом повторять рассуждения из статьи [14], в которой задачи H-free Edge Deletion (Completion, Editing) рассматривались с точки зрения существования полиномиальных ядер.

Лемма 40. *Пусть $H = J \cup K_t$, где J — это некоторый граф, а $t \geq 1$, при этом $V_\ell = V(K_t)$. Пусть $H' = H - v$, где v — это произвольная вершина K_t. Тогда если для задачи H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, то для задачи H'-free Edge Deletion (Completion, Editing) тоже существует $poly(OPT)$-приближение.*

Доказательство. Пусть задача H-free Edge Deletion (Completion, Editing) допускает $q(OPT)$-приближение для некоторого полинома q. Введем полином $p(x) = q(x) \cdot x$. Пусть дан инстанс (G', k) задачи H'-free Edge Deletion (Completion, Editing). Применим конструкцию 5 к $(G', p(k), H, V(H'))$, получим граф G. Пусть F' — решение для (G', k). Покажем, что F' также является решением для инстанса (G, k) задачи H-free Edge Deletion (Completion, Editing).

Предположим, что это не так, тогда в $G \triangle F'$ найдется индуцированный H. Так как F' — решение для (G', k), то H не может целиком лежать в $G' \triangle F'$. Значит, в H содержатся вершины из $V(G) \setminus V(G')$. Рассмотрим вершину $u \in V(G) \setminus V(G')$. Ребра из u не могут оказаться в F', поэтому ее степень в $G \triangle F'$ равна ℓ, а тогда ее степень в индуцированном H не превосходит ℓ. Это значит, что в $V(G) \setminus V(G')$ могли попасть только вершины из K_t. Заметим, что $V(G) \setminus V(G')$ — изолированное множество, а тогда не более одной вершины из K_t могло попасть в $V(G) \setminus V(G')$. Но в этом случае оставшийся граф H' лежит в $G' \triangle F'$. Получили противоречие с тем, что F' — решение для (G', k). Таким образом, F' — решение для (G, k). По лемме 22 если F — решение для инстанса $(G, p(k))$ задачи H-free Edge Deletion (Completion, Editing), то $F \cap G'$ — решение для инстанса $(G', p(k))$ задачи H'-free Edge Deletion (Completion, Editing). Тогда по лемме 4, если задача H-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение, то и задача H'-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение. $\square$

Следствие 16. *Пусть $H \in \overline{\mathcal{F}_1} \cup \overline{\mathcal{F}_6}$. Тогда H — неинтересный.*

Лемма 41. *Пусть H — это граф без изолированных вершин и вершин степени один, а $p \geq 2$ такое, что в H существует единственный подграф P (не обязательно индуцированный), являющийся путем длины p, все внутренние вершины которого в H имеют степень ровно два. Пусть H' получается из H удалением всех внутренних вершин пути P. Тогда если для задачи H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, то для задачи H'-free Edge Deletion (Completion, Editing) тоже существует $poly(OPT)$-приближение.*

Доказательство. Пусть задача H-free Edge Deletion (Completion, Editing) допускает $q(OPT)$-приближение для некоторого полинома q. Введем полином $p(x) = q(x) \cdot x$. Пусть (G', k) — инстанс задачи H'-free Edge Deletion (Completion, Editing). Применим конструкцию 5 к $(G', p(k), H, V(H'))$, получим граф G. Пусть F' — решение для (G', k). Покажем, что F' также является решением для (G, k). Предположим, что это не так, тогда в $G \triangle F'$ нашелся H. Пусть какая-то вершина H попала в $V(G) \setminus V(G')$. Так как ребра из этой вершины не могут быть в F', а ее степень равна минимальной степени в H, то вместе с ней в H попали и все ее соседи. Применим это рассуждение к ее соседям и так далее. Получаем, что вместе с этой вершиной в H лежит целый путь длины p, все внутренние вершины которого лежат в надстройке, а значит, имеют степень два в H. Таким образом, любая вершина из H, попавшая в надстройку, лежит в P. Тогда H' целиком лежит в $G' \triangle F'$, противоречие.

Обратно, пусть F — решение для инстанса $(G, p(k))$ задачи H-free Edge Deletion (Completion, Editing). Тогда по лемме 22 $F \cap G'$ является решением для инстанса $(G', p(k))$ задачи H'-free Edge Deletion (Completion, Editing). Тогда по лемме 4 мы получаем, что из существования $poly(OPT)$-приближения для H-free Edge Deletion (Completion, Editing) следует существование $poly(OPT)$-приближения для H'-free Edge Deletion (Completion, Editing). $\square$

Следствие 17. *Пусть $H \in \mathcal{F}_9 \cup \mathcal{F}_{10}$. Тогда H — неинтересный.*

Следствие 18. *Графы S_5, S_9, S_{15}, S_{22} — неинтересные.*

Лемма 42. *Пусть H — связный, но не двусвязный граф с хотя бы двумя вершинами. Пусть C — множество всех двусвязных компонент H, содержащих ровно одну точку сочленения H. Пусть J — двусвязная компонента с наименьшим количеством вершин среди компонент из C, и пусть такая J единственная. Пусть $v \in V(J)$ — точка сочленения H. Пусть $H' =$*

$H-(V(J)\setminus\{v\})$. *Тогда если для задачи H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, то для задачи H'-free Edge Deletion (Completion, Editing) тоже существует $poly(OPT)$-приближение.*

Доказательство. Пусть задача H-free Edge Deletion (Completion, Editing) допускает $q(OPT)$-приближение для некоторого полинома q. Введем полином $p(x) = q(x) \cdot x$. Пусть дан инстанс (G', k) задачи H'-free Edge Deletion (Completion, Editing). Применим конструкцию 5 к $(G', p(k), H, V(H'))$, получим граф G. Пусть F' — решение для (G', k). Покажем, что F' также является решением для инстанса (G, k) задачи H-free Edge Deletion (Completion, Editing). Предположим, что это не так, тогда в $G \triangle F'$ найдется индуцированный H. Так как F' — решение для (G', k), то H не может целиком лежать в $G' \triangle F'$. Значит, в H содержится вершина из некоторой надстройки V_i. Существует ровно одна вершина $v \in V(G')$, смежная с этой надстройкой. Заметим, что H не может целиком лежать в $G[\{v\} \cup V_i]$, поскольку в $\{v\} \cup V_i$ меньше вершин, чем в H. Значит, H пересекает $\{v\} \cup V_i$. Но так как ребра, инцидентные вершинам из V_i, не входят в F', то вершины H, попавшие в V_i, отделяются от остальных вершин H точкой сочленения. Значит, в $\{v\} \cup V_i$ могли попасть только компоненты двусвязности H, размер которых не превосходит $|\{v\} \cup V_i|$, а такая компонента двусвязности в H единственная, и это J. Но тогда в $G' \triangle F'$ найдется H', противоречие. Таким образом, F' — решение для (G, k).

Обратно, пусть F — решение для инстанса $(G, p(k))$ задачи H-free Edge Deletion (Completion, Editing). Тогда по лемме 22 $F \cap G'$ является решением для инстанса $(G', p(k))$ задачи H'-free Edge Deletion (Completion, Editing). Тогда по лемме 4 мы получаем, что из существования $poly(OPT)$-приближения для H-free Edge Deletion (Completion, Editing) следует существование $poly(OPT)$-приближения для H'-free Edge Deletion (Completion, Editing). $\square$

Следствие 19. *Графы S_{20} и S_{27} — неинтересные.*

Лемма 43. *Граф S_{35} — неинтересный.*

Доказательство. Пусть $H = S_{35}$, а H' — это H с удаленными двумя смежными вершинами степени три. Заметим, что $H' \notin \mathcal{Y}$. Пусть задача H-free Edge Deletion (Completion, Editing) допускает $q(OPT)$-приближение для некоторого полинома q. Введем полином $p(x) = q(x) \cdot x$. Пусть дан инстанс (G', k) задачи H'-free Edge Deletion (Completion, Editing). Применим конструкцию 5 к $(G', p(k), H, V(H'))$, получим граф G. Пусть F' — решение для (G', k). Покажем, что F' также является решением для инстанса (G, k) задачи H-free Edge Deletion (Completion, Editing). Предположим, что это не так, тогда в $G \triangle F'$ найдется индуцированный H. Так как F' — решение для (G', k), то H не может целиком лежать в $G' \triangle F'$. Значит, какая-то вершина H попала в $V(G) \setminus V(G')$. Назовем эту вершину u. Так как ребра из u не содержатся в F', а степень u равна минимальной степени в H, то вместе с u в H лежат все ее соседи. Тогда H содержит всю надстройку, в которую попала u. Но тогда u является концом ребра в H, соединяющего две вершины степени три, а такое ребро единственное, и концы этого ребра образуют $V(H) \setminus V(H')$. Тогда в $G' \triangle F'$ содержится H'. Получили противоречие с тем, что F' — решение для (G', k). Значит, в $G \triangle F'$ нет индуцированного H, и F' — решение для (G, k). По лемме 22, если F — это решение для инстанса $(G, p(k))$ задачи H-free Edge Deletion (Completion, Editing), то $F \cap G'$ — решение для инстанса $(G', p(k))$ задачи H'-free Edge Deletion (Completion, Editing). Тогда мы можем воспользоваться леммой 4 и получить, что если для H-free Edge Deletion (Completion, Editing) есть $poly(OPT)$-приближение, то и для H'-free Edge Deletion (Completion, Editing) есть $poly(OPT)$-приближение. А тогда граф H — неинтересный. $\square$

Назовем $\mathcal{M}$ *модульным разложением* графа G, если $\mathcal{M}$ — это разбиение вершин G на максимальные множества, или *модули*, такие, что для любого множества $M \in \mathcal{M}$ у всех вершин из M одинаковая окрестность вне M. Пусть $\mathcal{M}' \subseteq \mathcal{M}$, а $V' = \bigcup_{M \in \mathcal{M}'} M$. Тогда будем говорить, что $\mathcal{M}'$ соответствует V'.

Также нам понадобится следующая конструкция.

Конструкция 6. *Даны (G', p, ℓ), где G' — граф, и $p, \ell \geq 1$. Для каждого подмножества S из ℓ вершин G' введем клику C_S на $2p+1$ вершине и проведем ребра (c_i, s_j) между всеми парами вершин $c_i \in C_S$, $s_j \in S$. Полученный граф назовем G.*

Лемма 44. *Пусть H — нерегулярный граф, для которого выполняются следующие свойства:*

1. *$1 \leq \ell \leq 2$, $|V(H)| \geq 5$;*
2. *V_ℓ — независимое множество, $H[V_h \cup V_m]$ — связный граф, каждая вершина из V_h смежна хотя бы с одной вершиной из V_ℓ;*
3. *Каждая вершина из V_m соединена хотя бы с $\ell+1$ вершиной вне V_m, или не существует такой пары соседних вершин u, v в V_m, что $N[u] = N[v]$.*

Пусть $\mathcal{M}$ — модульное разложение графа H, и пусть $\mathcal{M}' \subseteq \mathcal{M}$ соответствует V_ℓ. Пусть H' — граф, полученный из H удалением одной вершины из каждого модуля $\mathcal{M}'$. Тогда если для задачи H-free Edge Deletion (Completion, Editing) существует poly(OPT)-приближение, то для задачи H'-free Edge Deletion (Completion, Editing) тоже существует poly(OPT)-приближение.

Доказательство. Пусть задача H-free Edge Deletion (Completion, Editing) допускает $q(OPT)$-приближение для некоторого полинома q. Введем полином $p(x) = q(x) \cdot x$. Пусть дан инстанс (G', k) задачи H'-free Edge Deletion (Completion, Editing). Применим конструкцию 6 к $(G', p(k), \ell)$, получим граф G. Пусть F' — решение для (G', k). Покажем, что F' также является решением для инстанса (G, k) задачи H-free Edge Deletion (Completion, Editing). Предположим, что это не так, тогда в $G \triangle F'$ нашелся индуцированный H. Предположим, что $(V_h \cup V_m) \cap (V(G) \setminus V(G')) = \emptyset$. Тогда в клики на вершинах из $V(G) \setminus V(G')$ попали только вершины из V_ℓ. Заметим, что в каждую такую клику могло попасть не более одной вершины из V_ℓ, так как V_ℓ — это независимое множество. Также вершины из одного модуля $\mathcal{M}'$ не могли попасть в разные клики на вершинах из $V(G) \setminus V(G')$, поскольку тогда бы у них были разные окрестности в H. Тогда в $G' \triangle F'$ найдется индуцированный H', что противоречит тому, что F' — решение для (G', k). Значит, $|(V_h \cup V_m) \cap (V(G) \setminus V(G'))| \geq 1$.

Пусть $|V_h \cap (V(G) \setminus V(G'))| \geq 1$ и вершина $v \in V_h$ попала в некоторую надстройку C, введенную для подмножества S из ℓ вершин графа G'. Так как у всех вершин из $V(H) \cap C$ одинаковая степень в H, то $V(H) \cap C \subseteq V_h$. Пусть $\ell = 1$, тогда $S = \{u\}$ для некоторой вершины $u \in V(G')$. Так как любая вершина из V_h по условию должна быть смежна хотя бы с одной вершиной из V_ℓ в H, то степень u в H равна ℓ. Заметим, что $N[v] \subseteq N[u]$, а значит, степень u в H не меньше степени v в H. Получили противоречие с тем, что $\ell < h$. Значит, $\ell = 2$, а $h \geq 3$. Тогда $S = \{u_1, u_2\}$ для некоторых вершин $u_1, u_2 \in V(G')$. Тогда v смежна с не более чем двумя вершинами из V_ℓ. Если v смежна с ровно двумя вершинами из V_ℓ, то это вершины u_1 и u_2. Но тогда так как $H[V_h \cup V_m]$ — связный, то $V_m = \emptyset$, а $V_h = V(H) \cap C$. Так как все вершины $V(H) \cap C$ имеют одинаковую замкнутую окрестность, то все вершины из

V_h смежны с u_1 и u_2. Тогда $H = K_2 \boxtimes 2K_1$. Но в таком графе всего четыре вершины, поэтому получаем противоречие с условием. Значит, v смежна с ровно одной вершиной из V_ℓ, пусть это вершина u_1. Если $u_2 \notin V_h \cup V_m$, то $V_h = V(H) \cap C$ и снова $u_1 \in V_\ell$ будет иметь степень в H не меньше, чем $v \in V_h$. Получаем противоречие с тем, что $\ell < h$. Значит, $u_2 \in V_h \cup V_m$. Если $|V_h \cap C| \geq 3$, то степень u_1 в H хотя бы 3, и мы получаем противоречие с тем, что $\ell = 2$. Значит, $|V_h \cap C| \leq 2$. Так как $h \geq 3$, то $|V_h \cap C| = 2$ и $h = 3$. Раз $\ell = 2$ и $h = 3$, то $V_m = \emptyset$. Значит, $u_2 \in V_h$. Вершины u_1 и u_2 несмежны в $G \triangle F'$, так как иначе степень u_1 была бы больше двух. Тогда у u_2 должен быть сосед $u_3 \in V_\ell$, отличный от u_1. Других соседей, помимо u_3 и двух вершин из $V_h \cap C$, у u_2 нет. Но тогда, так как $H[V_h \cup V_m]$ — связный, $V(H) = (V(H) \cap C) \cup \{u_1, u_2, u_3\}$, и у u_3 в H всего один сосед. Получили противоречие с тем, что $\ell = 2$. Значит, $|V_h \cap (V(G) \setminus V(G'))| = 0$.

Тогда $|V_m \cap (V(G) \setminus V(G'))| \geq 1$. Пусть вершина $v \in V_m$ попала в некоторую надстройку C. Так как вершины в $V(H) \cap C$ имеют одинаковую степень в индуцированном H, то $V(H) \cap C \subseteq V_m$. По пункту 3 условия леммы каждая вершина из V_m соединена хотя бы с $\ell + 1$ вершиной вне V_m или не существует такой пары соседних вершин u, v в V_m, что $N[u] = N[v]$ в H. Заметим, что v может быть соединена с не более чем ℓ вершинами вне V_m, так как вершины C соединены только с вершинами S, где $|S| = \ell$. Значит, для любой пары соседних вершин u, v в V_m $N[u] \neq N[v]$ в H, а тогда $|V_m \cap C| = 1$. Но тогда степень v не превосходит ℓ, противоречие с тем, что $v \in V_m$. Таким образом, в $G \triangle F'$ нет индуцированного H, а значит, F' — решение для (G, k).

Обратно, пусть F — решение для инстанса $(G, p(k))$ задачи H-free Edge Deletion (Completion, Editing). Покажем, что $F \cap G'$ является решением для инстанса $(G', p(k))$ задачи H'-free Edge Deletion (Completion, Editing). Предположим, что это не так, тогда в $G' \triangle (F \cap G')$ найдется индуцированный H'. Рассмотрим множество вершин $V(H) \setminus V(H')$. Для любой пары вершин из этого множества окрестности этих вершин в графе H различаются. Предположим, что нашлись вершины $u, v \in V(H) \setminus V(H')$ такие, что $N(u) \cap V(H') = N(v) \cap V(H')$. Так как V_ℓ — независимое множество, то из этого следует, что $N(u) \cap V(H) = N(u) \cap V(H') = N(v) \cap V(H') = N(v) \cap V(H)$. Получили противоречие, значит, для любой пары вершин из $V(H) \setminus V(H')$ их окрестности в $V(H')$ не совпадают, а также для каждой вершины $u \in V(H) \setminus V(H')$ в $N(u) \cap V(H')$ лежит ровно ℓ вершин. Тогда для каждой вершины $u \in V(H) \setminus V(H')$ рассмотрим надстройку в G, соответствующую $N(u)$. Так как в надстройке $2p(k) + 1$ вершина, а $|F| \leq p(k)$, то найдется такая вершина из надстройки, что ни одно из инцидентных ей ребер не лежит в F. А тогда в $G \triangle F$ найдется индуцированный H. Получили противоречие с тем, что F — решение для $(G, p(k))$. Таким образом, $F \cap G'$ — решение для $(G', p(k))$. А тогда по лемме 4 мы получаем, что из существования $poly(OPT)$-приближения для H-free Edge Deletion (Completion, Editing) следует существование $poly(OPT)$-приближения для H'-free Edge Deletion (Completion, Editing). □

Следствие 20. *Графы* S_7, S_8, S_{10}, S_{11}, S_{12}, S_{13}, $\overline{S_{14}}$, S_{18}, S_{19}, S_{21}, S_{23}, S_{24}, S_{25}, S_{26}, S_{28}, S_{29}, S_{30}, $\overline{S_{32}}$, $\overline{S_{33}}$, $\overline{S_{34}}$, $\overline{S_{36}}$ *— неинтересные.*

Следствие 21. *Пусть* $H \in \mathcal{F}_2 \cup \mathcal{F}_3 \cup \mathcal{F}_4$. *Тогда* H *— неинтересный.*

Лемма 45. *Граф* S_1 *— неинтересный.*

Доказательство. Пусть $H = S_1 = K_{2,3}$. Пусть $H' = C_4 \notin \mathcal{Y}$. Пусть для задачи H-free Edge Deletion (Completion, Editing) существует $q(OPT)$-приближение для некоторого полинома q. Введем полином $p(x) = q(x) \cdot x$. Покажем, что тогда для задачи H'-free Edge Deletion

(Completion, Editing) тоже существует $poly(OPT)$-приближение. Пусть дан инстанс (G', k) задачи H'-free Edge Deletion (Completion, Editing). Применим конструкцию 6 к $(G', p(k), 2)$, получим граф G. Пусть F' — решение для (G', k). Покажем, что F' также является решением для инстанса (G, k) задачи H-free Edge Deletion (Completion, Editing). Предположим, что это не так, тогда в $G \triangle F'$ найдется индуцированный H. Так как F' — решение для (G', k), то H не может целиком лежать в $G' \triangle F'$. Значит, в H попали вершины из некоторой надстройки C_S, являющейся кликой. Назовем множество таких вершин C. Так как $H[C]$ является кликой, и любые две вершины из C имеют одинаковую окрестность в $H - C$, то $|C| = 1$. Каждая вершина из C_S соединена с двумя вершинами G', а значит, степень в H вершины из C не превосходит двух. Так как в H минимальная степень равна двум, то степень в H вершины из C в точности равна двум. Тогда две вершины из S образуют одну из долей H. Так как не существует другой клики $C_{S'}$, вершины которой были бы одновременно смежны с обеими вершинами из S, то оставшиеся две вершины степени два из H должны были попасть в $G' \triangle F'$. Но тогда в $G' \triangle F'$ найдется H'. Получили противоречие с тем, что F' — решение для (G', k), значит, F' — решение для (G, k).

Обратно, пусть F — решение для инстанса $(G, p(k))$ задачи H-free Edge Deletion (Completion, Editing). Покажем, что $F \cap G'$ является решением для инстанса $(G', p(k))$ задачи H'-free Edge Deletion (Completion, Editing). Предположим, что это не так, тогда в $G' \triangle (F \cap G')$ нашелся индуцированный $H' = C_4$. Рассмотрим две противоположные вершины этого цикла и клику C_S, соответствующую этим двум вершинам. Так как в клике $2p(k) + 1$ вершина, а $|F| \leq p(k)$, то в C_S найдется такая вершина, что ни одно из ребер из нее в $V(G')$ не лежат в F. Тогда вершины H' вместе с этой вершиной индуцируют H в $G \triangle F$. Получили противоречие с тем, что F — решение для $(G, p(k))$, значит, $F \cap G'$ — решение для $(G', p(k))$. Тогда по лемме 4 мы получаем, что из существования $poly(OPT)$-приближения для H-free Edge Deletion (Completion, Editing) следует существование $poly(OPT)$-приближения для H'-free Edge Deletion (Completion, Editing). А тогда граф H — неинтересный. $\square$

Далее нам понадобится следующая конструкция, введенная в работе [2].

Конструкция 7. *Даны (G', p, t), где G' — это граф, и $p, t \geq 1$. Для каждого подмножества S из t вершин G' введем независимое множество I_S из $p+2$ вершин. Проведем ребра (v_i, u_j) для всех пар $v_i \in I_S$, $u_j \in V(G') \setminus S$. Пусть $I = \bigcup_{S \subseteq V(G'), |S| = t} I_S$. Полученный граф назовем G.*

Лемма 46. *Пусть H — такой граф, что $H[V_h]$ образует клику и для каждой пары вершин $u, v \in V_h$ граф $H - u$ изоморфен графу $H - v$. Пусть для любого $s \geq 2$ в H не существует независимого множества S размера s такого, что каждая вершина S имеет степень в H хотя бы $h - s + 1$. Тогда если для задачи H-free Edge Deletion (Completion, Editing) существует poly(OPT)-приближение, то для задачи $(H - u)$-free Edge Deletion (Completion, Editing) тоже существует poly(OPT)-приближение, где u — произвольная вершина из V_h.*

Доказательство. Пусть задача H-free Edge Deletion (Completion, Editing) допускает $q(OPT)$-приближение для некоторого полинома q. Введем полином $p(x) = q(x) \cdot x$. Пусть $H' = H - u$. Рассмотрим инстанс (G', k) задачи H'-free Edge Deletion (Completion, Editing). Применим конструкцию 7 к $(G', p(k), |V(H)| - h - 1)$, получим граф G. Пусть F' — решение для инстанса (G', k) задачи H'-free Edge Deletion (Completion, Editing). Покажем, что F' также является решением для инстанса (G, k) задачи H-free Edge Deletion (Completion, Editing). Предположим, что это не так и в $G \triangle F'$ нашелся H. Так как H не мог целиком попасть в $G' \triangle F'$, то существует вершина H, попавшая в $V(G) \setminus V(G')$. Пусть такая вершина ровно одна. Тогда она не соединена не более чем с $|V(H) - h - 1|$ остальными вершинами H. Но тогда ее степень в H хотя бы h, и она лежит в V_h. Значит, H' целиком лежит в $G' \triangle F'$, получили противоречие с тем, что F' — решение для (G', k). Тогда в $V(G) \setminus V(G')$ попало $s \geq 2$ вершин H. Рассмотрим любую из этих вершин. Ее степень в H хотя бы $|V(H)| - (|V(H)| - h - 1) - s = h - s + 1$. При этом вершины в $V(G) \setminus V(G')$ образуют независимое множество. Таким образом, мы нашли в H независимое множество размера $s \geq 2$, все вершины которого имеют степень $\geq h - s + 1$, противоречие. Таким образом, F' — решение для инстанса (G, k) задачи H-free Edge Deletion (Completion, Editing).

Обратно, пусть F — решение для инстанса $(G, p(k))$ задачи H-free Edge Deletion (Completion, Editing). Покажем, что $F \cap G'$ является решением для инстанса $(G', p(k))$ задачи H'-free Edge Deletion (Completion, Editing). Пусть это не так и в $G' \triangle (F \cap G')$ нашелся H'. Рассмотрим вершины в H', соответствующие $N(u)$ в H, и надстройку $I_{N(u)}$, соответствующую этим вершинам. Так как $|I_{N(u)}| = p(k) + 2$, а $|F| \leq p(k)$, то в $I_{N(u)}$ найдется вершина v, ребра из которой в $V(G')$ не лежат в F. Тогда вершины $\{v\} \cup V(H')$ индуцируют в $G \triangle F$ граф H. Получили противоречие с тем, что F — решение для $(G, p(k))$. Значит, $F \cap G'$ — решение для $(G', p(k))$. По лемме 4 мы получаем, что если для задачи H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, то и для $(H - u)$-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение. $\square$

Следствие 22. *Графы S_2, S_3, S_{16}, S_{17}, S_{31} — неинтересные.*

Лемма 47. *Пусть $H \in \overline{\mathcal{F}_8}$. Тогда H — неинтересный.*

Доказательство. Так как $H \in \overline{\mathcal{F}_8}$, то $H = (K_t - e) \cup K_1$ для некоторого $t \geq 6$. Пусть $H' = K_{t-2} \cup K_1$. H' получается из H удалением двух вершин степени $t - 2$. Покажем, что если для задачи H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, то и для задачи H'-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение. Предположим, что задача H-free Edge Deletion (Completion, Editing) допускает $q(OPT)$-приближение для некоторого полинома q. Введем полином $p(x) = q(x) \cdot x$. Пусть дан инстанс (G', k) задачи H'-free Edge Deletion (Completion, Editing). Применим конструкцию 7 к $(G', p(k), 1)$, получим граф G. Пусть F' — это решение для (G', k). Покажем, что F' также является решением для инстанса (G, k) задачи H-free Edge Deletion (Completion, Editing). Предположим, что это не так. Тогда в $G \triangle F'$ найдется индуцированный H. Он не может лежать целиком в $G' \triangle F'$, а значит, пересекается с I. Так как в I все вершины несмежны, то из них в H могло попасть не более трех вершин. Но тогда изолированная вершина H не могла оказаться в I, иначе она была бы несмежна максимум с тремя вершинами в H, а должна быть несмежна с $t \geq 6$. Значит, в I попало не более двух вершин H. Если ровно две вершины H попали в I, то это вершины степени $t - 2$ в H. Но тогда в $G' \triangle F'$ найдется индуцированный H', что неправда, поскольку F' — решение для (G', k). Значит, в I попала ровно одна вершина H. Тогда это вершина степени $t - 1$ в H, поскольку вершины из I несмежны ровно с одной вершиной из $V(G')$. А тогда в $G' \triangle F'$ найдется $(K_{t-1} - e) \cup K_1$, а значит, и $K_{t-2} \cup K_1 = H'$. Получили противоречие с тем, что F' — решение для (G', k), а значит, F' — решение для (G, k).

Обратно, пусть F — это решение для инстанса $(G, p(k))$ задачи H-free Edge Deletion (Completion, Editing). Покажем, что $F \cap G'$ — решение для инстанса $(G', p(k))$ задачи H'-free Edge Deletion (Completion, Editing). Предположим, что это не так и в $G' \triangle (F \cap G')$ нашелся индуцированный H'. Тогда рассмотрим множество вершин I_S, где S соответствует изолированной вершине H'. Так как $|I_S| = p(k) + 2$, а $|F| \leq p(k)$, то среди вершин I_S найдутся хотя бы две вершины, для которых ни одно ребро из них в $V(G')$ не лежит в F. Но тогда $V(H')$ вместе с этими вершинами индуцируют H в $G \triangle F$. Получили противоречие с тем, что F — решение для $(G, p(k))$. Значит, $F \cap G'$ — решение для $(G', p(k))$. А тогда по лемме 4 если для задачи H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, то и для задачи H'-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение. При этом $H' \notin \mathcal{Y}$, а значит, H — неинтересный. □

Немного модифицируем конструкцию 5 и получим еще одну конструкцию.

Конструкция 8. *Даны (G', p, H, V'), где G', H — графы, $V' \subseteq V(H)$, $p \in \mathbb{N}$. Применим конструкцию 5 к (G', p, H, V'). Рассмотрим все добавленные надстройки V_i. Для каждой пары надстроек V_i, V_j добавим ребра (x, y) между всеми парами вершин $x \in V_i$, $y \in V_j$. Полученный граф назовем G.*

Следующая лемма доказывается ровно так же, как лемма 22.

Лемма 48. *Пусть G — это результат применения конструкции 8 к (G', p, H, V') и $H' = H[V']$. Пусть F — решение для инстанса (G, p) задачи H-free Edge Deletion (Completion, Editing). Тогда $F \cap G'$ является решением для инстанса (G', p) задачи H'-free Edge Deletion (Completion, Editing).*

Лемма 49. *Пусть H — связный граф и $s \geq 1$. Тогда если для задачи sH-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, то и для задачи H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение.*

Доказательство. Будем доказывать по индукции. Для этого покажем, что если $s \geq 2$ и для задачи sH-free Edge Deletion (Completion, Editing) есть $q(OPT)$-приближение для некоторого полинома q, то и для задачи $(s-1)H$-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение. Введем полином $p(x) = q(x) \cdot x$. Пусть дан инстанс (G', k) задачи $(s-1)H$-free Edge Deletion (Completion, Editing). Применим конструкцию 8 к $(G', p(k), sH, V((s-1)H))$. Получим граф G. Пусть F' — решение для (G', k). Покажем, что F' также является решением для инстанса (G, k) задачи sH-free Edge Deletion (Completion, Editing). Предположим, что это не так, тогда в $G \triangle F'$ нашелся индуцированный sH. Так как надстройки не соединяются ребрами с G', а H — связный, то в $G' \triangle F'$ найдется tH, а в $G[V(G) \setminus V(G')]$ найдется $(s-t)H$ для некоторого $0 \leq t \leq s$. Но заметим, что в надстройках не могло оказаться двух копий H, поскольку иначе найдутся две вершины из разных копий H и при этом из разных надстроек. Но тогда такие вершины будут соединены ребром, а копии H в $(s-t)H$ не соединены. Значит $t \geq s-1$, а тогда в $G' \triangle F'$ найдется $(s-1)H$. Получили противоречие с тем, что F' — решение для (G', k). Таким образом, F' — решение для (G, k).

Обратно, пусть F — решение для инстанса $(G, p(k))$ задачи sH-free Edge Deletion (Completion, Editing). Тогда по лемме 48 $F \cap G'$ является решением для инстанса $(G', p(k))$ задачи $(s-1)H$-free Edge Deletion (Completion, Editing). А тогда по лемме 4 из наличия $poly(OPT)$-приближения для sH-free Edge Deletion (Completion, Editing) следует наличие $poly(OPT)$-приближения для $(s-1)H$-free Edge Deletion (Completion, Editing). □

Лемма 50. *Пусть H_1 — наибольшая компонента связности в H. Пусть H' — это объединение всех компонент H, изоморфных H_1. Тогда если для задачи H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, то и для задачи H'-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение.*

Доказательство. Пусть задача H-free Edge Deletion (Completion, Editing) допускает $q(OPT)$-приближение для некоторого полинома q. Введем полином $p(x) = q(x) \cdot x$. Пусть дан инстанс (G', k) задачи H'-free Edge Deletion (Completion, Editing). Применим конструкцию 5 к $(G', p(k), H, V(H'))$, получим граф G. Пусть F' — решение для (G', k). Покажем, что F' также является решением для инстанса (G, k) задачи H-free Edge Deletion (Completion, Editing). Предположим, что это не так, тогда в $G \triangle F'$ найдется H. Рассмотрим вершины H, попавшие в $V(G) \setminus V(G')$. Заметим, что они лежат в компонентах связности либо меньшего размера чем $|V(H_1)|$, либо равных по размеру, но не изоморфных H_1, поэтому это могли быть вершины только из $V(H) \setminus V(H')$. Таким образом, в $G' \triangle F'$ найдется H', что невозможно, поскольку F' — решение для (G', k). Таким образом, F' — решение для (G, k). Обратно, пусть F — решение для инстанса $(G, p(k))$ задачи H-free Edge Deletion (Completion, Editing). По лемме 22 $F \cap G'$ является решением для инстанса $(G', p(k))$ задачи H'-free Edge Deletion (Completion, Editing). Тогда по лемме 4 получаем, что если H-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение, то и H'-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение. □

Из лемм 49 и 50 получаем следствие 23.

Следствие 23. *Пусть H' — наибольшая компонента связности в H. Тогда если для задачи H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, то и для задачи H'-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение.*

Следствие 24. *Пусть $H \in \mathcal{F}_7$. Тогда H — неинтересный.*

Лемма 51. *Пусть $H = J \cup tK_1$ для некоторого графа J, ни одна компонента связности которого не является кликой, и $t \geq 2$. Пусть $H' = J \cup (t-1)K_1$. Тогда если для задачи H-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение, то и для задачи H'-free Edge Deletion (Completion, Editing) существует $poly(OPT)$-приближение.*

Доказательство. Пусть задача H-free Edge Deletion (Completion, Editing) допускает $q(OPT)$-приближение для некоторого полинома q. Введем полином $p(x) = q(x) \cdot x$. Пусть дан инстанс (G', k) задачи H'-free Edge Deletion (Completion, Editing). Введем граф $G = G' \cup K_{p(k)+1}$. Пусть F' — решение для (G', k). Покажем, что F' также является решением для инстанса (G, k) задачи H-free Edge Deletion (Completion, Editing). Предположим, что это не так, тогда в $G \triangle F'$ нашелся индуцированный H. Вершины H, попавшие в $K_{p(k)+1}$, индуцируют в H клику, являющуюся отдельной компонентой. В H такой компонентой может быть только K_1. Таким образом, H' целиком окажется в $G' \triangle F'$, получили противоречие с тем, что F' — решение для (G', k). Таким образом, F' — решение для (G, k).

Обратно, пусть F — решение для инстанса $(G, p(k))$ задачи H-free Edge Deletion (Completion, Editing). Покажем, что $F \cap G'$ — решение для $(G', p(k))$ задачи H'-free Edge Deletion (Completion, Editing). Предположим, что это не так, тогда в $G' \triangle (F \cap G')$ нашелся индуцированный H'. Так как $|F| \leq p(k)$, то в $K_{p(k)+1}$ найдется вершина, ребра из которой в $V(G')$ не лежат в F. Тогда $V(H')$ вместе с этой вершиной индуцируют H в $G \triangle F$. Получили противоречие с тем, что F — решение для $(G, p(k))$. Таким образом, $F \cap G'$ — решение для $(G', p(k))$. Тогда по лемме 4 получаем, что если H-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение, то и H'-free Edge Deletion (Completion, Editing) допускает $poly(OPT)$-приближение. □

Следствие 25. *Графы S_4 и S_6 — неинтересные.*

Таблица 7: Результаты для $\mathcal{S}$ и $\mathcal{F}$

H	Лемма	H	Лемма	H	Лемма	$H \in$	Лемма
S_1	45	S_{13}	20	S_{25}	20	$\mathcal{F}_1$	16
S_2	22	S_{14}	20	S_{26}	20	$\mathcal{F}_2$	21
S_3	22	S_{15}	18	S_{27}	19	$\mathcal{F}_3$	21
S_4	25	S_{16}	22	S_{28}	20	$\mathcal{F}_4$	21
S_5	18	S_{17}	22	S_{29}	20	$\mathcal{F}_5$	26
S_6	25	S_{18}	20	S_{30}	20	$\mathcal{F}_6$	16
S_7	20	S_{19}	20	S_{31}	22	$\mathcal{F}_7$	24
S_8	20	S_{20}	19	S_{32}	20	$\mathcal{F}_8$	47
S_9	18	S_{21}	20	S_{33}	20	$\mathcal{F}_9$	17
S_{10}	20	S_{22}	18	S_{34}	20	$\mathcal{F}_{10}$	17
S_{11}	20	S_{23}	20	S_{35}	43		
S_{12}	20	S_{24}	20	S_{36}	20		

Следствие 26. *Пусть $H \in \overline{\mathcal{F}_5}$. Тогда H — неинтересный.*

В таблице 7 для каждого графа $H \in \mathcal{S} \cup \mathcal{F}$ указан номер леммы или следствия, где доказывалось, что H или $\overline{H}$ — неинтересный. Пользуясь леммой 32, получаем, что все графы из $\mathcal{S} \cup \overline{\mathcal{S}} \cup \mathcal{F} \cup \overline{\mathcal{F}}$ — неинтересные. Таким образом, мы доказали следующую лемму.

Лемма 52. $\mathcal{H} \cup \overline{\mathcal{H}} \cup \mathcal{A} \cup \overline{\mathcal{A}} \cup \mathcal{D} \cup \overline{\mathcal{D}} \cup \mathcal{B} \cup \overline{\mathcal{B}}$ *— хорошее множество.*

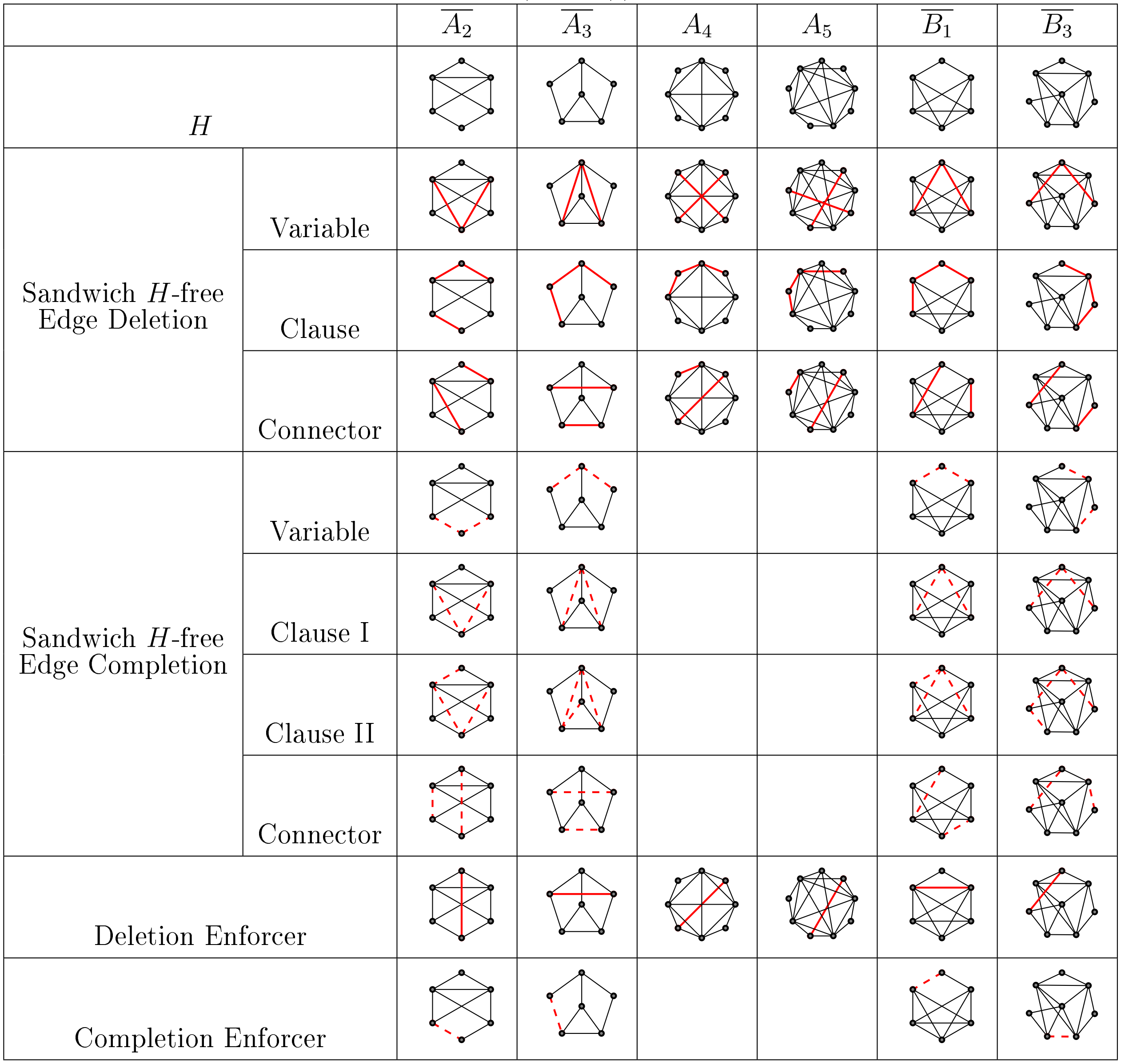
8 Избавляемся от некоторых из оставшихся графов

В этом разделе мы покажем отсутствие $poly(OPT)$-приближения для некоторых из оставшихся графов с помощью уже известной схемы: показать NP-трудность задачи Sandwich H-free Edge Deletion (Completion), потом отсутствие $poly(OPT)$-приближения для H-free Edge Deletion (Completion) с помощью Deletion (Completion) Enforcer'a, а потом отсутствие $poly(OPT)$-приближения для H-free Edge Editing с помощью Completion (Deletion) Enforcer'a. В таблице 8 приведены гаджеты, которые понадобятся нам в доказательствах. Красные сплошные ребра соответствуют разрешенным ребрам, а красные пунктирные — разрешенным антиребрам.

Рассмотрим произвольный двусвязный граф H. Пусть {Variable, Clause, Connector} — множество гаджетов для задачи Sandwich H-free Edge Deletion (Completion). Заметим, что если граф H пересекает {Variable, Clause, Connector}, то найдется граф G, полученный применением конструкции 2 к некоторой выполнимой формуле φ, выполняющему набору τ для φ и {Variable, Clause, Connector}, множество вершин $V' \subseteq V(G)$ такое, что индуцированный подграф $G[V']$ изоморфен H, а также ребро или антиребро e графа G такое, что (i) e принадлежит хотя бы двум гаджетам, а следовательно, соответствует в них разрешенным ребрам (антиребрам); (ii) e является 2-разделителем в $G[V']$, соответствующим 2-разделителю (u, v) в H.

Лемма 53. *Задачи Sandwich $\overline{A_2}$-free Edge Deletion и Sandwich $\overline{A_2}$-free Edge Completion — NP-трудные.*

Таблица 8: Гаджеты



Доказательство. Предположим, что $\overline{A_2}$ пересекает представленные в таблице 8 гаджеты для Sandwich $\overline{A_2}$-free Edge Deletion или Sandwich $\overline{A_2}$-free Edge Completion. Тогда найдутся описанные выше граф G и ребро (антиребро) e в нем, соответствующее 2-разделителю (u, v) в $\overline{A_2}$. Заметим, что в графе $\overline{A_2}$ для 2-разделителя (u, v) найдутся вершины x и y, смежные и с u, и с v, у которых есть общий сосед z, не смежный с u и v. Тогда вершины u, v, x и y должны попасть в один гаджет. Назовем такой гаджет X. Вершины X, в которые попали u, v, x и y, назовем u', v', x' и y' соответственно.

Пусть X — это Variable гаджет для Sandwich $\overline{A_2}$-free Edge Deletion, и $e = (u', v')$ — его разрешенное ребро. В зависимости от того, удалено ли второе разрешенное ребро в X, найдутся либо две, либо три вершины в X, одновременно смежные с обоими концами e. Среди них будут вершины x' и y'. Так как (x', y') не является разрешенным ребром, то в гаджете также должна присутствовать вершина z', соответствующая z в $\overline{A_2}$. Однако заметим, что для x' и y' либо не существует отличной от u', v', x' и y' вершины X, которая была бы смежна одновременно с x' и y', либо такая вершина единственная, и она соединена с u' или v' запрещенными ребром. Но в $\overline{A_2}$ вершина z несмежна с u и v. Таким образом, X не может быть Variable гаджетом для Sandwich $\overline{A_2}$-free Edge Deletion. X также не может быть Clause гаджетом для Sandwich $\overline{A_2}$-free Edge Deletion, поскольку тогда один из концов e имеет степень не более двух, а значит, не может быть одновременно смежен еще с двумя вершинами x' и y'. Пусть X — это Connector гаджет для Sandwich $\overline{A_2}$-free Edge Deletion, и $e = (u', v')$ — его разрешенное ребро. Заметим, что e не может быть ребром e_{in} этого гаджета, поскольку тогда мы получим противоречие тем же способом, что и с Variable гаджетом для Sandwich $\overline{A_2}$-free Edge Deletion. Значит, e — это ребро, один из концов которого имеет степень два, но тогда X не может быть Connector гаджетом для Sandwich $\overline{A_2}$-free Edge Deletion по тем же причинам, что и с Clause гаджетом для Sandwich $\overline{A_2}$-free Edge Deletion.

Также X не может быть Variable или Connector гаджетом для Sandwich $\overline{A_2}$-free Edge Completion, поскольку при любом способе добавления разрешенных ребер в гаджете не появится двух вершин, одновременно смежных с обоими концами e. Пусть X — это одна из частей Clause гаджета для Sandwich $\overline{A_2}$-free Edge Completion, и $e = (u', v')$ — его разрешенное антиребро. Если e — это удаленное ребро из второй части гаджета, то при любом способе добавления разрешенных ребер в гаджете не найдется двух вершин, которые были бы одновременно смежны с обоими концами e, а значит, X не может быть частью Clause гаджета для Sandwich $\overline{A_2}$-free Edge Completion. Если же e — это любое другое разрешенное антиребро X, то X не может быть частью Clause гаджета для Sandwich $\overline{A_2}$-free Edge Completion по тем же причинам, что и с Variable гаджетом для Sandwich $\overline{A_2}$-free Edge Deletion.

Таким образом, $\overline{A_2}$ не пересекает гаджеты для Sandwich $\overline{A_2}$-free Edge Deletion и Sandwich $\overline{A_2}$-free Completion. Тогда по лемме 14 получаем, что задачи Sandwich $\overline{A_2}$-free Edge Deletion и Sandwich $\overline{A_2}$-free Completion — NP-трудные. □

Лемма 54. *Задачи $\overline{A_2}$-free Edge Deletion (Completion, Editing) и A_2-free Edge Deletion (Completion, Editing) не допускают poly(OPT)-приближения, если $P \neq NP$.*

Доказательство. Заметим, что для любого 2-разделителя графа $\overline{A_2}$ с обеих сторон от него найдется вершина, смежная с обеими вершинами разделителя. Это значит, что если $\overline{A_2}$ пересекает Deletion или Completion Enforcer, то внутри этого гаджета найдется вершина, смежная одновременно с обоими концами выделенного ребра. Однако для выделенных ребер представленных в таблице 8 Deletion и Completion Enforcer'ов такой вершины не существует. Тогда по леммам 15 и 16 задачи $\overline{A_2}$-free Edge Deletion, $\overline{A_2}$-free Edge Completion, а также $\overline{A_2}$-free Edge Editing не допускают $poly(OPT)$-приближения, если $P \neq NP$. По лемме 3 задачи A_2-free Edge Deletion, A_2-free Edge Completion и A_2-free Edge Editing также не допускают $poly(OPT)$-приближения, если $P \neq NP$. □

Лемма 55. *Задачи Sandwich $\overline{A_3}$-free Edge Deletion и Sandwich $\overline{A_3}$-free Edge Completion NP-трудные.*

Доказательство. Предположим, что $\overline{A_3}$ пересекает представленные в таблице 8 гаджеты для Sandwich $\overline{A_3}$-free Edge Deletion или Sandwich $\overline{A_3}$-free Edge Completion. Тогда найдутся описанные выше формула φ и выполняющий набор τ для нее, граф G и ребро (антиребро) e в нем, соответствующее 2-разделителю (u, v) в $\overline{A_3}$. В $\overline{A_3}$ два 2-разделителя, и они симметричны друг другу. Рассмотрим 2-разделитель (u, v), для определенности пусть u — это вершина $\overline{A_3}$, участвующая в обоих 2-разделителях. Рассмотрим также вершины $x, y, z \in V(\overline{A_3})$ такие, что x смежна с обеими вершинами u и v, y смежна с x и v, а z смежна с u и y. Тогда вершины u,

v, x и y лежат в одном гаджете. Назовем такой гаджет X. Вершины X, в которые попали u, v, x и y, назовем u', v', x' и y' соответственно. Также заметим, что либо z тоже попала в X, либо (u', y') — разрешенное ребро (антиребро) в X.

Пусть X — это Variable гаджет для Sandwich $\overline{A_3}$-free Edge Deletion, и $e = (u', v')$ — его разрешенное ребро. Так как любой 2-разделитель $\overline{A_3}$ — это антиребро, то в X ребро e удалено. В этом случае второе разрешенное ребро X осталось в графе. Тогда вершина y' определяется однозначно, поскольку остальные вершины смежны одновременно и с u', и с v'. Так как у вершины y' степень в X ровно два, то однозначно восстанавливается вершина x'. Так как y' не инцидентна ни одному разрешенному ребру, то в X также должна найтись вершина z', соответствующая z в $\overline{A_3}$. Однако у вершины y' больше нет других соседей, а значит, мы получили противоречие, и X не может быть Variable гаджетом для Sandwich $\overline{A_3}$-free Edge Deletion. X также не может быть Clause гаджетом для Sandwich $\overline{A_3}$-free Edge Deletion, поскольку при любом способе удаления разрешенных ребер из X для ребра e не найдется вершины в гаджете, одновременно смежной с обоими концами e. Пусть X — это Connector гаджет для Sandwich $\overline{A_3}$-free Edge Deletion, и $e = (u', v')$ — его разрешенное ребро. Заметим, что вершина x' однозначно восстанавливается, поскольку при любом способе удаления разрешенных ребер существует всего одна вершина в X, одновременно смежная и с u', и с v'. У x' есть единственный сосед в X, отличный от u' и v'. Однако этот сосед не может быть y', поскольку он несмежен с обоими концами e. Таким образом, X не может быть Connector гаджетом для Sandwich $\overline{A_3}$-free Edge Deletion.

X также не может быть Variable гаджетом для Sandwich $\overline{A_3}$-free Edge Completion, поскольку при любом способе добавления разрешенных ребер в X не найдется вершины, смежной одновременно с обоими концами e. Заметим, что X не может быть первой частью Clause гаджета для Sandwich $\overline{A_3}$-free Edge Completion по тем же причинам, что и для Variable гаджета для Sandwich $\overline{A_3}$-free Edge Deletion. Пусть X — это вторая часть Clause гаджета для Sandwich $\overline{A_3}$-free Edge Completion, и $e = (u', v')$ — его разрешенное антиребро. Рассмотрим такой клоз $C = (\ell_1 \vee \ell_2 \vee \ell_3)$ формулы φ, что X — это часть Clause гаджета, построенного для этого клоза. Если $\tau(\ell_2) \vee \tau(\ell_3) = 1$, то ребро $e_{\ell_2 \vee \ell_3}$ добавлено в X. Так как любой 2-разделитель — это антиребро, то $e \neq e_{\ell_2 \vee \ell_3}$. Значит, e — это одно из ребер e_{ℓ_2} и e_{ℓ_3}. В этом случае X не может быть второй частью Clause гаджета для Sandwich $\overline{A_3}$-free Edge Completion по тем же причинам, что и для Variable гаджета для Sandwich $\overline{A_3}$-free Edge Deletion. Если же $\tau(\ell_2) \vee \tau(\ell_3) = 0$, то в X не добавлено ни одного ребра, а тогда в нем нет ни одного индуцированного K_3. При этом вершины x', y' и v' индуцируют K_3 в X. Получили противоречие, значит, X не может быть второй частью Clause гаджета для Sandwich $\overline{A_3}$-free Edge Completion. Также X не может быть Connector гаджетом для Sandwich $\overline{A_3}$-free Edge Completion по тем же причинам, что и для Connector гаджета для Sandwich $\overline{A_3}$-free Edge Deletion.

Таким образом, $\overline{A_3}$ не пересекает гаджеты для Sandwich $\overline{A_3}$-free Edge Deletion и Sandwich $\overline{A_3}$-free Completion. Тогда по лемме 14 получаем, что задачи Sandwich $\overline{A_3}$-free Edge Deletion и Sandwich $\overline{A_3}$-free Completion — NP-трудные. □

Лемма 56. *Задачи $\overline{A_3}$-free Edge Deletion (Completion, Editing) и A_3-free Edge Deletion (Completion, Editing) не допускают $poly(OPT)$-приближения, если $P \neq NP$.*

Доказательство. Заметим, что любой 2-разделитель $\overline{A_3}$ является антиребром, а тогда $\overline{A_3}$ не пересекает любой Deletion Enforcer. Тогда по лемме 15 получаем, что $\overline{A_3}$-free Edge Deletion не допускает $poly(OPT)$-приближения, если $P \neq NP$. Для любого 2-разделителя графа $\overline{A_3}$ с обеих сторон от него найдется вершина, смежная с обеими вершинами разделителя. Это значит, что если $\overline{A_3}$ пересекает Completion Enforcer, то внутри этого гаджета найдется вершина, смежная одновременно с обоими концами выделенного ребра. Однако для выделенного ребра представленного в таблице 8 Completion Enforcer'а такой вершины не существует. Таким образом, по лемме 16 получаем, что задачи $\overline{A_3}$-free Edge Completion и $\overline{A_3}$-free Edge Editing не допускают $poly(OPT)$-приближения при $P \neq NP$. По лемме 3 задачи A_3-free Edge Deletion, A_3-free Edge Completion и A_3-free Edge Editing также не допускают $poly(OPT)$-приближения при $P \neq NP$. □

Лемма 57. *Пусть $H \in \{A_4, A_5, \overline{B_3}\}$. Тогда задачи Sandwich H-free Edge Deletion и Sandwich H-free Edge Completion — NP-трудные.*

Доказательство. Предположим, что H пересекает представленные в таблице 8 гаджеты для Sandwich H-free Edge Deletion или Sandwich H-free Edge Completion. Тогда найдется описанный выше граф G и ребро (антиребро) e в нем, соответствующее 2-разделителю (u, v) в H. Заметим, что для 2-разделителя (u, v) графа H существуют две смежные вершины x и y, также смежные одновременно с u и v. Тогда эти вершины должны попасть в один гаджет с u и v. Рассмотрим любой из гаджетов для Sandwich H-free Edge Deletion и некоторое его разрешенное ребро e. Заметим, что при любом способе удаления разрешенных ребер в гаджете не найдется двух вершин, одновременно смежных с обоими концами e. Значит, H не пересекает гаджеты для Sandwich H-free Edge Deletion. Тогда по лемме 14 получаем, что задача Sandwich H-free Edge Deletion — NP-трудная. Если $H \in \{A_4, A_5\}$, $H = \overline{H}$, поэтому по лемме 8 Sandwich H-free Edge Completion — NP-трудная. Пусть $H = \overline{B_3}$. Рассмотрим любой из гаджетов для Sandwich H-free Edge Completion и некоторое его разрешенное антиребро e. Заметим, что при любом способе добавления разрешенных ребер в гаджете не найдется двух вершин, одновременно смежных с обоими концами e. Значит, H не пересекает гаджеты для Sandwich H-free Edge Completion. Тогда по лемме 14 получаем, что задача Sandwich H-free Edge Completion — NP-трудная. □

Лемма 58. *Пусть $H \in \{A_4, A_5\}$. Задача H-free Edge Deletion (Completion, Editing) не допускает $poly(OPT)$-приближения, если $P \neq NP$.*

Доказательство. Заметим, что для любого 2-разделителя графа H с обеих сторон от него найдется вершина, смежная с обеими вершинами разделителя. Это значит, что если H пересекает Deletion Enforcer, то внутри этого гаджета найдет вершина, смежная одновременно с обоими концами выделенного ребра. Однако для выделенного ребра представленного в таблице 8 Deletion Enforcer'а такой вершины не существует. Тогда по лемме 15 получаем, задача H-free Edge Deletion не допускает $poly(OPT)$-приближения при $P \neq NP$. Так как $H = \overline{H}$, то по лемме 3 задача H-free Edge Completion тоже не допускает $poly(OPT)$-приближения при $P \neq NP$. Применим еще раз лемму 15 и получим, что задача H-free Edge Editing не допускает $poly(OPT)$-приближения при $P \neq NP$. □

Лемма 59. *Задачи $\overline{B_3}$-free Edge Deletion (Completion, Editing) и B_3-free Edge Deletion (Completion, Editing) не допускают $poly(OPT)$-приближения, если $P \neq NP$.*

Доказательство. Заметим, что для любого 2-разделителя графа $\overline{B_3}$ с обеих сторон от него найдется вершина, смежная с обеими вершинами разделителя. Это значит, что если $\overline{B_3}$ пересекает Deletion Enforcer, то внутри этого гаджета найдется вершина, смежная одновременно с обоими концами выделенного ребра. Однако для выделенного ребра представленного в таблице 8 Deletion Enforcer'а такой вершины не существует. Тогда по лемме 15 получаем, что $\overline{B_3}$-free Edge Deletion не допускает $poly(OPT)$-приближения при $P \neq NP$. Так как любой

2-разделитель графа $\overline{B_3}$ является ребром, то $\overline{B_3}$ не пересекает любой Completion Enforcer. Тогда по лемме 16 получаем, что задачи $\overline{B_3}$-free Edge Completion и $\overline{B_3}$-free Edge Editing не допускают $poly(OPT)$-приближения при $P \neq NP$. По лемме 3 задачи B_3-free Edge Deletion, B_3-free Edge Completion и B_3-free Edge Editing также не допускают $poly(OPT)$-приближения при $P \neq NP$. □

Лемма 60. *Задачи Sandwich $\overline{B_1}$-free Edge Deletion и Sandwich $\overline{B_1}$-free Edge Completion — NP-трудные.*

Доказательство. У $\overline{B_1}$ есть всего один 2-разделитель (u, v), при этом с одной стороны от него лежит индуцированный $K_5 - e$, в котором $e = (u, v)$. Так как $K_5 - e$ трехсвязный, то если $\overline{B_1}$ пересекает гаджеты, то $K_5 - e$ должен целиком попасть в какой-то один гаджет и e должно соответствовать разрешенному ребру этого гаджета.

Рассмотрим любой из представленных в таблице 8 гаджетов для Sandwich $\overline{B_1}$-free Edge Deletion и Sandwich $\overline{B_1}$-free Edge Completion и некоторое его разрешенное ребро (антиребро) q. Предположим, что после удаления (добавления) каких-то разрешенных ребер в гаджете нашелся индуцированный $K_5 - e$, где e соответствует q. Это значит, что ребро q удалено. Но тогда при добавлении всех разрешенных ребер в гаджете найдется индуцированный K_5, а это не так. Значит, $\overline{B_1}$ не может пересечь этот гаджет так, чтобы $K_5 - e$ попал в этот гаджет и e соответствовало разрешенному ребру. Тогда по лемме 14 получаем, что задачи Sandwich $\overline{B_1}$-free Edge Deletion и Sandwich $\overline{B_1}$-free Edge Completion являются NP-трудными. □

Лемма 61. *Задачи $\overline{B_1}$-free Edge Deletion (Completion, Editing) и B_1-free Edge Deletion (Completion, Editing) не допускают $poly(OPT)$-приближения, если $P \neq NP$.*

Доказательство. Заметим, что 2-разделитель $\overline{B_1}$ является антиребром, а тогда $\overline{B_1}$ не пересекает любой Deletion Enforcer. Тогда по лемме 15 получаем, что $\overline{B_1}$-free Edge Deletion не допускает $poly(OPT)$-приближения при $P \neq NP$.

В $\overline{B_1}$ с обеих сторон от 2-разделителя найдется вершина, смежная с обеими вершинами разделителя. Это значит, что если $\overline{B_1}$ пересекает Completion Enforcer, то внутри этого гаджета найдется вершина, смежная одновременно с обоими концами выделенного ребра. Однако для выделенного ребра представленного в таблице 8 Completion Enforcer'а такой вершины не существует. Таким образом, по лемме 16 получаем, что задачи $\overline{B_1}$-free Edge Completion и $\overline{B_1}$-free Edge Editing не допускают $poly(OPT)$-приближения при $P \neq NP$. По лемме 3 задачи B_1-free Edge Deletion, B_1-free Edge Completion и B_1-free Edge Editing также не допускают $poly(OPT)$-приближения при $P \neq NP$. □

Таким образом, мы показали, что графы A_2 $\overline{A_2}$, A_3, $\overline{A_3}$, $A_4 = \overline{A_4}$, $A_5 = \overline{A_5}$, B_1, $\overline{B_1}$, B_3, $\overline{B_3}$ — неинтересные. Тогда множество графов $\mathcal{G} \cup \overline{\mathcal{G}}$, представленное в таблице 1, — хорошее. Заметим, что по лемме 3, если $\forall H \in \mathcal{G}$ задача H-free Edge Editing не допускает $poly(OPT)$-приближения, то $\forall H \in \overline{\mathcal{G}}$ задача H-free Edge Editing также не допускает $poly(OPT)$-приближения. Таким образом, мы доказали теорему 1.

Заключение

В данной работе были исследованы задачи H-free Edge Deletion (Completion, Editing) с точки зрения сложности приближения и были сформулированы гипотезы 1, 2, 3, согласно которым все графы на хотя бы пяти вершинах можно классифицировать на несколько групп по сложности приближения соответствующих им задач H-free Edge Deletion (Completion, Editing), при этом в каждой группе графы обладают характерными только этой группе структурными свойствами. Также в работе было сделано существенное продвижение в доказательстве этих гипотез, а именно доказана теорема 1. Теперь для построения полной классификации задач H-free Edge Deletion (Completion, Editing) для графов H с хотя бы пятью вершинами по сложности их приближения достаточно определить сложность приближения этих задач лишь для графов $H \in \mathcal{G} \cup \overline{\mathcal{G}}$, где $\mathcal{G}$ — множество, состоящее из семнадцати графов. Открытыми вопросами на данный момент также являются сложность приближения Min Horn Deletion-полных задач и сложность приближения задач H-free Edge Deletion (Completion, Editing) для графов H с не более чем четырьмя вершинами.

Также в работе мы проследили связь между сложностью приближения и сложностью кернелизации для задач H-free Edge Deletion (Completion, Editing). Мы показали, что в данном случае подходы, введенные для доказательства отсутствия полиномиальных ядер, можно использовать для доказательства отсутствия $poly(OPT)$-приближений. Также данная работа подкрепляет гипотезу о том, что сложность кернелизации задач H-free Edge Deletion (Completion, Editing) в целом ведет себя так же, как и сложность их приближения. Однако в случае со сложностью приближения ситуация обстоит несколько сложнее, поскольку задачи, соответствующие графам ровно с одним ребром или антиребром, образуют отдельный класс.

Список литературы

[1] Faisal N Abu-Khzam. “Kernelization algorithms for d-hitting set problems”. В: *Workshop on Algorithms and Data Structures*. Springer. 2007, с. 434—445.

[2] NR Aravind, RB Sandeep и Naveen Sivadasan. “Dichotomy results on the hardness of H-free edge modification problems”. В: *SIAM Journal on Discrete Mathematics* 31.1 (2017), с. 542—561.

[3] Ivan Bliznets и др. “Hardness of approximation for H-free edge modification problems”. В: *ACM Transactions on Computation Theory (TOCT)* 10.2 (2018), с. 1—32.

[4] Leizhen Cai. “Fixed-parameter tractability of graph modification problems for hereditary properties”. В: *Information Processing Letters* 58.4 (1996), с. 171—176.

[5] Leizhen Cai и Yufei Cai. “Incompressibility of H-Free Edge Modification Problems”. В: *Algorithmica* 71.3 (2015), с. 731—757.

[6] Yufei Cai. “Polynomial kernelisation of H-free edge modification problems”. Дис. … док. Chinese University of Hong Kong, 2012.

[7] Ehab S El-Mallah и Charles J Colbourn. “The complexity of some edge deletion problems”. В: *IEEE transactions on circuits and systems* 35.3 (1988), с. 354—362.

[8] Archontia C Giannopoulou и др. “Tree Deletion Set Has a Polynomial Kernel but No OPTˆO(1) Approximation”. В: *SIAM Journal on Discrete Mathematics* 30.3 (2016), с. 1371—1384.

[9] Sylvain Guillemot и др. “On the (non-) existence of polynomial kernels for P l-free edge modification problems”. В: *Algorithmica* 65.4 (2013), с. 900—926.

[10] Haim Kaplan, Ron Shamir и Robert E Tarjan. “Tractability of parameterized completion problems on chordal, strongly chordal, and proper interval graphs”. В: *SIAM Journal on Computing* 28.5 (1999), с. 1906—1922.

[11] Sanjeev Khanna и др. “The approximability of constraint satisfaction problems”. В: *SIAM Journal on Computing* 30.6 (2001), с. 1863—1920.

[12] Stefan Kratsch и Magnus Wahlström. “Two edge modification problems without polynomial kernels”. В: *International Workshop on Parameterized and Exact Computation*. Springer. 2009, с. 264—275.

[13] John M Lewis и Mihalis Yannakakis. “The node-deletion problem for hereditary properties is NP-complete”. В: *Journal of Computer and System Sciences* 20.2 (1980), с. 219—230.

[14] Dániel Marx и RB Sandeep. “Incompressibility of H-free edge modification problems: Towards a dichotomy”. В: *arXiv preprint arXiv:2004.11761* (2020).

[15] Mihalis Yannakakis. “Edge-deletion problems”. В: *SIAM Journal on Computing* 10.2 (1981), с. 297—309.