The Complexity of the Hajós Calculus for Planar Graphs

Kazuo Iwama and Suguru Tamaki

Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan.
{iwama, tamak}@kuis.kyoto-u.ac.jp

Abstract

The planar Hajós calculus is the Hajós calculus with the restriction that all the graphs that appear in the construction (including a final graph) must be planar. We prove that the planar Hajós calculus is polynomially bounded iff the Hajós calculus is polynomially bounded.

1 Introduction

One of the most important open questions in complexity theory is whether or not extended Frege systems, the most powerful proof systems ever known for propositional formulas, are polynomially bounded. Since extended Frege systems are very general, an obvious approach to this open question is to seek a reduction to another system which appears more structured and/or less powerful. Pitassi and Urquhart [21] made an important step to this goal, namely, they proved that the above open question is equivalent to whether the Hajós calculus, which is a simple, nondeterministic procedure for generating non-3-colorable graphs, is polynomially bounded. Thus, the famous open question in proof complexity is beautifully linked to the open question in graph theory; in order to prove superpolynomial lower bounds for the extended Frege systems, it now suffices to find a "hard example" from the set of *non-3-colorable graphs*. Thanks to the long and extensive research history of graph theory and graph algorithms, this is hopefully easier than finding a hard example from the set of formulas. In this paper, we make another step toward this direction by showing that it still suffices if Hajós calculus is restricted to within the class of *planar graphs*, not only for the final graph but also intermediate ones. More formally:

Our contribution The Hajós calculus consists of three rules (see the next section), each of which modifies a graph into another. For a given graph G, its *construction* is a sequence of graphs $G_1, G_2, \ldots, G_m = G$ such that each G_i is a K_4 or follows from its previous graph(s) by applying one of the rules. Suppose that G is a non-3-colorable *planar* graph. Since the Hajós calculus is complete, there must be such a construction if we allow *non-planar graphs* for G_i's. Our new generating system, the *planar Hajós calculus*, requires all the intermediate graphs to be also planar. Since each rule of the Hajós calculus can easily violate planarity, this requirement imposes a strong restriction in applying the rules and therefore the resulting system seems significantly weaker than the original one. (In fact, even the completeness proof needs much more work than the original proof.) Nevertheless we prove that the worst-case complexity of the planar Hajós calculus is polynomially equivalent to that of the general Hajós calculus, i.e., the former is polynomially bounded for all non-3-colorable planar graphs iff so is the latter for all non-3-colorable (general) graphs.

Thus, combined with [21], we would be able to claim a superpolynomial lower bound of extended Frege systems by finding planar non-3-colorable graphs which need superpolynomial steps for its construction by the planar Hajós calculus. To do so, we could use many graph properties specific to planar graphs. For example there is always a small separator for a planar graph, which enables us, for example, to design sub-exponential-time algorithms for many NP-hard problems (including 3-colorability) and to obtain nontrivial size lower bounds for planar circuits [19]. Planar graphs of course admit planar embedding, which is also useful for designing e.g., linear-time algorithms for isomorphism testing for planar graphs [15] and PTAS for the planar TSP [11]. Most importantly, every planar graph is 4-colorable [2, 3], and we have the detailed case-analysis for efficiently coloring planar graphs. We thus believe that our one-step from the Hajós calculus to the planar Hajós calculus is not too small. Note that, although it is very unlikely, we could also claim $\mathcal{NP} = co\mathcal{NP}$ by proving the planar Hajós calculus is polynomially bounded, by taking these advantages.

Related work We briefly review the history on proving lower bounds for propositional proof systems. As formalized by Cook and Reckhow [6], there exists a propositional proof system providing short (polynomial-size) proofs for all tautologies if and only if $\mathcal{NP} = co\mathcal{NP}$. In other words, to prove superpolynomial lower bounds for powerful proof systems is a good evidence for $\mathcal{NP} \neq co\mathcal{NP}$. To do so for the extended Frege systems is an obvious goal, but people had known that is extremely hard and research interests have naturally shifted into their subsystems. Resolution is one of the most studied such a proof system. First superpolynomial lower bounds for Resolution were obtained by Tseitin [26] in the special case of regular Resolution and this bound was improved to an exponential one by Galil [8]. Haken [14] proved the first superpolynomial (actually exponential) lower bounds for general Resolution. After Haken's breakthrough, several lower bounds were obtained for stronger proof systems. Ajtai [1] gave superpolynomial lower bounds for bounded-depth Frege proofs, and Pitassi et. al. [20] and Krajíček et. al. [18] improved the bound to an exponential one. These results lead exponential lower bounds for the subsystems of the Hajós calculus [21, 16]. There are also several proof systems for which superpolynomial lower bounds are known, including Gomory-Chvátal cutting planes [22], Lovász-Schrijver systems [7] and PCR [4]. More backgrounds on proof complexity can be found in [5, 17, 23, 24, 25, 27].

2 Hajós Calculus

Although the Hajós calculus generates non-k-colorable graphs for general k (≥ 3), we only consider $k = 3$ in this paper. The set of initial graphs in the Hajós calculus contains all graphs isomorphic to complete graph K_4. There are three rules for generating new graphs:

1. **Vertex/Edge Introduction Rule**: Add (any number of) vertices and edges.

2. **Join Rule**: Let G_1 and G_2 be disjoint graphs, a and b adjacent vertices in G_1, and a' and b' adjacent vertices in G_2. Construct a graph G_3 from $G_1 \cup G_2$ as follows. First, remove edges (a, b) and (a', b'); then add an edge (b, b'); lastly, contract vertices a and a' into a single vertex. (See Fig. 1(i))

3. **Contraction Rule**: Contract two nonadjacent vertices into a single vertex, and remove any resulting duplicated edges.

Vertex/Edge Introduction Rule implies that if a subgraph of G has a construction, G also has a construction. Rules 1 and 2 increase vertices and/or edges, but Rule 3 reduces vertices and edges,

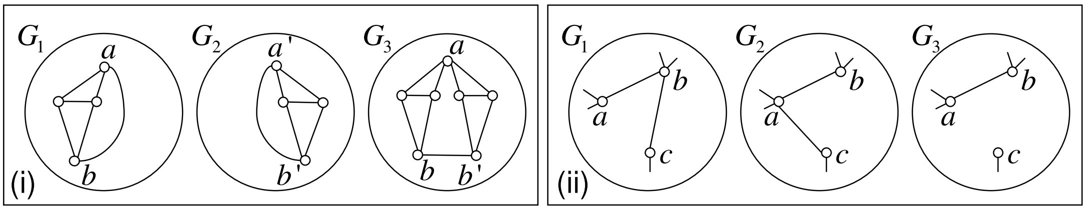

Figure 1: (i)Join Rule (ii)Edge Elimination Rule

thus the construction may not be polynomially bounded or the number of construction steps may not be bounded by polynomial in $|G|$. There is another version of the Hajós calculus, denoted by $\mathcal{HC}$. The system $\mathcal{HC}$ has the same set of initial graphs, as well as Rules 1 and 3 of the Hajós calculus, but Rule 2 is replaced by the following rule:

4. **Edge Elimination Rule**: Let G_1 and G_2 be two graphs with common vertex set $\{a, b, c, \dots\}$ which are identical except that G_1 contains edges (a, b) and (b, c) and not (a, c), whereas G_2 contains edges (a, b) and (a, c) and not (b, c). Then from G_1 and G_2, we can construct a graph G_3 that is identical to G_1 but does not contain (b, c) (See Fig. 1(ii)).

Let $\mathcal{C}$ and $\mathcal{C}'$ be two graph calculus systems, then $\mathcal{C}$ *p-simulates* $\mathcal{C}'$ if there is a polynomial-time computable function f so that for all graphs G, if σ is a graph construction of G in $\mathcal{C}'$, then $f(\sigma)$ is a graph construction of G in $\mathcal{C}$. $\mathcal{C}$ and $\mathcal{C}'$ are *p-equivalent* if $\mathcal{C}$ p-simulates $\mathcal{C}'$ and $\mathcal{C}'$ p-simulates $\mathcal{C}$.

Proposition 1 ([21]). *$\mathcal{HC}$ is p-equivalent to the Hajós calculus.*

3 Planar Hajós Calculus

Now we introduce our new system, the planar Hajós calculus. Suppose that a sequence of graphs $G_1, G_2, \dots, G_m$ satisfies the following conditions: (i) All G_i are planar. (ii) Each G_i is K_4 or is constructed from previous graph(s) by one of the three rules of $\mathcal{HC}$. Then we say that G_m is *constructed* by *planar* $\mathcal{HC}$ or $\mathcal{PHC}$. Note that Rules 1 and 3 (but not Rule 4) may violate the planarity of the graph. So, the definition is equivalent to the following: When we introduce a new edge between vertices a and b of G_i, then there must be a planar embedding of G_i such that a and b are on the same face. When we apply Contraction Rule between vertices a and b of G_i, then there must be a planar embedding of G_i such that for all vertices x being adjacent to a, vertex b is also adjacent to x or b and x are on the same face.

In some cases, this planarity restriction is quite annoying. Fig. 2(i) shows a simple example. Suppose that we wish to remove the chord (u, v) to make a face of size five in some planar graph as G_1. Then what we would do is to construct another planar graph as G_2 and apply Edge Elimination Rule to obtain G_3. One should notice, however, that this can be done because we can draw the other cord (u, w) without violating planarity and that it is no longer obvious if such a chord elimination is still possible for a face of size *four*.

To overcome this difficulty, we introduce a new Edge Elimination Rule.

5. **Edge Elimination Rule II**: Let G_1 be a graph with vertices $\{a, b, \dots\}$ that contains an edge (a, b), and G_2 be the same graph as G_1 except that vertices a and b (after removing the edge between them) are contracted. Then from G_1 and G_2, we can construct a graph G_3 that is identical to G_1 but does not contain (a, b) (See Fig. 2(ii)).

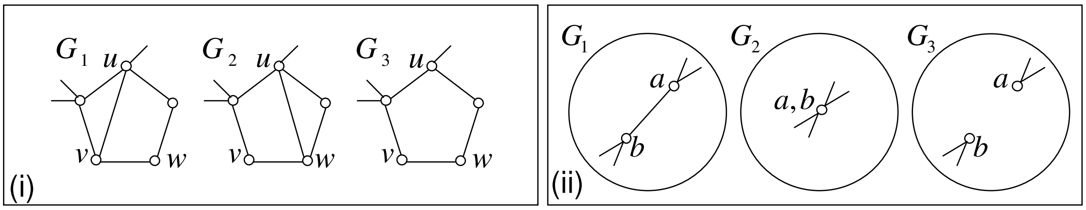

Figure 2: (i)Removing chords (ii)Edge Elimination Rule II

To make the difference clear, Rule 4 is called Edge Elimination Rule I from now on. This rule obviously keeps non-3-colorability and the following fact shows that it is at least as powerful as the Rule 4. See Fig. 1(ii). Let G_4 be a graph obtained by contracting an edge (a, c) of G_1. Then we get G_3 from G_2 and G_4 by Edge Elimination II, meaning Rule 4 can be simulated by Rules 5 and 3. (Consequently, notice that Rules 1, 3 and 5 are a new complete system for generating non-3-colorabe graphs.)

Thus adding Rule 5 to $\mathcal{PHC}$ may seem to increase the power of the system, but we can prove that this is not the case, i.e., Rule 5 can be simulated by $\mathcal{PHC}$ in polynomial steps, as shown in Lemma 3 of section 5. It turns out that the new rule is quite convenient for dealing with faces of size four, which plays an important role in the rest of the paper.

Obviously $\mathcal{PHC}$ is sound, i.e., all graphs generated by $\mathcal{PHC}$ are non-3-colorable (planar) graphs. Let $L_{\mathcal{PHC}}$ be the set of such graphs generated by $\mathcal{PHC}$. What we want to prove to attain our goal is that $\mathcal{HC}$ generates all non-3-colorable graphs in polynomial steps if and only if $\mathcal{PHC}$ generates all graphs in $L_{\mathcal{PHC}}$ in polynomial steps. Thus $L_{\mathcal{PHC}}$ does not necessarily contain all non-3-colorable planar graphs or $\mathcal{PHC}$ is not necessarily complete. In fact there is no obvious extension of the proof for the $\mathcal{HC}$'s completeness to the proof for the $\mathcal{PHC}$'s completeness. Fortunately, however, the proof of our main theorem immediately implies the completeness of $\mathcal{PHC}$, which is an important by-product of this paper.

4 Planarization of a Graphs

Intuitively speaking, our main theorem clams that $\mathcal{PHC}$ is as powerful as $\mathcal{HC}$. To prove this, the natural approach is to develop a simulation of $\mathcal{HC}$ by $\mathcal{PHC}$: Suppose that a planar graph G can be generated by $\mathcal{HC}$ by a sequence of (maybe non-planar) graphs $G_1, G_2, \ldots, G_m = G$. Then what we do is to define *planar* graphs $H_1, H_2, \ldots, H_m = G$ such that each H_i is "similar" to G_i and it can be generated by $\mathcal{PHC}$ from previous H_j's $(j < k)$ in polynomial steps. To define the similarity, we can use the so-called the *Crossover Gadget*; [10] showed that for a given (non-planar) drawing $\widehat{G}$ of a graph G, we can construct a planar graph H such that G is 3-colorable iff H is 3-colorable. (A graph is drawn in the plane in such a way that each vertex v is represented by a point and each edge (u, v) by a continuous line connecting the two points corresponding to u and v.)

Definition 1 ([10]). *The Crossover Gadget, denoted by $\diamond$, is a planar graph given in Fig. 3(i). Outer vertices a and c (b and d, also) are said to be opposite. One can easily see that opposite vertices must have the same color in any proper 3-coloring.*

Using this gadget, the non-planar drawing of G_1 of Fig. 3(ii) is converted to a planar graph G_1', where X and Y are Crossover Gadgets. More formally:

Definition 2. *For a given drawing G of a graph, its planarization $P(G)$ is a planar graph constructed by the following procedure: (i)Each crossing of G is replaced by a $\diamond$ (see Fig. 3(iii)(a)–(b)).*

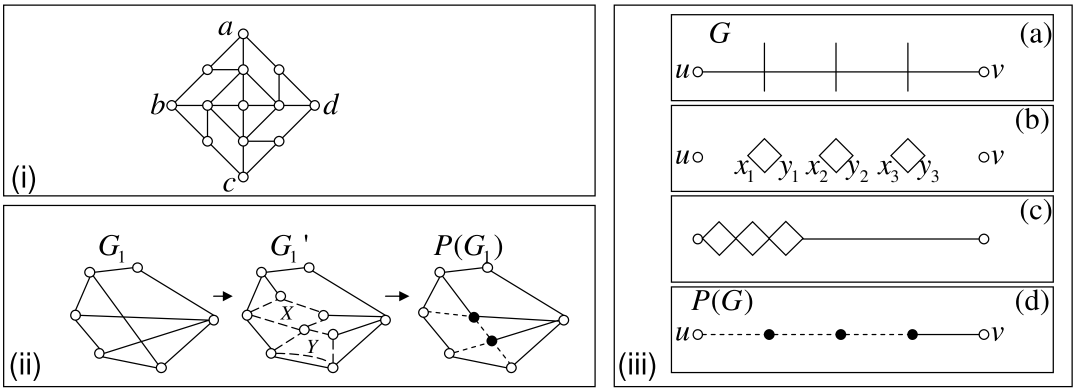

Figure 3: (i)Crossover Gadget (ii)Example of Planarization (iii)Planarization Process

(ii)Let $u, x_1, y_1, \ldots, x_k, y_k, v$ be vertices corresponding to edge (u, v) in G, where x_i and y_i are pairs of opposite vertices of each introduced $\diamond$'s, and consider pairs of vertices $(u, x_1), (y_1, x_2), \ldots, (y_k, v)$. Draw an edge for exactly one of these $k + 1$ pairs and contract all the others. (See Fig. 3(iii)(c)).

The structure as shown in Fig. 3(iii)(c) is called an *extended edge* (or *E-edge* for short) and is also illustrated as in Fig. 3(iii)(d), where dotted lines show contractions and •'s show Crossover Gadgets. Fig. 3(ii) shows such a representation of $P(G_1)$.

5 Basic Tools of $\mathcal{PHC}$

In this section we will prove a key lemma (Lemma 1). Suppose that there is a sequence $G_1, G_2, \ldots, G_m$ of planar graphs such that (i)G_1 is any (non-3-colorable, often omitted) planar graph (called an *axiom*) (ii)For each $2 \leq i \leq m$, G_i is K_4 or can be derived from previous graphs by $\mathcal{PHC}$ in polynomial steps. Then we write $G_1 \overset{*}{\Rightarrow} G_m$. We also write $G_1, G_2 \overset{*}{\Rightarrow} G_m$ if we need two axioms.

Lemma 1 (Redrawing). *Suppose G_1 and G_2 are two drawings of the same (not necessarily planar) graph. Then $P(G_1) \overset{*}{\Rightarrow} P(G_2)$ in poly$(|G_1|) + |G_2|)$ steps.*

The following lemmas provide convenient tools to prove $G_1 \overset{*}{\Rightarrow} G_2$ and to prove Lemma 1.

Lemma 2 (Triangle Elimination). *Let G_1 be a planar graph having a vertex v with degree at most two, and G_2 be the (obviously planar) graph obtained by removing v and its outgoing edges from G_1. Then $G_1 \overset{*}{\Rightarrow} G_2$ in polynomial steps.*

Proof. If v's degree is zero, all we have to do is to merge it to a nearby vertex. Suppose that v's degree is one. Then v has only one edge, (u, v), and if u is adjacent to another vertex w, then we can contract v and w. Otherwise, contract u and v with u' and v' such that an edge exists between them (If no such u' and v' exist, then the graph would be 3-colorable).

So, we can restrict ourselves to the case that v is of degree two. See Fig. 4(i). Let a and b be the two vertices adjacent to v and there may or may not be an edge between a and b. We add vertices and edges as G_3 and G_4, and get G_5 by Edge Elimination I. Now we are going to remove triangle a, v', v'' (vertices v', v'' and the three edges). This is the main part of this lemma and therefore we call this procedure *Triangle Elimination*. If a is a part of another triangle a, c', c'' as shown in G_6, then we just contract v' and c' and v'' and c''.

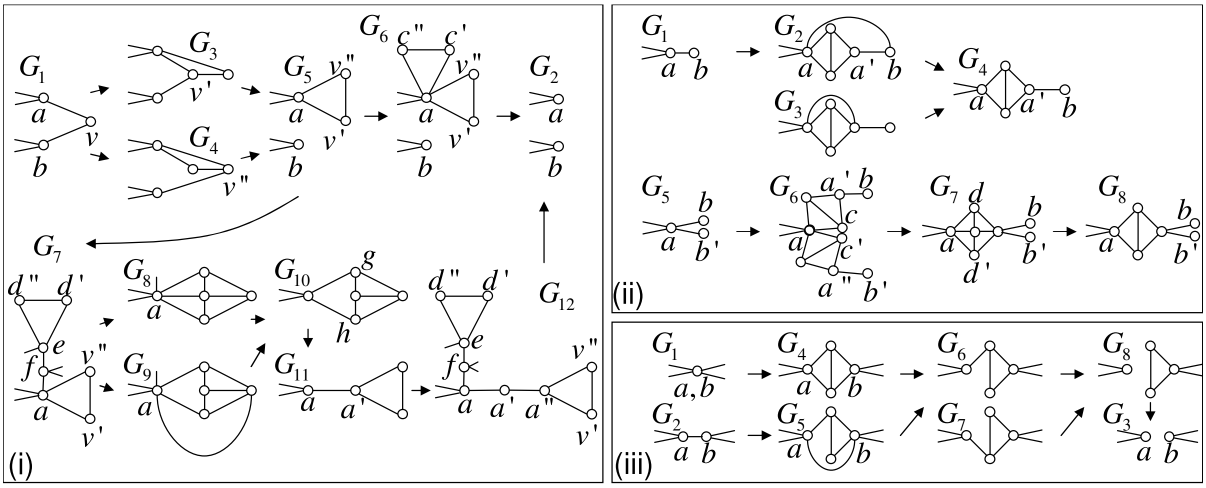

Figure 4: (i)Triangle Elimination (ii)Equality Introduction (iii)Edge Elimination II

Otherwise, we look for a triangle near a (say, e, d', d'' in G_7) which is guaranteed to exist somewhere since the underlying graph is a non-3-colorable, planar graph [12]. Then we continue to change the graph into as G_8 and G_9 by Vertex/Edge Introduction then G_{10} by Edge Elimination I, and G_{11} by Contraction (of vertices g and h), which allows us to introduce one extra edge (a, a') to the triangle. By repeating the same procedure, we can get another extra edge (a', a'') as in G_{12}.

Now we can contract a' and f, a'' and e, v' and d', and v'' and d''. Extension to the general case is straightforward. $\square$

Lemma 3 (Simulation of Edge Elimination II). *Edge Elimination II can be simulated by $\mathcal{PHC}$ in polynomial steps.*

Proof. For the simulation, we first need a tool, what we call *Equality Introduction* (see Fig. 4(ii)). Consider an arbitrary vertex, say, a, as in G_5. Our goal is to split a into two vertices a and a' and to put two triangles with a shared edge between them as G_8. The edges from a are arbitrarily divided into from a and from a' whenever the resulting graph is a planar graph. If the number of such divided edges from a' (or from a) is one, see $G_1 \sim G_4$. From G_1 to G_2, a simple Vertex/Edge Introduction is enough, G_3 can be constructed from K_4, and G_4 is due to Edge Elimination I from G_2 and G_3. If there are two edges from a', see $G_5 \sim G_8$ (The case that there are three or more edges from a' is similar and omitted). Repeat the above procedure twice to get G_6 and contract a' and a'' and c and c' to get G_7. Finally G_8 can be obtained by contracting d and d'.

Now the simulation of Edge Elimination II goes like Fig. 4(iii). From G_1 to G_4 is by Equality Introduction, G_2 to G_5 by Vertex/Edge Introduction, G_6 (and also $G_7 = G_6$) by Edge Elimination I. G_8 is obtained by Edge Elimination I and finally we get G_3 by Triangle Elimination. $\square$

Lemma 4 (Crossover Construction). *Crossover Gadget G_1 as shown in Fig. 5 can be constructed by $\mathcal{PHC}$.*

Proof. First we get $X(2)$ by Equality Introduction to K_4. Then G_3, G_4, G_6, G_8, G_9 are obtained from $X(2)$ by (after contracting c and f for G_3, G_6 and G_9) Vertex/Edge Introduction. For example, G_3 has a subgraph obtained by contracting c and f of $X(2)$. Note that labels a to g are

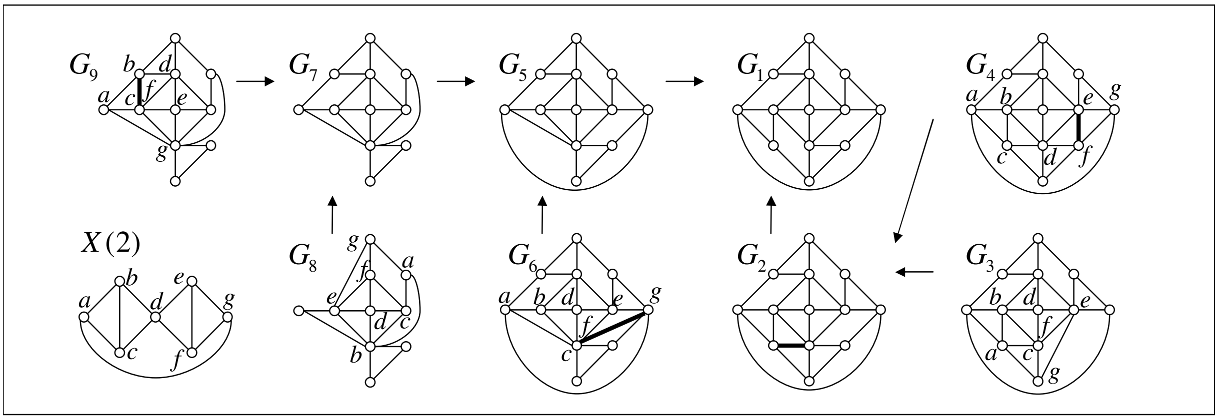

Figure 5: Crossover Construction

used to show corresponding vertices. All the remaining graphs are obtained by Edge Elimination II which can now be used by Lemma 3. For example, we get G_2 from G_3 and G_4 since G_3 is a graph obtained by contracting e and f of G_4 (edge (e,f) of G_4 is given as a bold line in the figure and similarly for the others). □

Lemma 5 (Crossover Introduction). *As Equality Introduction, a Crossover Gadget can be added. See Fig. 6(i).*

Proof. From G_1 to G_4, we just use Vertex/Edge Introduction (the added part is a Crossover Gadget whose two opposite vertices are merged). G_3 is by Crossover Construction that is possible by Lemma 4. Use just Vertex/Edge Introduction to make G_5 similar to the whole underlying graph. Finally G_2 is obtained by Edge Elimination II. □

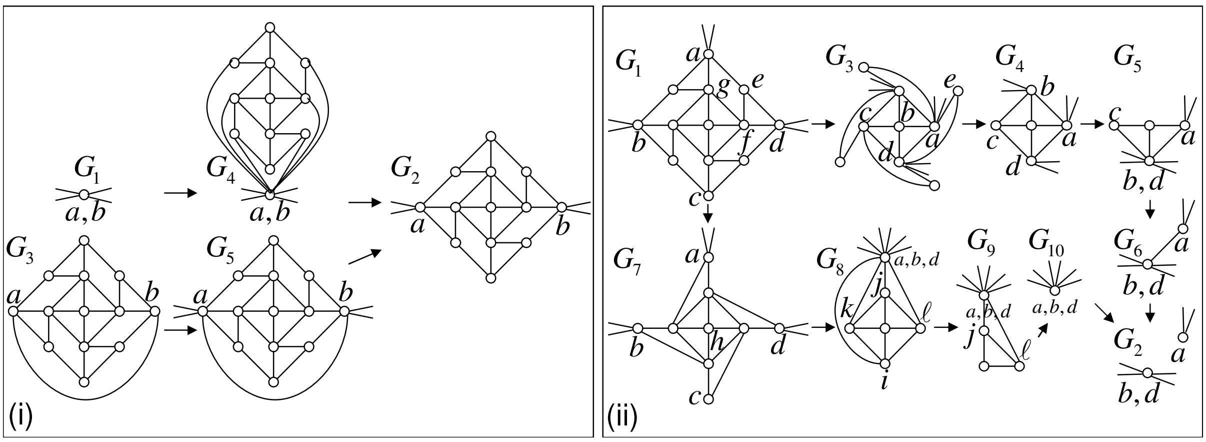

Figure 6: (i)Crossover Introduction (ii)Crossover Elimination

Lemma 6 (Crossover Elimination). *Let a, b, c and d be four outer vertices of a Crossover Gadget and b and d be opposite. Moreover c is free, i.e., c is not connected to any vertices except those in the Crossover Gadget. Then this Crossover Gadget can be removed, i.e., b and d are merged* *into a single vertex, a also remains, but all the other vertices and edges of the Crossover Gadget can be removed in polynomial steps. Namely, G_1 is changed to G_2 in Fig. 6(ii).*

Proof. Contracts vertices a and f (and three others similarly) to get G_3, and remove triangles to get G_4. Contract b and d (this is possible since c has no edges other than the three edges of the gadget). Two Triangle Eliminations to get G_6. As a different direction from the original graph, merge e and g (and three others) to get G_7, and contract c to h, b to a and d to a to get G_8. G_9 is obtained by applying two Contractions, i and j and k and l, G_{10} is by Triangle Elimination. Finally use Edge Elimination II from G_6 and G_{10} to G_2. □

Now we are ready to prove Lemma 1.

Proof of Lemma 1. Let G_1 and G_2 be two drawings of the same graph G. We are going to show that $P(G_1) \overset{*}{\Rightarrow} P(G_2)$ can be done (in polynomial steps) by the following algorithm. For exposition, we use the example in Fig. 7(i) (recall that a Crossover Gadget is represented by •). Note that vertices of the same label in $P(G_1)$ and $P(G_2)$ correspond to the same vertex of G.

Step 1 $P(G_2)$ is just added to $P(G_1)$ (by Vertex/Edge Introduction).

Step 2 Connect each pair of two vertices of the same label by using Crossover Gadgets as shown in Fig. 7(i). Let this new graph be G_3. Note that we may need two or more Crossover Gadgets to connect a single pair of vertices to maintain newly created crossings but it is easily seen that we can bound the total number of those Crossovers by a polynomial in $|P(G_1)| + |P(G_2)|$. Each vertex label in $P(G_1)$ is changed from ℓ to ℓ' (a to a', b to b', etc., as in the Figure).

Step 3 We now delete all the edges of $P(G_1)$ one by one: Suppose that we want to delete edge (b', c'). Then all we have to do is to create a graph which is exactly the same as G_3 except that vertices b' and c' are contracted (and then Edge Elimination II can be used to remove the edge). To to so, consider the cycle consisting of E-edge (b, c), edge (b', c'), and Crossover Gadgets connecting b and b', and c and c' (Fig. 7(ii)(a)). Note that the cycle is "twisted" and one can easily see that at most one twist is enough for each cycle (The following procedure becomes easier if there is no twist).

Now see Fig. 7(ii)(b). Our goal is to construct G_3 with contracted b' and c'. We start with a planar graph in Fig. 7(ii)(d) consisting of a single Crossover Gadget (let its outer vertices be e, f, g and h, e and g and f and h are opposite) such that e and f are connected by a single edge and g and h are contracted. Obviously this graph is non-3-colorable, and it can be generated by $\mathcal{PHC}$ in finite steps. (See Fig. 8. G_1 is just by Crossover Construction. G_2 is obtained from G_1 by two contractions between b and c and d and c. G_3 is obtained from G_1 by contracting c and d and adding an edge (a, b). Note that labels a to d of G_1 are used to show corresponding vertices. Finally we get G_4, which is exactly the same graph in Fig. 7(ii)(d), from G_2 and G_3 by Edge Elimination II since G_2 and G_3 are the same graph if the bold (a, c) in G_3 is contracted.) We then insert two Crossover Gadgets at vertices e and f and get Fig. 7(ii)(e), which is exactly the same as (b). Now we add vertices and edges to make it the same as G_3 excepting the contracted b' and c'. Let this new graph be G_3' and apply Edge Elimination II to delete the edge (b, c) from G_3 as in Fig. 7(ii)(c).

Repeat this procedure to remove all the edges of $P(G_1)$ part. Thus we obtain the graph as in Fig. 7(iii)(a).

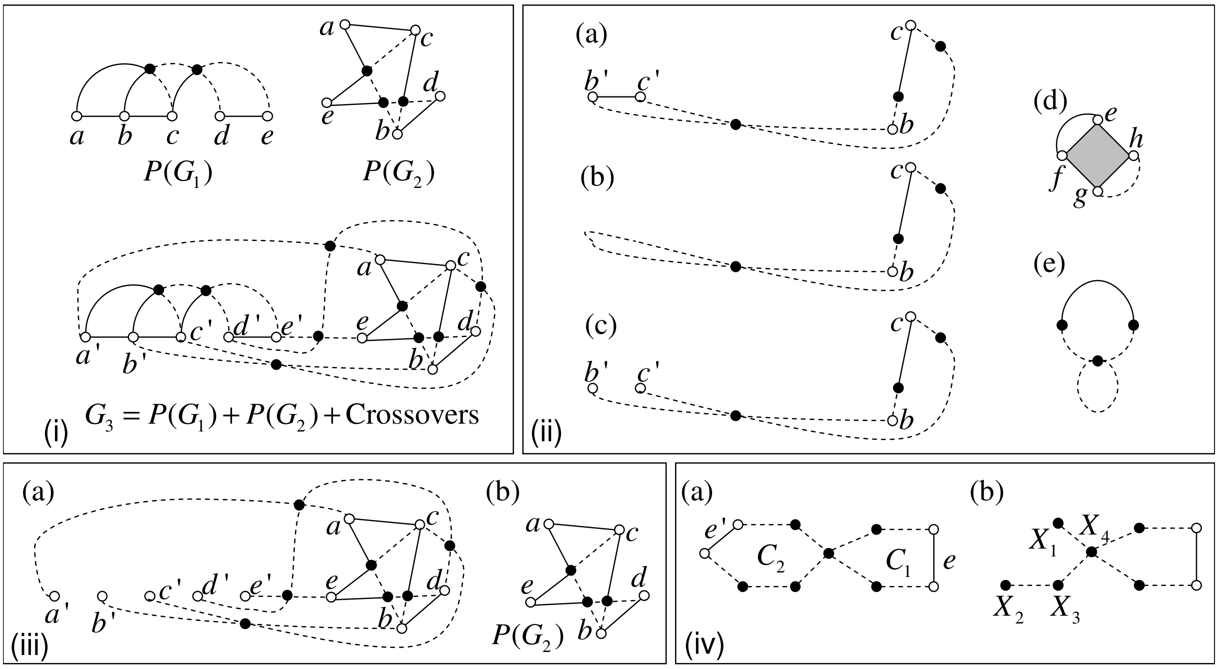

Figure 7: Redrawing

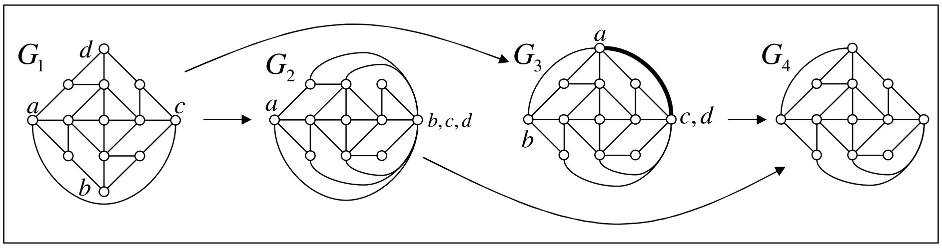

Figure 8: Construction of the Gadget in Fig. 7(ii)(d)

Step 4 Remove all the Crossover Gadgets excepting those within $P(G_2)$ to get Fig. 7(iii)(b). Recall that when we remove the Crossover Gadgets, one by one, we need to find a Crossover Gadget such that at least one of its outer vertices is free. To see this is always possible until all the Crossover Gadgets disappear, see the cycle as in Fig. 7(iv)(a). Note that the cycle is twisted and we can regard that it consists of two cycles, C_1 and C_2, each including an edge (e or e'). Suppose that edge e' is removed at step 3. Then the cycle C_2 is "cut", as shown in Fig. 7(b). Thus Crossover Gadgets X_1 and X_2 have free outer vertices and can be removed. Then X_3 has a free vertex and is removed. Then X_4 can be removed and the second cycle C_1 is also cut and Crossover Gadgets included this cycle can also be removed similarly.

This complete the proof for $P(G_1) \overset{*}{\Rightarrow} P(G_2)$. It is not hard to see that the procedure needs only polynomial steps. □

6 Main Theorem

We are now ready to prove our main theorem.

Theorem 1. *$\mathcal{PHC}$ is polynomially bounded iff so is $\mathcal{HC}$.*

Proof. We first prove the if-part. Suppose that $\mathcal{HC}$ is polynomially bounded for any (non-3-colorable) graph. Then it is obviously polynomially bounded for any (non-3-colorable) planar graph G. Hence there is a sequence of (not necessarily planar) graphs

$$G_1, G_2, \ldots, G_m = G$$

such that each G_i is (i) K_4 or (ii) for some $j < i$, G_i is generated from G_j by Rule 1 (Vertex/Edge Introduction) or Rule 3 (Contraction) of $\mathcal{HC}$ or (iii) for some $j, k < i$, G_i is generated from G_j and G_k by Rule 4 (Edge Elimination I) of $\mathcal{HC}$, all in time polynomial in $|G|$. For this sequence of graphs, we prove that there exists a sequence of drawings

$$H_1, H_2, \ldots, H_m, H$$

such that:

(i) H_i is a (maybe non-planar) drawing of G_i and H is an arbitrary planar drawing of G.

(ii) For each $1 \le i \le m$, $K_4 \overset{*}{\Rightarrow} P(H_i)$ or for some $j < i$, $P(H_j) \overset{*}{\Rightarrow} P(H_i)$ or for some $j, k < i$, $P(H_j), P(H_k) \overset{*}{\Rightarrow} P(H_i)$, all in polynomial steps. Here, "polynomial" means polynomial in $|P(H_j)| + |P(H_k)|$, which also means polynomial in $|G|$ since $|P(H_i)|$ is bounded by a polynomial in $|G_i|$ for all i and $|G_i|$ is bounded by a polynomial in $|G|$ by assumption.

(iii) $P(H_m) \overset{*}{\Rightarrow} H$ in polynomial (the same as above) steps.

Now we shall prove that for each G_i and G, there exists the corresponding H_i and H that satisfy these three conditions by induction, which obviously means that any non-3-colorable planar graph (G) can be generated by $\mathcal{PHC}$ in a polynomial number of steps. If $i = 1$, then G_1 must be a K_4. Then we can select H_1 as the planar drawing of K_4, and obviously $K_4 \overset{*}{\Rightarrow} P(H_1)$ in 0 steps.

For G_i ($i \ge 2$), there are several cases:

Case 1 G_i is a K_4. Completely the same as above.

Case 2 G_i is obtained from G_j ($j < i$) by Vertex/Edge Introduction. By induction hypothesis H_j is a proper drawing of G_i. To add an vertex, just add one in anywhere H_j to obtain H_i, which is obviously a proper drawing of G_i and satisfies the three conditions. If an edge is added between v_1 and v_2 of G_j, then we draw an edge between the corresponding vertices of H_i, which is also a proper drawing of G_i. For $P(H_i)$ we may need to add Crossover Gadgets along the added edge. The number of such Crossover Gadgets is at most the number of already existing (E-)edges and thus a polynomial number of steps suffice for $P(H_j) \overset{*}{\Rightarrow} P(H_i)$.

Case 3 G_i is obtained from G_j ($j < i$) by contracting two vertices, v_1 and v_2. To obtain H_i, we just "drag" v_1' to v_2', where v_1' and v_2' correspond to v_1 and v_2 of G_j, respectively. For $P(H_j) \overset{*}{\Rightarrow} P(H_i)$, see Fig. 9(i). Again we drag v_i' into the face v_2' is on in $P(H_j)$, where we may need to add (at most a polynomial number of) Crossover Gadgets as shown in Fig. 9(i). After that the two vertices are contracted in a single step. Thus the whole $P(H_j) \overset{*}{\Rightarrow} P(H_i)$ needs polynomial steps.

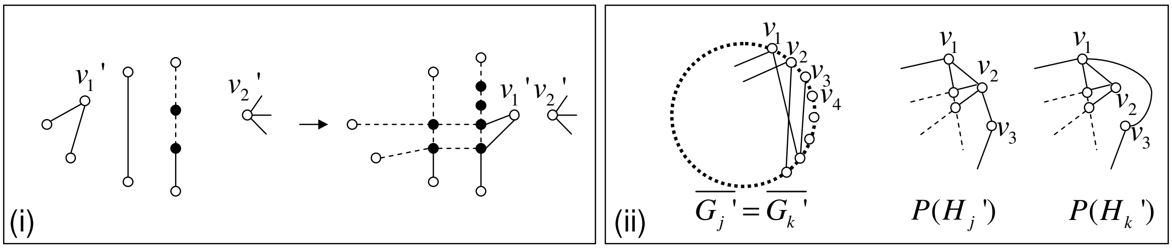

Figure 9: (i)Case 3 (ii)Case 4

Case 4 G_i is obtained from G_j and G_k ($j, k < i$) by Edge Elimination I. Let v_1, v_2 and v_3 be important vertices such that edge (v_1, v_2) exists both in G_j and G_k, edge (v_2, v_3) only in G_j, edge (v_1, v_3) only in G_k. All the other parts of G_j and G_k are the same. Let G'_j (G'_k, respectively) be the graph obtained from G_j (G_k, respectively) by removing the above two edges (v_1, v_2) and (v_2, v_3) ((v_1, v_2) and (v_1, v_3), respectively). By definition, G'_j and G'_k are the same graph and have the same drawing $\bar{G}'_j$ and $\bar{G}'_k$. This uniqueness of the drawing is important when we handle $P(H'_j)$ and $P(H'_k)$ later, and for such a drawing, we can use for instance the following method. The vertices are placed on a circle in the clockwise order of $v_1, v_2, v_3, \ldots, v_n$, and each edge is drawn as a straight line (See Fig. 9(ii)).

Now we put the removed two edges back to each of $\bar{G}'_j$ and $\bar{G}'_k$, obtaining H'_j and H'_k, where (v_1, v_2) and (v_2, v_3) are drawn as straight lines, but (v_1, v_3) is drawn as going around the outside of v_2 without any crossings. Their planarization $P(H'_j)$ and $P(H'_k)$ are given in Fig. 9(ii). Apparently H_j and H'_j are drawings of the same graph G_j and so are H_k and H'_k. Hence, by Lemma 1, $P(H_j) \overset{*}{\Rightarrow} P(H'_j)$ and $P(H_k) \overset{*}{\Rightarrow} P(H'_k)$, both in polynomial steps. Since $P(H'_j)$ and $P(H'_k)$ are exactly the same graph excepting edge (v_2, v_3) in $P(H'_j)$ and (v_2, v_3) in $P(H'_k)$, we can apply Edge Elimination I to get the graph $P(H_i)$. Because of the drawing rule above mentioned, we can determine H_i from $P(H_i)$ uniquely, which is obviously a drawing of G_i.

Case 5 Deriving of H from $P(H_m)$. Recall that H is a planar drawing of G and H_m is a (possibly non-planar) drawing of G_m, but since G_m and G are the same graph, H and H_m are drawing of the same graph. Thus we can use Lemma 1, i.e., $P(H_m) \overset{*}{\Rightarrow} H$ in polynomial steps. This completes the proof of the if-part.

The proof of the only-if part is easier but rather technical. Suppose that $\mathcal{PHC}$ is polynomially bounded. Let G be any (possibly non-planar) non-3-colorable graph and we denote its reasonable (without too many crossings) drawing also by G. Then the size of $P(G)$ is bounded by a polynomial and it can be generated by $\mathcal{PHC}$ in polynomial steps. In order to show that $\mathcal{HC}$ is polynomially bounded, it now suffices to show that G can be derived from $P(G)$ by $\mathcal{HC}$ in polynomial steps. Note that this is nothing other than a sequence of Crossover Eliminations. See Fig. 10(i): G_1 is a Crossover Gadget we want to remove. G_3 is obtained by Contractions of a and c, b and d and pairs of vertices labeled by s, t, v, w (recall we do not have to preserve planarity). G_4 is by Triangle Elimination (we need a care as mentioned below). G_5 and G_7 are by Contractions of b, d and a, c, and s and b, d, respectively. G_6 and G_8 are both by sequences of triangle Eliminations. Finally, G_2 is by Edge Elimination II.

Recall that the previous proof for Triangle Elimination needed the fact that any non-3-colorable planar graph has a triangle as a subgraph. In the above derivation, we cannot use this property since the graph may no longer be planar. So, in the following, we redesign the procedure for

Triangle Elimination by assuming that the graph includes a chord-less cycle of odd length. (Any non-3-colorable graph has such a cycle since otherwise the graph is bipartite.) See Fig. 10(ii). By using the same procedure as before, we can make a triangle *cde* and a "shaft" *abc* which connects the triangle and the odd cycle. Our goal is to remove this triangle and shaft. Recall that we can change the length of shaft arbitrarily.

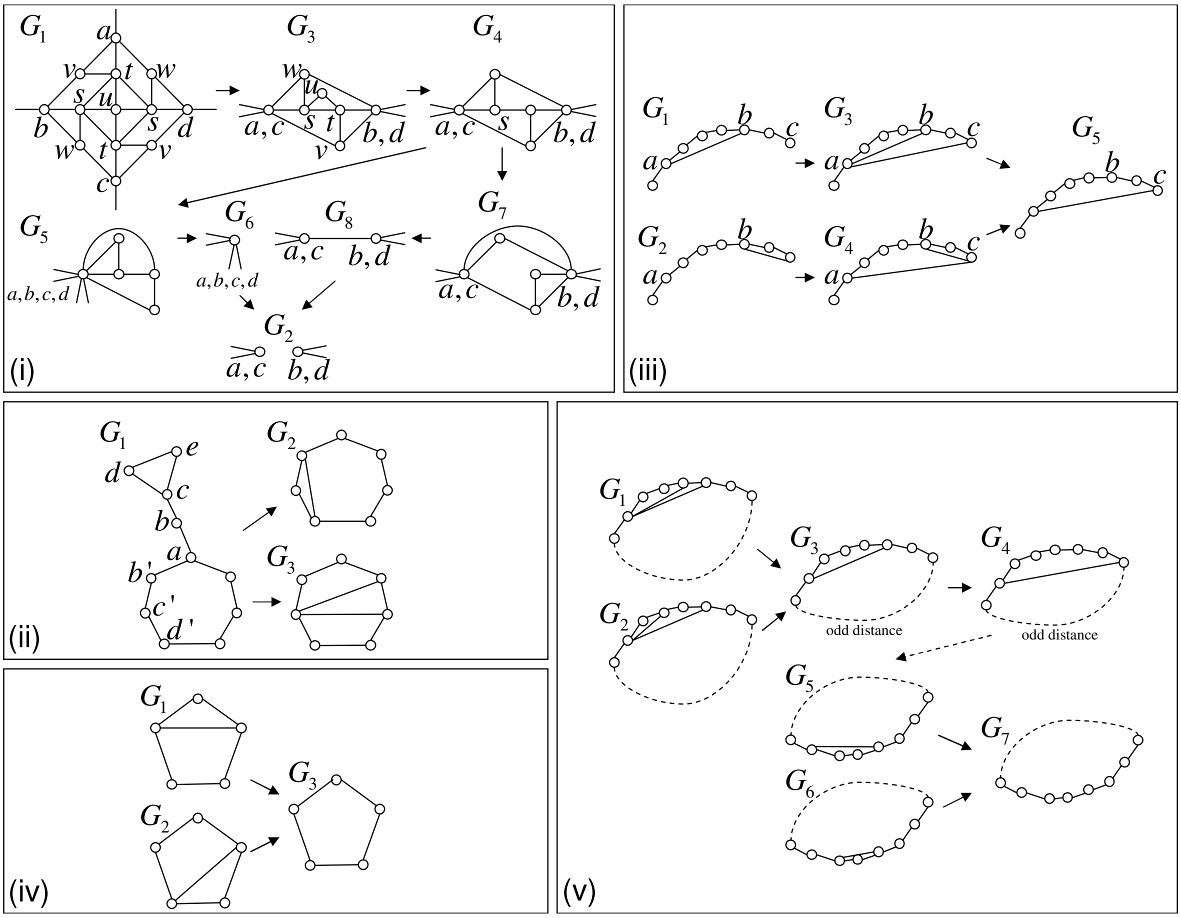

Figure 10: Crossover Elimination and Triangle Elimination

We have three basic operation: (i)*Chord of size three*(*3-chord*). As shown in Fig. 10(ii), we can replace the triangle and shaft by a chord which connects two cycle vertices of distance two (as in G_2). This can be done by, for instance, contracting b and b', c and c', d and d', and e and b'. (ii)*Inner triangle.* As shown in G_3, we can replace the triangle and chord by a inner triangle consisting of one cycle edge + two chords by a procedure similar to (i). (iii)*Chord Shift.* See Fig. 10(iii). Suppose that the triangle and shaft is replaced by chord ab (G_1). Then we also apply 3-Chord to the original graph and get G_2. G_3 and G_4 are obtained by Vertex/Edge Introduction from G_1 and G_2 respectively. Then Edge Elimination I from G_3 and G_4, we can get G_5 where the one endpoint of the chord is "shifted" two positions on the cycle.

Now the triangle and shaft can be removed as follows: If the cycle is a triangle then we are done as before. If the cycle is of size five, then see Fig. 10(iii). By 3-chord, we can make G_1 and G_2, followed by Edge Elimination I. Suppose that the cycle is of size seven or more. See Fig. 10(v). G_1 is obtained by Inner Triangle, where two chords connect vertices of distance three and distance four, and G_2 by 3-Chord + Edge Addition. G_3 is by Edge Elimination I and G_4 by Chord Shift.

Notice that in G_3 the chord connects two vertices whose lower-half distance is odd and this is also true in G_4. Repeating Chord Shift, we can reach, from the original graph, G_5 where the chord connects two cycle vertices of distance three. G_6 is obtained by 3-Chord and finally G_7 is obtained by Edge Elimination I.

Thus Triangle Elimination is still possible for non-planar non-3-colorable graphs, completing the proof of the only-if part. □

If we allow arbitrary steps for generation, the above proof claims that if a planar non-3-colorable graph G is generated by $\mathcal{HC}$, then so is by $\mathcal{PHC}$. Since the former is complete, we have the following theorem:

Theorem 2. *$\mathcal{PHC}$ is complete.*

7 Concluding Remarks

Recall that our final goal is to find a hard example for $\mathcal{PHC}$. Note that if the generation system is more deterministic, or application of each rule is more restricted, then it is usually better to prove lower bounds. In this sense, we should seek even more restricted graph calculus whose complexity is p-equivalent to that of $\mathcal{PHC}$. We already have candidates, for example, generation systems for degree-restricted non-3-colorable planar graphs.

References

[1] M. Ajtai. The complexity of the pigeonhole principle. *Combinatorica*, 14(4):417–433, 1994.

[2] K. Appel and W. Haken. Every planar map is four colorable. Part I. Discharging. *Illinois J. Math.*, 21:429–490, 1977.

[3] K. Appel, W. Haken and J. Koch. Every planar map is four colorable. Part II. Reducibility. *Illinois J. Math.*, 21:491–567, 1977.

[4] S. R. Buss, D. Grigoriev, R. Impagliazzo and T. Pitassi. Linear gaps between degrees for the polynomial calculus modulo distinct primes. *J. Comput. Syst. Sci.*, 62(2):267–289, 2001.

[5] P. Beame and T. Pitassi. Propositional proof complexity. Past, present, and future. *Bulletin of the European Association for Theoretical Computer Science*, 65:66–89, 1998.

[6] S. A. Cook and R. A. Reckhow. The relative efficiency of propositional proof systems. *J. Symb. Log.*, 44(1):36–50, 1979.

[7] S. Dash. An exponential lower bound on the length of some classes of branch-and-cut proofs. *Mathematics of Operations Research*, 30(3):678–700, 2005.

[8] Z. Galil. On the complexity of regular resolution and the Davis-Putnam procedure. *Theor. Comput. Sci.*, 4(1):23–46, 1977.

[9] M. R. Garey and D. S. Johnson. Computers and Intractability. A Guide to the Theory of NP-Completeness. W. H. Freeman and Company, New York, 1978.

[10] M. R. Garey, D. S. Johnson and L. J. Stockmeyer. Some simplified NP-complete graph problems. *Theoretical Computer Science*, 1(3):237–267, 1976.

[11] M. Grigni, E. Koutsoupias and C. H. Papadimitriou. An approximation scheme for planar graph TSP. In *Proc. of FOCS 1995*, pp. 640–645, 1995.

[12] H. Grötzsch. Ein Dreifarbensatz für dreikreisfreie Netze auf der kugel. *Wiss. Z. Martin Luther Univ. Halle-Wittenberg, Math. Nat. Reihe*, 8:109–120, 1959.

[13] G. Hajós. Über eine Konstruktion nicht n-färbbarer Graphen. *Wissenschaftliche Zeitschrift der Martin-Luther-Universität Halle-Wittenberg*, A 10:116–117, 1961.

[14] A. Haken. The intractability of resolution. *Theor. Comput. Sci.*, 39:297–308, 1985.

[15] J. E. Hopcroft and J. K. Wong. Linear time algorithm for isomorphism of planar graphs (Preliminary Report). In *Proceedings of the 6th annual ACM symposium on Theory of Computing*, pp. 172–184, 1974.

[16] K. Iwama and T. Pitassi. Exponential lower bounds for the tree-like Hajós calculus. *Information Processing Letters*, 54(5):289–294, 1995.

[17] J. Krajíček. Bounded Arithmetic, Propositional Logic, and Complexity Theory. Cambridge University Press, New York, 1995.

[18] J. Krajíček, P. Pudlák and A. Woods. Exponential lower bounds to the size of bounded depth Frege proofs of the pigeonhole principle. *Random Structures and Algorithms*, 7(1):15–40, 1995.

[19] R. J. Lipton and R. E. Tarjan. Applications of a planar separator theorem. *SIAM Journal on Computing*, 9(3):615–627, 1980.

[20] T. Pitassi, P. W. Beame and R. Impagliazzo. Exponential lower bounds for the pigeonhole principle. *Computational Complexity*, 3:97–140, 1993.

[21] T. Pitassi and A. Urquhart. The complexity of the Hajós calculus. *SIAM Journal on Discrete Mathematics*, 8(3):464–483, 1995.

[22] P. Pudlák. Lower bounds for resolution and cutting planes proofs and monotone computations. *J. of Symb. Logic*, 62(3):981–998, 1997.

[23] P. Pudlák. The lengths of proofs. In S. R. Buss (ed.), *Handbook of Proof Theory*, Elsevier, pp. 547–637, 1998.

[24] A. A. Razborov. Lower bounds for propositional proofs and independence results in bounded arithmetic. In *Proc. of ICALP 1996*, pp. 48–62, 1996.

[25] N. Segerlind. The complexity of propositional proofs. *The Bulletin of Symbolic Logic*, 13(4):482–537, 2007.

[26] C. S. Tseitin. On the complexity of derivation in the propositional calculas. In A. O. Slisenko (ed.), *Studies in Constructive Mathematics and Mathematical Logic, Part II*, 1968.

[27] A. Urquhart. The complexity of propositional proofs. *Bulletin of Symbolic Logic*, 1(4):425–467, 1995.