Maximum matching in graphs with an excluded minor

Raphael Yuster [*] *Uri Zwick* [†]

Abstract

We present a new randomized algorithm for finding a maximum matching in *H-minor free* graphs. For every fixed H, our algorithm runs in $O(n^{3\omega/(\omega+3)}) < O(n^{1.326})$ time, where n is the number of vertices of the input graph and $\omega < 2.376$ is the exponent of matrix multiplication. This improves upon the previous $O(n^{1.5})$ time bound obtained by applying the $O(mn^{1/2})$-time algorithm of Micali and Vazirani on this important class of graphs.

For graphs with *bounded genus*, which are special cases of H-minor free graphs, we present a randomized algorithm for finding a maximum matching in $O(n^{\omega/2}) < O(n^{1.19})$ time. This extends a previous randomized algorithm of Mucha and Sankowski, having the same running time, that finds a maximum matching in a *planar* graphs.

We also present a deterministic algorithm with a running time of $O(n^{1+\omega/2}) < O(n^{2.19})$ for *counting* the number of *perfect* matchings in graphs with bounded genus. This algorithm combines the techniques used by the algorithms above with the counting technique of Kasteleyn. Using this algorithm we can also count, within the same running time, the number of *T-joins* in planar graphs. As special cases, we get algorithms for counting Eulerian subgraphs ($T = \phi$) and odd subgraphs ($T = V$) of planar graphs.

1 Introduction

A *matching* in a graph is a set of pairwise disjoint edges. A *perfect matching* in a graph with n vertices is a matching of size $n/2$, and a *maximum matching* is a matching of largest possible size. The problems of finding a maximum matching, and of counting the number of perfect matchings are fundamental in both practical and theoretical computer science.

The first polynomial time algorithm for finding a maximum matching in a general graph was obtained by Edmonds [Edm65]. The currently fastest deterministic algorithms for this problem run in $O(mn^{1/2})$ time (see [MV80, Blu90, Vaz94, GT91]), where m and n are the number of edges and vertices, respectively, in the input graph. For dense graphs, better *randomized* algorithms are known. Lovász [Lov79] showed that the cardinality of a maximum matching can be determined, with high probability, by computing the rank of a matrix. In particular, checking whether a graph has a perfect matching amounts to checking whether a determinant of a certain matrix, whose construction involves randomization, is nonzero. This randomized algorithm can be implemented to run in $O(n^{\omega})$ time, where ω is the exponent of fast matrix multiplication. Coppersmith and Winograd [CW90] showed that $\omega < 2.376$. Recently, Mucha and Sankowski [MS04b] solved a long standing open problem and showed that a maximum matching can be *found*, with high probability, in $O(n^{\omega})$ time.

For graphs with $m = \Theta(n)$, the algorithms of [MV80, Blu90, Vaz94, GT91] run in $O(n^{1.5})$ time. For *planar* graphs an improved randomized algorithm with a running time of $O(n^{\omega/2}) < O(n^{1.19})$ was obtained recently by Mucha and Sankowski [MS04a]. They use their Gaussian elimination technique from [MS04b], as well as the *nested dissection* technique of Lipton, Rose and Tarjan [LRT79], which relies, in turn, on the *planar separator theorem* of Lipton and Tarjan [LT79].

The classical Kuratowski-Wagner Theorem [Kur30, Wag37] states that a graph is planar if and only if it has no K_5 nor $K_{3,3}$ minors. (For three different proofs of the theorem, see [Tho81].) A graph G' is a *minor* of a graph G if G' can be obtained from a subgraph of G by contracting edges. A graph is *H-minor free* if H is not a minor of G. An H-minor free graph is also said to have *excluded* H-minor.

Families of graphs with an excluded H-minor, for some fixed graph H, are the cornerstone of the seminal theory of *graph minors* developed over the last 20 years, in a series of more than 20 papers, by Robertson and Seymour. These families are, to date, the most studied families of graphs in modern graph theory. The graph minor theory of Robertson and Seymour culminated, in [RS04], with a proof of the profound *graph minor theorem*, also known as the *Wagner's conjecture*, that states that in every infinite set of finite graphs, there

*Department of Mathematics, University of Haifa, Haifa 31905, Israel. E–mail: raphy@math.haifa.ac.il

†School of Computer Science, Tel Aviv University, Tel Aviv 69978, Israel. E–mail: zwick@cs.tau.ac.il. Research supported by a GIF grant.

is a graph which is isomorphic to a minor of another. One of the consequences of this theorem is that for any surface S there is a finite set of graphs $F(S)$, such that a graph can be embedded in S (without crossing edges) if and only if it does not contain a graph from $F(S)$ as a minor. (This result actually follows from a restricted version of Wagner's conjecture which was already proved in [RS90].) For a very recent survey of the theory of graph minor see Lovász [Lov06].

The *genus* of a surface in R^3 is the largest number of non intersecting simple closed curves that can be drawn on the surface without separating it. The sphere is the simplest nontrivial surface in R^3. It has genus 0. The *genus* of a graph is the smallest integer g such that the graph can be embedded in an (orientable) surface of genus g, without edge crossings. Planar graphs are therefore of genus 0. Genus 1 graphs are graphs that can be embedded on the *torus*. By the results of Robertson and Seymour stated above, graphs of genus g are exactly the graphs which have no minor in a finite family $F = F_g$ of graphs. In particular, they are a subset of the class of H-minor free graphs, for every $H \in F$.

Classes of H-minor free graphs are much more general, however, than the classes of genus g graphs. For the example, the class of K_5-free graphs contains all the planar graphs, and many other graphs, but there is no bounded genus surface on which all the graphs from this family can be embedded.

The question we try to answer in this paper is the following: Can the algorithm of Mucha and Sankowski [MS04a] be extended to find maximum matchings in graphs of genus g? Can it be made to work for H-minor free graphs, for any fixed graph H?

Modifying the algorithm of [MS04a] to work for graphs of genus g, without increasing the running time, turns out to be a relatively straightforward task. The following theorem is proved in Section 3:

THEOREM 1.1. *Let $g \geq 0$ be a fixed integer. There exists an $O(n^{\omega/2}) < O(n^{1.19})$ time algorithm that, given a graph G on n vertices, either certifies that the genus of G is greater than g, or else finds, with high probability, a maximum matching in G.*

Extending the algorithm of Mucha and Sankowski [MS04a] to work on H-minor free graphs, for every fixed graph H, is a much more demanding task. We cannot retain the original $O(n^{\omega/2})$ running time the algorithm, but we can still get the following non-trivial result which is the main result of this paper:

THEOREM 1.2. *For every fixed graph H, there is an $O(n^{3\omega/(\omega+3)}) < O(n^{1.326})$ time algorithm that given a graph G on n vertices, either certifies that G has an H-minor, or else finds, with high probability, a maximum matching in G.*

The new algorithm uses, of course, the Gaussian elimination technique of Mucha and Sankowski [MS04a, MS04b], the nested dissection technique of Lipton, Rose and Tarjan [LRT79], and a recent algorithm of Reed and Wood [RW05] for finding *separators* in H-minor free graphs, which in turn, uses a previous algorithm for the problem obtained by Alon, Seymour and Thomas [AST90]. These powerful tools on their own, however, are not enough. The new algorithm also relies on a new technique that enables us to obtain small separators of certain graphs that are obtained from H-minor free graph by *vertex splitting*, even though these graphs are *not* necessarily H-minor free. A detailed description of the algorithm, and a proof of Theorem 1.2, appear in Section 2.

Perfect matching, if they exist, can be found in polynomial time. *Counting* the number of perfect matching, in contrast, is a #P-complete problem, as shown by Valiant [Val79]. Perfect matchings in *planar* graphs can, however, be counted, using the *Pfaffian orientations* technique of Kasteleyn [Kas67]. Galluccio and Loebl [GL99a, GL99b] extended the result of Kasteleyn [Kas67] and showed that perfect matchings in genus g graphs, for every fixed g, can also be counted in polynomial time, even though genus g graphs do not necessarily have Pfaffian orientations.

By combining the algorithm of [MS04a] with the *Pfaffian orientations* technique of Kasteleyn [Kas67] and its extensions by Galluccio and Loebl [GL99a, GL99b] we obtain an efficient algorithm for counting the number of perfect matchings in genus g graphs.

THEOREM 1.3. *Let $g \geq 0$ be a fixed integer. There exists an $\tilde{O}(n^{1+\omega/2}) < O(n^{2.19})$ time algorithm that, given a graph G on n vertices, either certifies that the genus of G is greater than g, or else returns the number of perfect matchings in G.*

This, in particular, improves on a $\tilde{O}(n^{2.5})$-time algorithm for counting perfect matching in planar graphs given by Wilson [Wil97]. Wilson uses his algorithm to generate *random* perfect matchings of planar graphs. The proof of Theorem 1.3 appears in Section 4.

Finally, by utilizing a reduction obtained by Galluccio and Loebl [GL99b] the algorithm for counting perfect matchings can also be used to count T-*joins* of bounded genus graphs. Let $G = (V, E)$ be a graph and $T \subseteq V$. A T-join of the graph G is a subset $E' \subseteq E$ of the edges such that the degree of a vertex v in the subgraph $G' = (V, E')$ is *odd* if and only if $v \in T$. If $T = \phi$, then a T-join is an Eulerian subgraph of G, i.e., a spanning subgraph in which the degrees of all the vertices are even. If $T = V$, then a T-join is an odd subgraph of G, i.e., a subgraph in which the degrees of all vertices are odd. (Such a subgraph may exist, of course, only if $|V|$ is even.) In particular, we get an $\tilde{O}(n^{1+\omega/2}) < O(n^{2.19})$ algorithm for counting the number of Eulerian subgraphs, and odd subgraphs of a planar graph. We also get an $\tilde{O}(n^{2+\omega/2}) < O(n^{3.19})$ algorithm for counting T-joins of a planar graph with a specified cardinality.

2 Maximum matching in H-minor free graphs

In this section we prove Theorem 1.2. In order to describe the algorithm proving Theorem 1.2 we need quite a few definitions and several tools that will also be useful in the rest of this paper.

2.1 Graph minors A graph H is a *minor* of a graph G if H can be obtained from a subgraph of G by contracting edges. When an edge uv is contracted, the vertices u and v are unified and the neighbors of the unified vertex is the union of the sets of neighbors of u and v. If H is not a minor of G we say that G is H-minor free, or, alternatively, that G has an excluded H-minor. For example, a graph is K_3-minor free if and only if it is a forest, and a graph is K_4-minor free if and only if it is a *series-parallel* graph. Planar graphs are the intersection of the family of K_5-minor free graphs and the family of $K_{3,3}$-minor free graphs.

An H-model in G is a set of vertex-disjoint connected subgraphs $\{X_v \; : \; v \in V(H)\}$ indexed by the vertices of H, such that for every edge $uv \in E(H)$, there is an edge $xy \in E(G)$ with $x \in X_u$ and $y \in X_v$. Clearly G has an H-minor if and only if G has an H-model.

2.2 Separator theorems We say that a graph $G = (V, E)$ has a (k, α)-*separation*, if V can be partitioned into three parts, A, B, C so that $A \cap B = \emptyset$, $|A \cup C| \le \alpha|V|$, $|B \cup C| \le \alpha|V|$, $|C| \le k$, and if $uv \in E$ and $u \in A$ then $v \notin B$. We say that A and B are separated by C, that C is a separator and that the partition (A, B, C) *exhibits* a (k, α)-separation.

By the seminal result of Lipton and Tarjan [LT79], a planar graph with n vertices has an $(O(\sqrt{n}), 2n/3)$-separation. In fact, they also show how to compute such a separation in linear time. Subsequently, Alon, Seymour and Thomas [AST90] extended the result of Lipton and Tarjan to H-minor free graphs. However, unfortunately, the running time of their algorithm is $O(n^{1.5})$ for every fixed H, which means that we cannot use it in our setting.

When the existence of an $(f(n), \alpha)$-separation can be proved for each n-vertex graph belonging to a hereditary family (closed under taking subgraphs), one can recursively continue separating each of the separated parts A and B until the separated pieces are small enough. This yields a separator tree. Notice that planarity, as well as being H-minor free, is a hereditary property. More formally, we say that a graph $G = (V, E)$ with n vertices has an $(f(n), \alpha)$-*separator tree* if there exists a full rooted binary tree T so that the following holds:

(i) Each $t \in V(T)$ is associated with some $V_t \subset V$.
(ii) The root of T is associated with V.
(iii) If $t_1, t_2 \in V(T)$ are the two children of $t \in V(T)$ then $V_{t_1} \subset V_t$ and $V_{t_2} \subset V_t$. Furthermore, if $A = V_{t_1}$, $B = V_{t_2}$ and $C = V_t \setminus (V_{t_1} \cup V_{t_2})$ then (A, B, C) exhibits an $(f(|V_t|), \alpha)$-separation of $G[V_t]$ (the subgraph induced by G_t).
(iv) If t is a leaf then $|V_t| = O(1)$.

By using divide and conquer, the result of Lipton and Tarjan mentioned above can be stated as follows (see also Gilbert and Tarjan [GT87] for a simplified version of their algorithm).

THEOREM 2.1. *A planar graph with n vertices has an $(O(\sqrt{n}), 2/3)$-separator tree and such a tree can be found in $O(n \log n)$ time.*

2.3 Gaussian elimination and nested dissection Let A be an $n \times n$ matrix. The *representing graph* of A, denoted $G(A)$ is defined by the vertex set $\{1, \dots, n\}$ where, for $i \ne j$ we have an edge ij if and only if $A_{i,j} \ne 0$ or $A_{j,i} \ne 0$.

Generalizing the nested dissection method of George [Geo73], Lipton, Rose and Tarjan [LRT79] and Gilbert and Tarjan [GT87] proved the following.

THEOREM 2.2. *Let A be a symmetric positive definite $n \times n$ matrix. If, for some positive constant $\alpha < 1$, and for some constant $\beta \ge 1/2$, $G(A)$ has bounded degree and an $(O(n^{\beta}), \alpha)$-separator tree, and such a tree is given, then Gaussian elimination on A can be performed with $O(n^{\omega\beta})$ arithmetic operations. The resulting LU factorization of A is given by matrices L and D, $A = LDL^T$, where L is unit lower-triangular and has $\tilde{O}(n^{2\beta})$ non-zero entries and D is diagonal.*

We note that the requirement that the graph $G(A)$ has bounded degree is not needed in Theorem 2.2 if we use a *strong* separator tree (in a strong separator tree the separator vertices at each node of the tree also appear in both children of the node). See [GT87] for a detailed description of both versions.

2.4 Skew-adjacency matrices Let $G = (V, E)$ be an undirected graph with $V = \{1, \dots, n\}$. With each edge $e \in E$ we associate a variable x_e. Let $\vec{G}$ be any orientation of G. Define the *skew adjacency matrix* $A_s(\vec{G})$ by

$$a_{ij} = \begin{cases} +x_e, & \text{if } e = ij \text{ and } (i,j) \in E(\vec{G}); \\ -x_e, & \text{if } e = ij \text{ and } (j,i) \in E(\vec{G}); \\ 0, & \text{otherwise.} \end{cases}$$

Lovász [Lov79] proved that the rank of $A_s(\vec{G})$ is twice the size of a maximum matching in G. This observation, together with some additional ideas, leads to an $O(n^\omega)$ time randomized algorithm for deciding whether a graph has a perfect matching. Generalizing the ideas from [Lov79], together with the nested dissection method, Mucha and Sankowski [MS04a] obtained the following result:

THEOREM 2.3. *Let G be a graph with n vertices and with maximum degree at most some constant k. Furthermore, suppose that for some constants $\alpha < 1$ and $\beta \geq 1/2$, G is equipped with an $(O(n^\beta), \alpha)$-separator tree. Then, a maximum matching in G can be found, with high probability, in $O(n^{\omega\beta})$ time.*

Although Mucha and Sankowski [MS04a] state their result only for planar graphs (and hence only use the case $\beta = 1/2$ of Theorem 2.2), the more general statement of Theorem 2.3 follows without change from their result. The idea of their proof is as follows. Suppose G satisfies the conditions of Theorem 2.3. Take an arbitrary orientation of G and construct (a sparse representation of) the resulting skew-adjacency matrix $A = A_s(\vec{G})$. Let $B = AA^T$. Notice that B is symmetric and if A is non-singular (namely, if G has a perfect matching), then B is positive definite. Since G has maximum degree at most k, it is straightforward that the representing graph $G(B)$ has an $(O(n^\beta), \alpha)$-separator tree (a separator of $G(B)$ corresponds to a separator of G by taking the separator vertices and their neighbors). Thus, one can apply Theorem 2.2 to $G(B)$. Mucha and Sankowski [MS04a] show that instead of doing the arithmetic operations in the field of rational functions with integer coefficients (which makes each arithmetic operation very expensive) it suffices to perform the operations over a finite field, sacrificing a deterministic algorithm for a randomized one, but now the bit complexity of each arithmetic operation is only $O(\log n)$. It is shown in [MS04b], in a highly non-trivial way, how to produce a maximum matching from the resulting LU factorization of $G(B)$.

2.5 Vertex splitting The maximum degree requirement in Theorem 2.3 is very limiting. There is a general technique that transforms every graph G to another graph G' so that the latter has maximum degree at most k, where $k \geq 3$, and so that the number of perfect matchings of G and G' is the same. Furthermore, there is an easy translation of maximum matchings in G to maximum matchings in G' and vice versa.

Suppose G has a vertex u of degree at least $k+1$. Pick two neighbors of u, say v, w. Add two new vertices u' and u'', add the edges $uu', u'u'', u''v, u''w$ and delete the original edges uv, uw. Clearly, this vertex-splitting operation does not change the number of perfect matchings, and increases the size of the maximum matching by 1. By repeatedly performing vertex splitting until there are no vertices with degree greater than k, we obtain a desired *vertex split* graph G'. See, e.g., Figure 1. Clearly, if G has n vertices and $O(n)$ edges, then G' has $O(n)$ vertices and $O(n)$ edges as well.

If G is a planar graph, it is easy to obtain a vertex-split G' of G so that G' is planar as well. Indeed, consider a plane embedding of G. Always take the neighbors v and w of u so that u, v, w are on the same face. This observation, together with Theorem 2.3 and Theorem 2.1 yield the randomized $O(n^{\omega/2})$ time algorithm for maximum matching in planar graphs.

2.6 The algorithm In order to apply Theorem 2.3 to H-minor free graphs, we need to overcome two major obstacles. The first one is that vertex splitting does not necessarily preserve H-minor freeness, as shown in Figure 2. Furthermore, unlike the planar case where there is an easy embedding argument that enables vertex splitting while maintaining planarity, there is no such argument for H-minor free graphs, and, in fact, it is simply **not true**. Consider an $n \times n$ grid and an additional vertex that is connected to all n^2 vertices of the grid. Clearly, this graph has no K_6-minor. However, for every positive integer k, if n is sufficiently large, performing vertex splitting on this graph in *any order* introduces a K_k-minor (we thank Robin Thomas for providing us with this example). The second obstacle comes from the requirement in Theorem 2.3 that the graph be equipped with a separator tree. One can use the algorithm of Alon, Robertson and Seymour [AST90], mentioned earlier, and create a separator tree for a given H-minor free graph. This, however, already takes more then $O(n^{1.5})$ time which, for the maximum matching problem, is already worse than the known $O(n^{1.5})$ time algorithms.

Our first lemma, which is interesting in its own right, overcomes the first obstacle. For a constant δ, we say that a graph is δ*-locally sparse* if every subgraph with x vertices induces at most δx edges. Notice that if all the graphs of a hereditary graph family are δ-sparse, then they are also δ-locally sparse. Thus, planar graphs are locally sparse (with $\delta = 3$), and so are H-minor

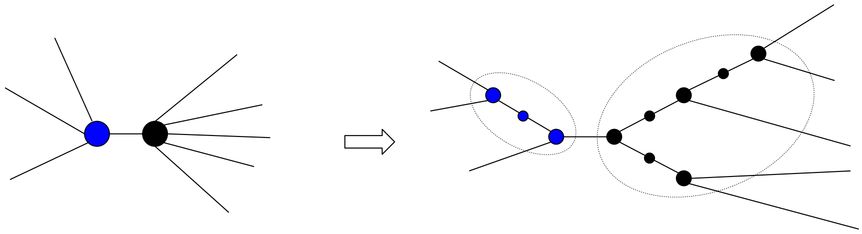

Figure 1: Vertex splitting.

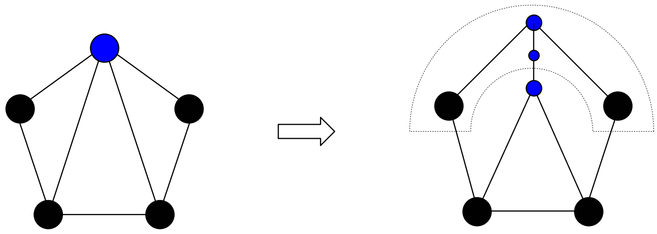

Figure 2: Vertex splitting can introduce K_4-minors.

free graphs, graphs with bounded genus, graphs with bounded degree and graphs with bounded degeneracy.

LEMMA 2.1. *Let ALG be an algorithm that given an n-vertex H-minor free graph, generates a partition (A, B, C) that exhibits an $(O(n^\beta), 2/3)$-separation in $O(n^\gamma)$ time (we assume $\gamma > 1$ and $\beta \geq 1/2$). Then, given an H-minor free graph G with n vertices, there is a vertex-split graph G' of G of bounded maximum degree so that G' has an $(O(n^\beta), \alpha)$-separator tree where $\alpha < 1$ is a constant that only depends on H. Furthermore, a separator tree for G' can be constructed in $O(n^\gamma)$ time.*

Proof. Let Q_k denote a rooted tree with maximum out-degree at most 2 obtained by vertex-splitting the star S_{k+1} with k edges. The root of Q_k is associated with the root of the star and each of the k leaves of Q_k is associated with a leaf of S_{k+1}. For $k \geq 2$ we always assume that the root has degree 2. Notice that $Q_0 = S_1$, $Q_1 = S_2$, $Q_2 = S_3$ but Q_3 already has 5 edges and, generally, Q_k has $3k - 4$ edges for $k \geq 2$.

We begin our algorithm by invoking algorithm *ALG* on $G = (V, E)$. In $O(n^\gamma)$ time we obtain a partition (A, B, C) that exhibits an $(O(n^\beta), 2/3)$-separation. Thus, in particular, $|C| = O(n^\beta)$. We create a new graph G_1 which is a (partial) vertex splitting of G. We replace each $v \in C$ with a copy of Q_3, denoted Y_v, and denote the three leaves with of Y_v with v_a, v_b, v_c and the root of Y_v with v. Notice that Y_v has five edges and six vertices. For each $u \in A$ so that $(u, v) \in E$, we connect u with v_a. Similarly, for each $u \in B$ so that $(u, v) \in E$, we connect u with v_b. Let $C(v)$ be the set of neighbors of v in C. We grow a copy of $Q_{|C(v)|}$ rooted at v_c, and label the leaves of this copy with v_u where $u \in C(v)$. Finally, for each original edge (u, v) where $u, v \in C$ we add the edge (u_v, v_u). This defines the new graph G_1 which, as can bee seen, is obtained from G via vertex splitting. In fact, we can save some new vertices so that no unnecessary vertex splitting ever occurs, as follows. If v has less than 3 neighbors in A we do not need v_a, we can connect them directly to v. Similarly for v_b and v_c.

For $v \in C$, let k_v denote the number of neighbors of v in C. Notice that G_1 has at most $n + 5|C| + 3\sum_{v\in C} k_v$ vertices. We now define a partition of the vertex set of G_1. We set $A_1 = A \cup \{v_a \ : \ v \in C\}$ (in fact, if v_a does not exist as noted in the above vertex-saving remark we do not take it into A_1). Similarly, $B_1 = B \cup \{v_b \ : \ v \in C\}$ and C_1 consists of all the remaining vertices. Namely, for each $v \in C$, all the vertices of Y_v except v_a, v_b (there are at most 4 such vertices) and all the vertices of the $Q_{|C(v)|}$ rooted at v_c all belong to C_1. Notice that $|C_1| \leq 4|C| + 3\sum_{v\in C} k_v$. But since G is δ-locally sparse (for some constant $\delta = \delta(H)$), the number of edges in the subgraph induced by $G[C]$ is at most $\delta|C|$. Thus, $|C_1| \leq 4|C| + 6\delta|C| \leq (4 + 6\delta)O(n^\beta)$. Notice also that no edge crosses from A_1 to B_1, and that $|A_1| \leq |A| + |C| \leq 2n/3$ and $|B_1| \leq |B| + |C| \leq 2n/3$.

Let G_A be the subgraph of G_1 induced by A_1 and let Let G_B be the subgraph of G_1 induced by B_1. Notice that both G_A and G_B are isomorphic to subgraphs of G and hence are H-minor free. We may therefore recursively apply our vertex-splitting algorithm to G_A and G_B respectively, resulting in a separator tree T_A for G'_A, the graph obtained from G_A after all the vertex splittings performed by the recursive calls, and a separator tree T_B for G'_B, the graph obtained from G_B after all the vertex splittings performed by the recursive calls. Both G'_A and G'_B have bounded degree. The overall vertex split of G, denoted G' is obtained by connecting the vertices of C', all of which have degree at most 4 in $G[C']$, to their neighbors in A_1 and B_1 respectively (notice that these neighbors retained their label also in G'_A and G'_B respectively).

Since we have made no redundant vertex splitting, the overall number of vertices in G' is at most Cn where $C = C(H)$ is a constant. The size of B_1 is at least $n/3$ and hence the size of G'_B is at least $n/3$. Thus, the size of G'_A is at most $|G'| - n/3 + O(n^\beta) < \alpha|G'|$ for a suitable constant $\alpha < 1$, and the same holds for the size of G'_B. The separator tree for G' is obtained by taking the disjoint trees T_A and T_B, and adding to them a common root associated with the entire vertex set of G'. Notice that the size of the initial separating set, C', is $O(n^\beta)$, as required. The standard analysis of the recursive calls to ALG yields an overall running time of $O(n^\gamma)$, as required.

The vertex splitting in Lemma 2.1 is performed inline with the recursive applications of ALG. This is *crucial*. Indeed, we could have obtained a separator tree for G by recursively applying ALG and only in the end perform the vertex splitting, but now a component of the origial tree could become very large. Indeed, if v is a vertex of some tree component C and v has $\Theta(n)$ neighbors in descendant components then vertex splitting would cause C to blow-up and become a component of size $\Theta(n)$.

In a recent result, Reed and Wood [RW05] generalize the result of Alon, Robertson and Seymour [AST90] in an interesting way. They show that a separator for an H-minor free graph can be found more quickly, if we are willing to get a somewhat larger separator. More precisely, they obtain the following result.

Theorem 2.4. *Let $\epsilon \in [0, 1/2]$ be fixed and let H be a fixed graph. There is an algorithm with running time $O(n^{1+\epsilon})$ that, given an n-vertex graph G, either outputs:* (a) *an H-model of G, or* (b) *a partition (A, B, C) that exhibits an $(O(n^{(2-\epsilon)/3}), 2/3)$-separation.*

Notice that the case $\epsilon = 1/2$ degenerates to the result of Alon, Seymour and Thomas [AST90] for fixed H. We may use the algorithm of Theorem 2.4 as ALG in Lemma 2.1. We therefore obtain the following corollary.

Corollary 2.1. *Let $\epsilon \in (0, 1/2]$ be fixed and let H be a fixed graph. There is an algorithm with running time $O(n^{1+\epsilon})$ that, given an n-vertex graph G, either outputs:* (a) *an H-model of G, or* (b) *a bounded degree vertex-split graph G' of G and an $(O(n^{(2-\epsilon)/3}), \alpha)$-separator tree for G where $\alpha = \alpha(H) < 1$.*

Completing the description of the algorithm: We choose $\epsilon = (2\omega - 3)/(3 + \omega)$. Given a graph G with n vertices, we apply Corollary 2.1. In $O(n^{3\omega/(3+\omega)})$ time we either find an H-model of G, certifying that G has an H-minor, or else we construct an $(O(n^{3/(3+\omega)}), \alpha)$-separator tree for a bounded degree vertex-split graph G' of G. Equipped with this separator tree we run the algorithm of Theorem 2.3 and obtain, with high probability, and in $O(n^{3\omega/(3+\omega)})$ time, a maximum matching of G', which we can directly translate to a maximum matching of G. □

3 Maximum matchings in bounded genus graphs

In order to apply Theorem 2.3 and vertex splitting to graphs with bounded genus, we need two results. The first is a generalization of Theorem 2.1 which was proved by Gilbert, Hutchinson and Tarjan [GHT84].

Theorem 3.1. *Let g be a fixed nonnegative integer. Given an embedding of a graph G with n vertices on a surface with genus g, an $(O(\sqrt{n}), 2/3)$-separator tree for G can be constructed in $O(n \log n)$ time.*

Notice, however, that the algorithm in Theorem 3.1 requires that the embedding be given. Only much later, Mohar [Moh99] showed that such an embedding can be found in linear time.

Theorem 3.2. *Let g be a fixed nonnegative integer. There exists a linear time algorithm that, for a given graph G, either finds an embedding of G in a fixed surface of genus g or determines that the genus of G is greater than g.*

We note that the embedding is purely combinatorial and is given by a *rotation system* (a cyclic permutation π_v of edges incident with v, representing their circular order around v on the surface; see, e.g., Mohar [Moh99] and the next subsection). Notice that once such an embedding is obtained, it is straightforward to perform vertex splitting while not increasing the genus, just as in the case of planar graphs.

Proof of Theorem 1.1: Give a graph G with n vertices, we first apply the algorithm of Theorem 3.2 and either obtain an embedding of G on a surface of genus g or a certificate showing that G has genus greater than g.

Once equipped with the embedding, we perform vertex splitting and obtain a graph G', with genus at most g, and with maximum degree at most 3. Next, we apply Theorem 3.1 to obtain an $(O(\sqrt{n}), 2/3)$-separator tree for G'. We then apply the algorithm of Theorem 2.3 and obtain a maximum matching of G' with probability at least 2/3, which is directly translated to a maximum matching of G. The overall running time is, therefore, $O(n^{\omega/2})$. □

4 Counting perfect matchings in bounded genus graphs

The theory of Pfaffian orientations was introduced by Kasteleyn [Kas67] to solve some enumeration problems arising from statistical physics. These orientations can be used to count perfect matchings in *Pfaffian orientable* graphs. Such graphs include all planar graphs, all $K_{3,3}$-free graphs (see [Lit74, Vaz89]), and many others.

In the rest of this section we follow the definitions from [Jer03]. Let $G = (V, E)$ be an undirected graph, C an even cycle in G, and $\vec{G}$ an orientation of G. We say that C is *oddly oriented by* $\vec{G}$ if, when traversing C in either direction, the number of co-oriented edges (i.e., edges whose orientation in $\vec{G}$ and in the traversal is the same) is odd.

Definition 4.1. *An orientation $\vec{G}$ of G is* Pfaffian *if the the following condition holds: for any two perfect matchings M, M' in G, every cycle in $M \cup M'$ is oddly oriented by $\vec{G}$.*

Note that all cycles in the union of two perfect matchings are even.

Let $G = (V, E)$ be an undirected graph with $V = \{1, \ldots, n\}$. With each edge $e \in E$ we associate a variable x_e. For $M \subset E$ let $x(M) = \prod_{e \in M} x_e$. The *matching polynomial* of G is $M(G) = \sum_M x(M)$ where the sum is taken over all perfect matchings of G. Recall the definition of the skew adjacency matrix $A_s(\vec{G})$ of an orientation $\vec{G}$ of G given in the previous section. Kasteleyn proved the following result (see, e.g., [Kas67, LP86, Jer03]).

Theorem 4.1. *For any Pfaffian orientation $\vec{G}$ of G,*

$$M(G) = \sqrt{det\ A_s(\vec{G})}\ .$$

In particular, if we assign $+1$ to all the variables we obtain that

$$\#\text{ perfect matchings in } G = \sqrt{det\ A(\vec{G})}$$

where $A(\vec{G})$ is the matrix obtained from $A_s(\vec{G})$ by assigning $+1$ to all the variables. Thus, once we are given a Pfaffian orientation, computing the number of perfect matchings reduces to computing a determinant. Not all graphs have a Pfaffian orientation. For example, $K_{3,3}$ does not have one. The computational complexity of deciding, for an arbitrary graph G, whether G has a Pfaffian orientation, is open. It is neither known to be in P nor to be NP-complete. In a seminal paper of Robertson, Seymour and Thomas [RST99], the restriction of this decision problem to bipartite graphs was shown to be in P. This was also proved independently by McCuaig [McC04].

It is not difficult to prove that a planar graph has a Pfaffian orientation. In fact, let G be a planar graph embedded in the plane. If G contains bridges, they may be oriented arbitrarily. Now consider the graph without its bridges. Each connected component is 2-edge connected. Suppose that we orient each connected component so that every face, except the (outer) infinite face, has an odd number of edges that are oriented clockwise. Then, the resulting orientation is Pfaffian (see, e.g., [Jer03] for a short proof). Hopcroft and Tarjan gave the first linear time algorithm that determines whether a graph can be embedded in the plane [HT74]. An extension of this algorithm given in [CNAO85], that also runs in linear time, returns an embedding in the form of a rotation system. Namely, for each vertex we are given the list of its neighbors in a clockwise ordering. Once equipped with a rotation system of a 2-edge connected planar graph, given any edge uv it is straightforward to traverse the two faces to which uv belongs. Start with $u = v_0$ and $v = v_1$, and for $i = 2, 3, \ldots$, let v_i be the vertex following v_{i-2} in the cyclic list of neighbors of v_{i-1}. Once returning to v_0 a face has been discovered. To discover the other face, start with $v = v_0$ and $u = v_1$.

Proposition 4.1. *Given a planar graph G with n vertices, a Pfaffian orientation of G can be constructed in $O(n)$ time.*

Proof. As noted earlier, it suffices to prove the proposition for 2-edge connected planar graphs, that are equipped with a rotation system. In particular, as shown above, we can list all the faces in linear time. Lovász and plummer [LP86] showed how to order the faces in linear time so that each face (except the last face, which is the outer face) contains at least one edge not appearing in earlier faces. This is done by constructing a spanning tree of the dual, rooted by the outer face, and listing the faces in reverse order of their distance from the root. Now, start with the first face, and orient all but one of its edges arbitrarily. Orient the last edge so that an odd number of edges are oriented clockwise. Now, continue to the next face. It has at least one edge that is still not oriented. Thus, we can orient all the remaining non-oriented edges on this face so that an odd number of edges are oriented clockwise. We continue in this manner until all faces, except the outer face, have an odd number of edges that are oriented clockwise. The resulting orientation is Pfaffian.

Proof of Theorem 1.3 for planar graphs: Given a graph G with n vertices, we can obtain, in $O(n)$ time, an embedding of G in the plane (see, e.g., [CNAO85, BM04]). If G is not planar, the algorithm detects this fact. Otherwise, we use the rotation system to obtain, in $O(n)$ time, a vertex split graph G' of G with maximum degree at most 3, so that G' is planar as well. Notice that the number of perfect matchings of G' and G is the same. Next, we apply Proposition 4.1 and obtain a Pfaffian orientation $\vec{G'}$ of G', in $O(n)$ time. Let $A = A(\vec{G'})$ be the matrix obtained from $A_s(\vec{G'})$ by assigning $+1$ to all the variables, and let $B = AA^T$. Using Theorem 2.1 we obtain an $(O(\sqrt{n}), 2/3)$-separator tree for G' in $O(n \log n)$ time. Since G' has maximum degree at most 3, we can use the separator tree for G' to obtain an $(O(\sqrt{n}), 2/3)$-separator tree for $G(B)$. We now apply Theorem 2.2 and compute the LU factorization $B = LDL^T$. The number of arithmetic operations is $O(n^{\omega/2})$. However, notice that the entries of B are integers whose absolute values are bounded by 3. Thus, the numerators and denominators of the rationals that are obtained during the Gaussian elimination have only $\tilde{O}(n)$ bits, and the bit complexity of each arithmetic operation is $\tilde{O}(n)$. Thus, the overall running time is $\tilde{O}(n^{1+\omega/2})$. By multiplying the rational numbers in the diagonal of D (in $\tilde{O}(n^2)$ time) we obtain $det\ B$. Finally, notice that

$$\#\text{ perfect matchings in } G = (det\ B)^{1/4}.$$

□

For general graphs we may not have a Pfaffian orientation. However, Galluccio and Loebl [GL99a] found a way to overcome this obstacle in the case of graphs with bounded genus. They proved that if G has genus g, then it is possible to find 4^g orientations of G, so that the matching polynomial is a linear combination of the square roots of the determinants of the corresponding skew-adjacency matrices. The coefficients of this linear combination are explicitly given. The orientations can be found quickly as well since each of them corresponds to a Pfaffian orientation of a planar subgraph of G (see [GL99a]). The number of perfect matchings can now be computed by assigning 1 to the variables in each of the 4^g matrices, and applying the algorithm for planar graphs, given above, to compute the determinant of each matrix. Thus, the overall running time is still $\tilde{O}(n^{1+\omega/2})$.

5 Counting T-joins in planar graphs

Let $G = (V, E)$ be a graph. As in Section 3, we associate a variable x_e with each $e \in E$. For $F \subset E$ let $x(F) = \prod_{e \in F} x_e$. For $T \subset V$, the *T-join polynomial* of G is the polynomial $\mathcal{T}(G,T) = \sum_F x(F)$ where F is taken over all the edge sets of the T-joins of G.

Galluccio and Loebl [GL99b] found a construction that relates the T-join polynomial of a graph with the matching polynomial of another graph, while preserving the genus. We describe their reduction.

Definition 5.1. *Let $G = (V, E)$ be a graph and let $v \in V$. Let $e_1, \ldots, e_k$ be an ordering of the edges of G incident with v. The* even splitting *of v is the graph $G' = (V', E')$ such that $V' = V - \{v\} \cup \{v_1, \ldots, v_{6k}\}$, and $E' = E - \{e_1, \ldots, e_k\} \cup \{e'_1, \ldots, e'_k\} \cup \{v_1v_2, v_2v_3, \ldots, v_{6k-1}v_{6k}\} \cup \{v_1v_3, v_4v_6, \ldots, v_{6k-2}v_{6k}\}$ where e'_i is obtained from e_i by replacing v by v_{6i-4} for $i = 1, \ldots, k$. We say that e'_i is the image of e_i in G'. The* odd splitting *of v is obtained from the even splitting of v by deleting vertices v_{6k}, v_{6k-1}, v_{6k-2}.*

Notice that if G is embedded in an orientable surface and $e_1, \ldots, e_k$ is a clockwise ordering of the edges incident with v, then the graph obtained by an even (odd) splitting is also embeddable on the same surface.

Let $G = (V, E)$ be a graph and $T \subset V$. Denote by $G_T = (V_T, E_T)$ the graph obtained from G by odd splitting of all vertices of T and even splitting of all vertices of $V - T$. Notice that, in particular, if G is planar we can guarantee that G_T is also planar and also notice that G_T can be constructed from G in linear time. Now, let $M(G_T)$ be the matching polynomial of G_T and let $\mathcal{T}(G,T)$ be the T-join polynomial of G. The following is observed in [GL99b].

Theorem 5.1. *$\mathcal{T}(G,T)$ is obtained from $M(G_T)$ by performing the following replacement of the variables of $M(G_T)$. If $e' \in E_T$ is an image of some $e \in E$ then replace $x_{e'}$ with x_e. Replace all other variables by 1.*

If $G = (V, E)$ is a planar graph, so is $G_T = (V_T, E_T)$. Hence, let $\vec{G_T}$ be a Pfaffian orientation of G_T, and let $A_s(\vec{G_T})$ be the associated skew adjacency matrix. Replace each variable of $A_s(\vec{G_T})$ as follows. If $e' \in E_T$ is an image of some $e \in E$ then replace $x_{e'}$ with x. Replace all other variables with 1. Each element of the obtained matrix, denoted $A_s(\vec{G_T}, x)$, is one of $\{+1, -1, 0, x, -x\}$. Let $n_G(r, T)$ be the number of T-joins of G consisting of precisely r edges. Notice that $n_G(T) = \sum_{r=0}^{m} n_G(r, T)$ where $m = |E|$. Put

$f(x) = \sum_{r=0}^{m} n_G(r,T)x^r$ and notice that $f(1) = n_G(T)$. By Theorem 4.1 and by Theorem 5.1 we have

COROLLARY 5.1.

$$f(x) = \sum_{r=0}^{m} n_G(r,T)x^r = \sqrt{det\ A_s(\vec{G_T}, x)}.$$

Since $f(1) = n_G(T)$ is the number of perfect matchings of G_T, the algorithm of Section 4 can be used to compute $n_G(T)$ in $\tilde{O}(n^{1+\omega/2})$ time.

Finally, we note that if G has bounded genus, then the algorithm of Theorem 1.1 can also be used to find a T-join of G, if one exists, with high probability, in $\tilde{O}(n^{\omega/2})$ time.

6 Counting T-joins of a given cardinality in planar graphs

In this section we show that $n_G(r,T)$, the number of T-joins of size r, can be computed in $\tilde{O}(n^{2+\omega/2})$. For that, we need to show how to compute the (square root of the) symbolic determinant $det\ A_s(\vec{G_T}, x)$.

Let A be the matrix obtained from $A_s(\vec{G_T}, x)$ by replacing x with (the huge integer) $K = |V_T|^{|V_T|}$. Recall that $|V_T| = O(n)$ where $n = |V|$.

LEMMA 6.1. *$det\ A$, and hence $\sqrt{det\ A}$, can be computed in $\tilde{O}(n^{2+\omega/2})$ time.*

Proof. We use the same proof used in Section 4. The only difference is that the huge number K appearing in some entries of B (possibly multiplied by 3) has $O(n \log n)$ bits. Thus, each arithmetic operation during the Gaussian elimination now costs $\tilde{O}(n^2)$ time.

LEMMA 6.2. *Given $D = \sqrt{det\ A}$, the coefficients of $f(x)$ can be determined in $\tilde{O}(n^2)$ time.*

Proof. By Corollary 5.1, $D = f(K)$. Also notice that $n_G(r,T) \le \binom{m}{r} < K$. Thus, we can determine the coefficients $n_G(r,T)$ by considering the number D as a number in base K. Since $D < (m+1)K^{m+1}$ and since $\log D = \tilde{O}(n^2)$, the lemma follows.

The method presented in Lemma 6.1 can be generalized to find the coefficients of the determinant of a skew adjacency matrix that has more than one free variable. This is useful for solving the following generalized matching problem. Suppose that the edges of an n-vertex planar graph G are colored using k colors, and suppose that $\alpha_1, \ldots, a_k$ are nonnegative integers satisfying $\alpha_1 + \cdots + \alpha_k = n/2$. How many perfect matchings does G have, in which precisely α_i edges of the matching are colored with color i, for $i = 1, \ldots, k$? This problem can be solved using the same approach as in Section 5 in $\tilde{O}(n^{k+\omega/2})$ time. The idea is to assign different variables to edges of different color. We omit the details.

7 Concluding remarks

Our main result is an $O(n^{3\omega/(\omega+3)}) < O(n^{1.326})$ time algorithm for finding maximum matchings in H-minor free graphs, for every fixed graph H.

Acknowledgment

The authors thank Robin Thomas for useful discussions.

References

[AST90] N. Alon, P.D. Seymour, and R. Thomas. A separator theorem for nonplanar graphs. *Journal of the American Mathematical Society*, 3(4):801–808, 1990.

[Blu90] N. Blum. A new approach to maximum matching in general graphs. In *Proc. of 17th ICALP*, pages 586–597, 1990.

[BM04] J.M. Boyer and W.J. Myrvold. On the cutting edge: Simplified $O(n)$ planarity by edge addition. *Journal of Graph Algorithms and Applications*, 8(2):241–273, 2004.

[CNAO85] N. Chiba, T. Nishizeki, S. Abe, and T. Ozawa. A linear algorithm for embedding planar graphs using PQ-trees. *Journal of Computer and System Sciences*, 30(1):54–76, 1985.

[CW90] D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. *Journal of Symbolic Computation*, 9:251–280, 1990.

[Edm65] J. Edmonds. Paths, trees, and flowers. *Canadian Journal of Mathematics*, 17:449–467, 1965.

[Geo73] A. George. Nested dissection of a regular finite element mesh. *SIAM Journal on Numerical Analysis*, 10:345–363, 1973.

[GHT84] J.R. Gilbert, J.P. Hutchinson, and R.E. Tarjan. A separator theorem for graphs of bounded genus. *Journal of Algorithms*, 5(3):391–407, 1984.

[GL99a] A. Galluccio and M. Loebl. On the theory of Pfaffian orientations. I. Perfect matchings and permanents. *Electron. J. Combin.*, 6:R6, 18 pp. (electronic), 1999.

[GL99b] A. Galluccio and M. Loebl. On the theory of Pfaffian orientations. II. T-joins, k-cuts, and duality of enumeration. *Electronic Journal of Combinatorics*, 6:R7, 10 pp. (electronic), 1999.

[GT87] J.R. Gilbert and R.E. Tarjan. The analysis of a nested dissection algorithm. *Numerische Mathematik*, 50(4):377–404, 1987.

[GT91] H.N. Gabow and R.E. Tarjan. Faster scaling algorithms for general graph matching problems. *Journal of the ACM*, 38(4):815–853, 1991.

[HT74] J. Hopcroft and R. Tarjan. Efficient planarity testing. *Journal of the ACM*, 21:549–568, 1974.

[Jer03] M. Jerrum. *Counting, sampling and integrating: Algorithms and complexity.* Birkhäuser, 2003.

[Kas67] P.W. Kasteleyn. Graph theory and crystal physics. In *Graph Theory and Theoretical Physics*, pages 43–110. Academic Press, London, 1967.

[Kur30] C. Kuratowski. Sur le problème des courbes gauches en topologie. *Fundamenta Mathematicae*, 15:271–283, 1930.

[Lit74] C.H.C. Little. An extension of Kasteleyn's method of enumerating the 1-factors of planar graphs. In *Combinatorial mathematics (Proc. Second Australian Conf., Univ. Melbourne, Melbourne, 1973)*, pages 63–72. Lecture Notes in Math., Vol. 403. Springer, Berlin, 1974.

[Lov79] L. Lovász. On determinants, matchings, and random algorithms. In *Fundamentals of computation theory*, volume 2 of *Math. Res.*, pages 565–574. Akademie-Verlag, Berlin, 1979.

[Lov06] L. Lovász. Graph minor theory. *Bull. Amer. Math. Soc. (N.S.)*, 43(1):75–86 (electronic), 2006.

[LP86] L. Lovász and M.D. Plummer. *Matching theory*. North-Holland Publishing Co., Amsterdam, 1986. Annals of Discrete Mathematics, 29.

[LRT79] R.J. Lipton, D.J. Rose, and R.E. Tarjan. Generalized nested dissection. *SIAM Journal on Numerical Analysis*, 16(2):346–358, 1979.

[LT79] R.J. Lipton and R.E. Tarjan. A separator theorem for planar graphs. *SIAM Journal on Applied Mathematics*, 36(2):177–189, 1979.

[McC04] W. McCuaig. Pólya's permanent problem. *Electronic Journal of Combinatorics*, 11(1):Research Paper 79, 83 pp. (electronic), 2004.

[Moh99] B. Mohar. A linear time algorithm for embedding graphs in an arbitrary surface. *SIAM Journal on Discrete Mathematics*, 12(1):6–26, 1999.

[MS04a] M. Mucha and P. Sankowski. Maximum matchings in planar graphs via gaussian elimination. In *Proc. of 12th ESA*, pages 532–543, 2004.

[MS04b] M. Mucha and P. Sankowski. Maximum matchings via Gaussian elimination. In *Proc. of 45th FOCS*, pages 248–255, 2004.

[MV80] S. Micali and V. V. Vazirani. An $O(\sqrt{|V|} \cdot |E|)$ algorithm for finding maximum matching in general graphs. In *Proc. of 21st FOCS*, pages 17–27, 1980.

[RS90] N. Robertson and P.D. Seymour. Graph minors. VIII. A Kuratowski theorem for general surfaces. *Journal of Combinatorial Theory. Series B*, 48(2):255–288, 1990.

[RS04] N. Robertson and P.D. Seymour. Graph minors. XX. Wagner's conjecture. *Journal of Combinatorial Theory. Series B*, 92(2):325–357, 2004.

[RST99] N. Robertson, P.D. Seymour, and R. Thomas. Permanents, Pfaffian orientations, and even directed circuits. *Annals of Mathematics. Second Series*, 150(3):929–975, 1999.

[RW05] B. Reed and D.R. Wood. Fast separation in a graph with an excluded minor. In *Proc. of the 2005 European Conference on Combinatorics, Graph Theory and Applications (EURO-COMB)*, DMTCS proc. AE, pages 45–50, 2005.

[Tho81] C. Thomassen. Kuratowski's theorem. *Journal of Graph Theory*, 5:225–241, 1981.

[Val79] L.G. Valiant. The complexity of computing the permanent. *Theoretical Computer Science*, 8(2):189–201, 1979.

[Vaz89] V.V. Vazirani. NC algorithms for computing the number of perfect matchings in $K_{3,3}$-free graphs and related problems. *Information and Computation*, 80(2):152–164, 1989.

[Vaz94] V.V. Vazirani. A theory of alternating paths and blossoms for proving correctness of the $O(\sqrt{V}E)$ general graph maximum matching algorithm. *Combinatorica*, 14(1):71–109, 1994.

[Wag37] K. Wagner. Über eine Eigenschaft der ebenen Komplexe. *Mathematische Annalen*, 114(1):570–590, 1937.

[Wil97] D.B. Wilson. Determinant algorithms for random planar structures. In *Proc. of 8th SODA*, pages 258–267, 1997.