Planarity, Determinants, Permanents, and (Unique) Matchings*

Samir Datta[1], Raghav Kulkarni**[2], Nutan Limaye[3], and Meena Mahajan[3]

[1] Chennai Mathematical Institute, Siruseri, Chennai 603 103, India.
sdatta@cmi.ac.in
[2] Dept. of Computer Science, Univ. of Chicago, U.S.A. raghav@cs.uchicago.edu
[3] The Institute of Mathematical Sciences, Chennai 600 113, India.
{nutan,meena}@imsc.res.in

Abstract. Viewing the computation of the determinant and the permanent of integer matrices as combinatorial problems on associated graphs, we explore the restrictiveness of planarity on their complexities and show that both problems remain as hard as in the general case, *i.e.* GapL and #P complete. On the other hand, both bipartite planarity and bimodal planarity bring the complexity of permanents down (but no further) to that of determinants. The permanent or the determinant modulo 2 is complete for $\oplus$L, and we show that parity of paths in a layered grid graph (which is bimodal planar) is also complete for this class. We also relate the complexity of grid graph reachability to that of testing existence/uniqueness of a perfect matching in a planar bipartite graph.

1 Introduction

For many natural problems on graphs, imposing planarity does not reduce the complexity. For instance, vertex cover is NP-complete, and remains so even for planar degree-3 graphs; so does planar 3-dimensional matching [19]. The circuit value problem is P-complete, and remains so even if the graph underlying the circuit is restricted to be planar. In [24] and [35], the complexity of several counting problems has been investigated under planar restrictions. More recently, [40] establishes that counting vertex covers remains #P-complete even when restricted to 3-regular planar bipartite graphs. Thus there is some evidence to believe that planarity is not a real restriction at all.

However, there are notable exceptions. In the circuit setting, for instance, monotone circuit value is P-complete, but monotone planar circuit value is in NC [41, 18] (see also [27]). Constant-width circuits characterize NC^1 [8], while planar constant-width circuits characterize its subclass ACC^0 [20]. In the purely graph-theoretic setting, counting the number of perfect matchings in a graph is #P-hard [36] (and remains hard even if the graph is bipartite), while counting

* A preliminary version of this paper appeared in [16].

** Part of this work was done while this author was visiting the Chennai Mathematical Institute.

the same number in a planar graph is in NC [38, 28]. This fact is intimately related to the algebraic nature of the problem involved: in the first case, computing the permanent of a 0-1 matrix is hard, while in the second case, finding a Pfaffian orientation and computing the Pfaffian is easy (equivalent to computing the determinant). Another very recent exception concerns reachability. Given a directed graph G and two vertices s and t, determining whether there is a path from s to t is the canonical complete problem for nondeterministic logspace NL. However, if the graph is planar, then a recent result from [10, 11], building on the techniques of [33, 4], shows that the presence, and even the absence, of a path can be detected in unambiguous logspace UL. While UL is known to coincide with NL in the non-uniform setting, and even in the uniform setting under a plausible hardness condition [6], as of now they are not known to coincide unconditionally. So the result of [11] can be seen as an instance of planarity reducing the complexity of a problem.

Thus we see that the condition of planarity can be exploited in establishing better upper bounds in some cases. Since for many natural problems, the instances that arise in practice are indeed planar, it is worthwhile trying to better understand how planarity can help. With this motivation, we examine the complexity of determinant, permanent, and unique perfect matchings when restricted to planar instances. Recall that both the determinant and the permanent of the adjacency matrix of a graph G count the total weight of all cycle covers in G, with the one difference that the determinant considers the *signed* weight. Computing the determinant (over integers or rationals) is known to be GapL-complete [15, 34, 37, 39], while computing the permanent is known to be #P-complete, [36]. However, testing whether the 0-1 permanent is zero is in P (this is because the 0-1 permanent equals the number of perfect matchings in a related bipartite graph) and thus significantly easier than #P or NP, whereas testing whether the determinant (of an integer matrix, not necessarily 0-1) is zero is complete for the exact-counting-in-logspace class $\mathsf{C_{=}L}$ [3], and thus at least as hard for NL. Interestingly, the permanent mod 2 equals the determinant mod 2 and is thus easy to compute; in fact it is complete for the parity logspace class $\oplus$L. Another complete problem for $\oplus$L is checking whether the number of $s \rightsquigarrow t$ paths in a directed acyclic graph (DAG) is odd. Testing whether a bipartite graph has a perfect matching, B-PM, is known to be hard for NL [13], while testing whether a bipartite graph has a unique perfect matching, B-UPM, is known to be hard for NL and in $\mathsf{C_{=}L} \cap \mathsf{NL}^{\oplus \mathsf{L}}$ [23]. We examine planar restrictions of these and related problems.

Our main results are summarized in Table 1. (The terms involved are explained in the respective sections.) In some cases, we have said that a problem is A-hard where A is another problem rather than a complexity class. We use the name of the problem A to also denote the class of problems reducible to A via suitably restrictive reductions (typically first-order projections, or many-one reductions computable in AC^0, but sometimes also logspace many-one reductions).

The rest of this paper is organised as follows. Section 2 briefly describes the notation needed to describe the results of the paper. Section 3 describes the

Problem	General bound	Restriction	Our New Bound
Total signed weight of cycle covers (Determinant of adjacency matrix)	GapL-complete	planar	GapL-hard
Total weight of cycle covers (Permanent of adjacency matrix)	#P-complete	planar	#P-hard
		planar bipartite	GapL-complete
		planar bimodal	GapL-complete
Total weight of perfect matchings (Permanent of bip-adjacency matrix)	#P-complete	planar	GapL-complete
Parity of $\#s \rightsquigarrow t$ paths in DAG	$\oplus$L-complete	planar, even layered grid graph	$\oplus$L-hard
Bipartite UPM	NL-hard	planar	L-hard, co-LGGR-hard
	in $\mathsf{C_{=}L} \cap \mathsf{NL}^{\oplus\mathsf{L}}$	planar	in $\oplus$L
			equiv to GGUPM
Bipartite PM	NL-hard	planar	L-hard, GGR-hard
			equiv to GGPM

Table 1. Main results

outline of a technique that is repeatedly used later, and the details from one step. Section 4 describes the hardness results concerning determinant and permanent, and 5 describes the membership algorithms. Section 6 describes the hardness of $\oplus$LGGR for $\oplus$L, and Section 7 describes the results concerning Planar-B-UPM.

2 Notation and Preliminaries

L and P denote deterministic logspace and polynomial time computation, respectively. We consider the nondeterministic classes NP and NL, their counting counterparts #P and #L, and the closures of these under subtraction GapP and GapL. The reader is referred to any standard complexity theory book (*e.g.* [7, 31]) for details. We also consider (1) the exact counting in logspace class $\mathsf{C_{=}L}$; a language L is in $\mathsf{C_{=}L}$ if and only if some GapL function vanishes exactly on strings in L, and (2) the parity logspace class $\oplus$L; L is in $\oplus$L if and only if some GapL function takes odd values exactly on strings in L. It is known that $\mathsf{NL} \subseteq \mathsf{C_{=}L}$ and that $\oplus\mathsf{L}^{\oplus\mathsf{L}} = \oplus\mathsf{L}$. The canonical complete problem for NL is Reachability in a directed acyclic graph. A complete problem for GapL is computing the determinant of an integer matrix; hence testing singularity of a matrix is complete for $\mathsf{C_{=}L}$. See for instance [1].

We consider planar graphs specified by the planar combinatorial embeddings: such an embedding specifies, for each vertex, the cyclic ordering of edges incident on it in some plane drawing. Testing planarity and obtaining planar combinatorial embeddings can be done in L by the results of [5, 32]. A planar embedding of a directed graph is said to be bimodal if at every vertex, all the incoming edges appear contiguously in the cyclic ordering. Not every planar graph has a bimodal embedding. The reader is referred to any graph drawing book for more details; see for instance [30].

A grid graph is a directed graph with vertices laid out on the plane at integer coordinates, and edges going unit distance east-west or north-south only. A grid graph is layered if all horizontal edges are in the same direction (say left-to-right, or x-monotone), and so are all vertical edges (y-monotone). GGR and LGGR denote the reachability problem restricted to instances (G, s, t) where G is a grid graph or a layered grid graph respectively.

For any directed graph H with a special source vertex s and sink vertex t, define the split graph $\text{Split}(H)$ as follows: (1) split every node v into two nodes, v_{in} and v_{out}, (2) for every edge (u, v) in the original graph, draw an edge from u_{out} to v_{in}, with the same weight, (3) draw the edges from v_{in} to v_{out} for each v, with weight 1, and (4) delete s_{in} and t_{out}; rename s_{out} and t_{in} as s and t. Note that $\text{Split}(H)$ is always bipartite, and we can always obtain a bipartition with s and t in different parts. Further, if H has a bimodal planar embedding, then $\text{Split}(H)$ is also bimodal planar, and the witnessing embedding can be easily obtained from that of H. (If H is planar but not bimodal, then $\text{Split}(H)$ may not be planar at all.)

Corresponding to any $n \times n$ matrix M, we can associate two graphs: G_M is a directed graph on n vertices, with edge $\langle i, j \rangle$ having weight $M(i, j)$, and H_M is an undirected bipartite graph on $2n$ vertices, with edge $(i, n + j)$ having weight $M(i, j)$. M is said to be the adjacency matrix of G_M and the bipartite adjacency matrix of H_M. A cycle cover in a graph is a collection of vertex disjoint cycles spanning the graph. The determinant of a matrix M, $\mathsf{Det}(M)$, equals the total signed weight of all cycle covers in G_M, while its permanent, $\mathsf{Perm}(M)$, equals the total unsigned weight of all cycle covers in G_M. The sign of a cycle cover is $(-1)^k$, where k is the number of even length cycles in the cover. $\mathsf{Perm}(M)$ also equals the total weight of all perfect matchings in H_M. Here the weight of a cycle cover or matching is the product of the weights of its constituent edges.

3 Drawing a graph on the plane

A unifying technique behind all the hardness results except those concerning UPM is that of "Planarizing" a graph by first drawing the graph on a plane (potentially with intersection among the edges) and then replacing each crossing by a planar gadget so as to preserve some property (*e.g.* the number of cycle covers or the parity of the number of s, t-paths).

Thus the generic template for the hardness reductions described in Sections 4 and 6 is as follows: Given the input instance $G = (V, E)$,

1. (Optional) Preprocess the graph to satisfy some constraint (*e.g.* bounded degree constraint) along with bipartiteness.
2. Obtain a drawing of G on a plane, such that the edges are straight line segments and no two crossings share the same coordinates.
3. Uniformly replace every crossing in the drawing by a planarizing gadget H to get a new planar graph G'.

4. (Optional) Post-process the planar graph to convert it into a graph which has some additional properties.

All the above reductions will be computable in L.

We now describe a way to perform Step 2 above.

Proposition 1. *A bipartite graph can be drawn on the plane with straight-line edges, and with no two crossings sharing the same coordinates. The combinatorial embedding corresponding to such a drawing (including the order of crossings along each edge) can be obtained in logspace.*

Proof. The maximal bipartite graph with n vertices in each part is $K_{n,n}$, so it suffices to show how to draw it. To draw $K_{n,n}$, place vertices of the first part on the x-axis, vertex u_i at $(0, i)$. Place vertices of the second part on the $x = 1$ line suitably spaced apart; place vertex v_j at $(1, n^{2j})$.

To see why this ensures that at most two edges intersect at a point, recall from elementary coordinate geometry that three lines $y = m_i x + c_i$ (for $i = 1, 2, 3$) intersect in a common point if and only if the determinant $\begin{vmatrix} 1 & c_1 & m_1 \\ 1 & c_2 & m_2 \\ 1 & c_3 & m_3 \end{vmatrix} = 0$. The line corresponding to the edge (u_i, v_j) has the equation $y = (n^{2j} - i)x + i$. Thus we want to ensure that $\begin{vmatrix} 1 & i_1 & t^{j_1} - i_1 \\ 1 & i_2 & t^{j_2} - i_2 \\ 1 & i_3 & t^{j_3} - i_3 \end{vmatrix} = \begin{vmatrix} 1 & i_1 & t^{j_1} \\ 1 & i_2 & t^{j_2} \\ 1 & i_3 & t^{j_3} \end{vmatrix}$ is non-zero for distinct i_1, i_2, i_3 and distinct j_1, j_2, j_3, where $t = n^2$. But this determinant equals $(i_3 - i_2)t^{j_1} + (i_1 - i_3)t^{j_2} + (i_2 - i_1)t^{j_3}$, which is clearly non-zero for $t = n^2$ (since none of the terms are 0 by the distinctness of the i's and j's, while the term with the largest value of j is at least n times larger in absolute value than the other two terms and hence cannot be cancelled).

Determining whether two edges (u_i, v_j) and (u_k, v_l) intersect is trivial: they intersect if and only if $i < k$ and $j > l$, or if $i > k$ and $j < l$. Determining the order of crossings along an edge – for edge (u_i, v_j), is the crossing with (u_k, v_l) closer to u_i than the crossing with (u_r, v_s)? – is also easy, depending only on some simple arithmetic operations and comparisons involving n, i, j, k, l, r, s.

The above procedure can be performed in L since both iterated product and division over the naturals can be performed in these classes [22]. $\square$

Remark 1. This can be extended in the obvious way to a layered graph. Let G have vertices in m layers, n per layer, with all edges between a vertex at layer k and a vertex at layer $k + 1$ for some k. Then embed u_{ik} (the ith vertex at layer k) at (k, i) if k is even, and at (k, n^{2i}) if k is odd. The above construction goes through, still in logspace.

4 Planarizing the Determinant and the Permanent: retaining hardness

Computing the determinant (over integers) is known to be GapL-complete [15, 34, 37, 39]. We show that it remains hard if the matrix is restricted to be the adjacency matrix of a planar graph. Weights in {0,1} suffice, and if the graph is required to be bipartite then weights in {-1,0,1} suffice. Further, a natural complete problem for GapL is finding the total weight of all $s \leadsto t$ paths in a weighted directed acyclic graph DAG. We show that this problem remains GapL-hard even if the DAG is restricted to be planar. However, to achieve planarity we crucially require negative weights.

We also investigate the complexity of the planar permanent. The permanent itself is #P-complete [36], though the hardness is under Turing reductions. (The number of queries required can be brought down to one, [42], but there cannot be a many-one reduction unless P = NP, since existence of a matching can be tested in polynomial time.) There are two types of planar restrictions we can consider, and they have quite a different flavour. We want to compute $\mathsf{Perm}(M)$ when either the graph G_M or the graph H_M (see Section 2) is planar. If we require H_M to be planar, then #P-hardness is lost, because the total weight of perfect matchings in a planar (bipartite or otherwise) graph can be done in GapL using the framework of Pfaffians; see [38, 28]. We show that this is in fact not just in GapL but also GapL-complete. Though [28] shows that computing the Pfaffian is GapL-complete, the underlying graphs are not planar. We show hardness without recourse to Pfaffians.

If we require that the graph G_M is planar, then we are counting the total weight of cycle covers in a planar graph. We show that this restriction continues to be as hard as the original problem, *i.e.* #P-hard. On the other hand, if G_M is restricted to be bimodal planar, or simultaneously planar and bipartite, then we show that computing $\mathsf{Perm}(M)$ is GapL-hard. This is the best lower bound possible, since in the next section we also show that in these cases we can also evaluate the permanent in GapL.

The results of this section can be summarized as follows:

Theorem 1. *The following problems are hard for* GapL *via* $\leq_m^{\log}$ *reductions.*

1. Det(M) *for planar* G_M *(total signed weight of cycle covers in planar graph), even when* M *has only 0-1 entries.*
2. Perm(M) *for planar bipartite bimodal* G_M *(total weight of cycle covers in planar bipartite graph with a bimodal embedding), when* M *has entries from* $\{-1, 0, 1\}$.
3. Perm(M) *for planar bipartite* H_M *(total weight of perfect matchings in planar bipartite graph), when* M *has entries from* $\{-1, 0, 1\}$.

Further, computing Perm(M) *for planar* G_M *(total weight of cycle covers in planar graph) is hard for* #P, *even if* M *is restricted to have 0-1 entries.*

After a basic starting step described below (Section 4.1), each part of this Theorem is proved in a separate sub-section.

4.1 GapL $\leq_m^{\log}$ Total Weight of $s \leadsto t$ Paths in $\{-1, 0, 1\}$ Planar DAGs

We start with the canonical GapL-complete problem Directed Path Difference (see for instance [34, 29]). The input is a directed graph G with special vertices

s, t_+ and t_-, and the desired output is the difference in the number of $s \rightsquigarrow t_+$ paths and the number of $s \rightsquigarrow t_-$ paths. That is, computing the difference below is GapL-complete.

$$\#(G, s, t_+, t_-) = \#(s \rightsquigarrow t_+) - \#(s \rightsquigarrow t_-)$$

Without loss of generality, we can assume that

1. G is acyclic and layered (vertices appear in layers and all edges go from a layer to the next layer). (In particular, this implies that the undirected graph underlying G is bipartite.)
2. s is on the first layer and t_+ and t_- on the last layer. and all $s \rightsquigarrow t_+$ or $s \rightsquigarrow t_-$ paths are of even length.
3. All edges have weight 1.
4. The number of vertices is odd.

(G is essentially the configuration graph of the underlying NL machine, with time-stamped configurations at the vertices. t_+ and t_- correspond to the unique Accept and Reject configurations respectively, while s is the Start configuration.)

We create a new vertex t and add edge $\langle t_+, t\rangle$ with weight 1, and edge $\langle t_-, t\rangle$ with weight -1, to get G_1. All $s \rightsquigarrow t$ paths are of odd length, and G_1 remains bipartite. The hard function is the total weight of all $s \rightsquigarrow t$ paths in G_1. Now we planarize G_1 as follows:

The graph G_1 is drawn in the plane (with edge crossings) as described in Proposition 1 and the remark after it. In logspace, we can determine which edges cross in this layout, and the linear order of the crossings involving any one edge. Now we replace each crossing by the gadget shown in Figure 1 to get a planar graph G_2. Observe that for any vertices a, b in G_1, the weight of each $a \rightsquigarrow b$ path as well as the parity of the length of the path is preserved in G_2. Some new paths are introduced, for instance between A and D in the figure, but their net contribution is zero. Since G (and G_1) was bipartite, so is G_2. (Here bipartiteness is in the undirected sense: there are no undirected odd cycles.) Also, the embedding of G_2 we have is *upward planar*; it is planar and all edges are monotonic with respect to the x-coordinate. In particular, this implies that the embedding of G_2 is bimodal. Without loss of generality, assume that G_2 has an odd number N of vertices.

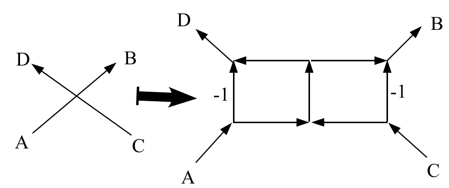

Fig. 1. Planarizing gadget for weighted paths (determinant)

(Note: Using the techniques of Section 6, we can even ensure that G_2 is a *layered grid graph*.)

Now we adapt Toda's proof [34] that Directed Path Difference reduces to Determinant, starting with G_2. There are two ways to proceed, described in the next two subsections.

4.2 GapL $\leq_m^{\log}$ Planar 0-1 Determinant

Subdivide every edge of weight 1 into two edges of weight 1. Change all weights -1 to 1. Add an edge from t to s and add self loops at every node except s and t; these edges are all of weight 1. Call this graph G_3; clearly, it is also planar. As argued in [34], every $s \rightsquigarrow t$ path ρ in G_2 corresponds uniquely to a cycle cover in G_3, consisting of one big cycle (ρ, with subdivisions wherever applied, and the $\langle t, s\rangle$ edge) and several self-loops. The weight of this cycle cover is 1. The sign of this cycle cover depends on the number of even-length cycles in it. But except for the big cycle, all other cycles are of length 1. So this cycle cover is positive if and only if the big cycle has odd length. If ρ has p edges of weight $+1$ and q edges of weight -1, then the big cycle has length $2p + q + 1$. So the sign of the cycle cover in G_3 is positive if and only if the path weight in G_2 is positive.

Thus, if A_3 is the adjacency matrix of G_3, then

$$\mathsf{Det}(A_3) = \#(G_2, s, t) = \#(G_1, s, t) = \#(G, s, t_+, t_-)$$

It is easy to see that G_2, G_3 can be obtained from (G, s, t) in logspace.

4.3 GapL$\leq_m^{\log}$ {-1,0,1} Bipartite Planar Bimodal Determinant / Permanent

The above method loses bipartiteness not just because it adds self-loops (we can ignore these), but also because of asymmetric subdivisions for weight 1 or -1. To avoid this, we consider a slightly different construction.

Starting with the graph G_2, we first construct its split graph. To this we add edges $\langle v_{out}, v_{in}\rangle$ for each $v \notin \{s, t\}$, and the edge $\langle t, s\rangle$; all these edges have weight 1. Call this graph G_4. Note that G_4 is planar, bipartite, and bimodal.

As argued above, every $s \rightsquigarrow t$ path ρ in G_2 corresponds uniquely to a cycle cover in G_4, consisting of one big cycle (ρ and the $\langle t, s\rangle$ edge) and several 2-cycles of the form (v_{in}, v_{out}), and there are no other cycle covers. The weight of this cycle cover is the weight of ρ. Its sign depends on the number of even cycles in it. First consider the big cycle. By assumption, ρ is of odd length, say $2l + 1$ edges in G_2. Then it has $2l$ internal vertices, each of which is split in G_4. So the big cycle is of even length $4l + 2$. Now, the remaining vertices of G_2 (and there are $N - 2l - 2$ of these), are covered in the cycle cover by their split 2-cycles. So the total number of even cycles is $N - 2l - 2 + 1$, which is even since we ensured that N was odd. Thus all cycle covers in G_4 have positive sign.

Thus, if A_4 is the adjacency matrix of G_4, then

$$\mathsf{Det}(A_4) = \mathsf{Perm}(A_4) = \#(G_2, s, t) = \#(G_1, s, t) = \#(G, s, t_+, t_-)$$

It is easy to see that G_4 can be obtained from (G, s, t) in logspace.

4.4 $\mathsf{GapL} \leq_m^{\log}$ Total Weight of perfect matchings in $\{-1, 0, 1\}$ bipartite planar graph

Now consider the situation when we want to compute $\mathsf{Perm}(M)$ and the bipartite graph H_M is planar. $\mathsf{Perm}(M)$ is exactly the total weight of all perfect matchings in H_M. We show that this is GapL-hard, by describing an undirected graph that is planar and bipartite, such that the total weight of all perfect matchings in it is a GapL-hard function. We showed in Section 4.1 that every GapL- function can be expressed as $\#(G_2, s, t)$ for a directed layered planar bipartite bimodal graph G_2. We construct the split graph $\mathrm{Split}(G_2)$ (as defined in Section 2) and we let G_5 be the underlying undirected graph of $\mathrm{Split}(G_2)$. Then G_5 is planar and bipartite. Furthermore, $s \rightsquigarrow t$ paths in G_2 are in 1-1 correspondence with perfect matchings in G_5 of the same weight. Thus the sum of the weights of the perfect matchings in G_5 is precisely $\#(G_2, s, t)$. (See [13, 23] for details.)

4.5 0-1 Permanent $\leq_m^{\log}$ 0-1 Planar Permanent

We now show that computing $\mathsf{Perm}(M)$, when G_M is planar, is as hard as computing arbitrary permanents (*i.e.* #P-hard). Recall that $\mathsf{Perm}(M)$ computes the total weight of all cycle covers in G_M. Let N be the $n \times n$ matrix whose permanent we wish to compute. Consider the matrix $A = \begin{pmatrix} 0_n & N \\ I_n & 0_n \end{pmatrix}$ where I_n and 0_n denote the identity and the all-zeros matrices of size n. Clearly $\mathsf{Perm}(A) = \mathsf{Perm}(N)$. Consider any drawing of the directed bipartite graph G_A as discussed in Section 3.

As in Section 4.1, we replace each crossing with a planarity gadget so as to preserve the total weights of cycle covers. The planarity gadget used is shown in Figure 2. Cycle covers using exactly one of the two edges AB or CD will now use the corresponding length 3 path $AXYB$ or $CYXD$. Cycle covers using neither of these edges will now use the 2-cycle XY. Cycle covers using both edges are essentially spliced; locally, we use instead of edges AB and CD the paths AXD and CYB. Thus if AB and CD were on the same cycle earlier, they are now on different cycles. If they were on different cycles to begin with, they remain on different cycles due to planarity. (The cycles will cross an even number of times.)

Applying this planarity gadget to all crossings, we obtain a planar graph G_6 with adjacency matrix M. Since $\mathsf{Perm}(M) = \mathsf{Perm}(A) = \mathsf{Perm}(N)$, we have established the hardness of planar permanent.

Note that the planarity gadget in Figure 2 preserves neither bipartiteness nor bimodality. This is not surprising, given the results of the next section.

5 Easy versions of Planar Permanent restrictions

We now show that certain planar restrictions of the permanent problem are significantly easier than #P, in fact, they are computable in GapL. We establish the following theorem.

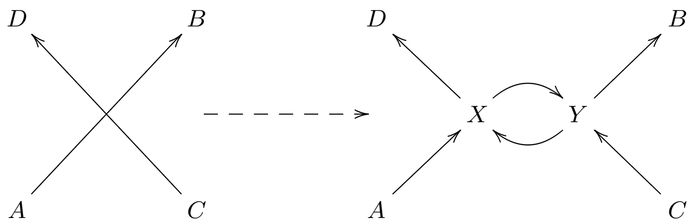

Fig. 2. Planarizing gadget for weighted cycle covers (permanent)

Theorem 2. *1.* Perm*(M) for planar bipartite G_M (total weight of cycle covers in a planar bipartite graph) is computable in* GapL*.*

2. Perm*(M) for planar bimodal G_M (total weight of cycle covers in a planar bimodal graph) is computable in* GapL*.*

3. Even-Odd Crossings Difference – the difference between the number of cycle covers with even number of crossings and the number of cycle covers with odd number of crossings, in a given plane drawing of a (possibly non-planar) graph G – for bipartite graphs, is computable in $\mathsf{L}^{\mathsf{GapL}}$.

Proposition 2. *Computing* Perm*(M) for planar bipartite H_M is* GapL*-complete.*

Proof. In Section 4.4 we showed that Perm(M) for planar bipartite H_M is GapL-hard. Since finding the total weight of perfect matchings in planar graphs can be computed in GapL ([38, 28]), this too is a completeness result.

5.1 Perm(M) for Bipartite Planar G_M is in GapL

Let $G_M = (V, E)$ be the given bipartite (directed) graph, with bipartition $X \dot{\cup} Y$. Let E_1 be those edges of E directed from X to Y, and E_2 be the remaining edges, and let $G_i = (V, E_i)$ for $i = 1, 2$ be planar bipartite undirected graphs. Since bipartite-testing is in L as a consequence of [32], we can compute in logspace an appropriate renumbering of vertices so that the adjacency matrix has the form $M = \begin{pmatrix} 0_n & A_1 \\ A_2 & 0_n \end{pmatrix}$ where $H_{A_1} = G_1$ and $H_{A_2} = G_2$. (If G_M were undirected, we would have $A_1 = A_2^T$.) Clearly, Perm(M) = Perm$(A_1) \times$ Perm(A_2). But Perm(A_i) equals the total weight of perfect matchings in G_i, and since G_i is planar, this can be computed in GapL (see [38, 28]). Hence Perm(M) can be computed in GapL.

5.2 Perm(M) for Planar Bimodal G_M is in GapL

To see how to compute Perm(M) in GapL when G_M is bimodal, observe that Split(G_M) is then planar bipartite bimodal, and the cycle covers in the two graphs are in 1-1 correspondence. By Section 5.1 above, we know that the total weight of cycle covers in a planar bipartite graph can be computed in GapL; hence the same can be done for a planar bimodal graph.

5.3 Even-odd Crossings Difference for Bipartite graphs is in $\mathsf{L}^{\mathsf{GapL}}$

If we can replace the crossings in a graph drawing by a gadget which preserves the weighted sum of cycle covers and also preserves bipartiteness or bimodality, then arbitrary permanents would be expressible as planar bipartite permanents, implying the unlikely collapse of #P to GapL. This suggests that such gadgets are unlikely to exist.

However, we do have a bipartiteness preserving gadget which reduces the Even-Odd Crossings Difference problem to cycle covers in planar graphs. The Even-Odd Crossings Difference problem is as follows: Given a specific drawing $D(G)$ of the graph G, compute the difference between the number of cycle covers with even number of crossings Even-CC($D(G)$) and the number of cycle covers with odd number of crossings Odd-CC($D(G)$). To achieve the reduction, we want to replace each crossing in $D(G)$ by a planar gadget such that in the resulting planar graph H, the total weight of cycle covers corresponds to this difference.

The gadget shown in Figure 3 will do the job. To see this, consider the sum of the weights of the cycle covers in the new graph. Every cycle cover $\mathcal{C}$ in G can be extended to a cycle cover in H in several ways. Consider the local situation at the crossing depicted in the figure. The table below shows the number of ways in which $\mathcal{C}$ can be extended at this crossing.

crossing edges used in $\mathcal{C}$	extensions to H
neither AB nor CD	4 positive extensions, 2 negative extensions
AB but not CD	2 positive extensions
CD but not AB	2 positive extensions
both AB and CD	2 negative extensions

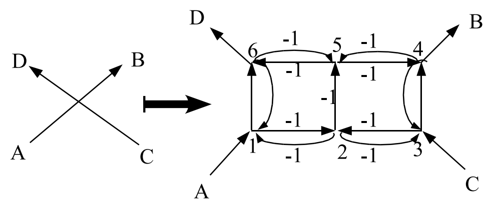

Fig. 3. Planarizing gadget for even-odd-difference

Consider the first case: $\mathcal{C}$ uses neither AB nor CD. The extensions must cover vertices 1,2,3,4,5,6 without using any of the edges $A1$, $C3$, $4B$, $6D$. The possibilities are: (123456) with weight +1, (165432) with weight +1, (1256)(34) with weight −1, (16)(2543) with weight −1, (16)(23)(45) with weight +1, (12)(34)(56) with weight +1.

In the second case, $\mathcal{C}$ uses AB but not CD. The extensions must use both edges $A1$ and $4B$ but neither of $C3$ and $6D$. The possibilities are: $(A1654B)(23)$

and $(A1234B)(56)$, both with weight $+1$. (The path $A1254B$ cannot be used in an extension because it leaves no way of covering vertices 3 and 6.) Similarly, in the third case where $\mathcal{C}$ uses CD but not AB, the only possibilities are $(C3216D)(45)$ and $(C3456D)(12)$, both with weight $+1$.

In the fourth case, $\mathcal{C}$ uses both AB and CD. So the extensions must use all of the edges $A1$, $C3$, $4B$, $6D$. The possibilities are $(A16D)(C3254B)$ and $(A1256D)(C34B)$, both with weight -1.

Now, let k be the total number of gadgets used in planarizing $D(G)$. Then from the table we see that every cycle cover $\mathcal{C}$ in the original graph contributes, upto the sign, exactly 2^k. Furthermore, if $\mathcal{C}$ uses c crossings in $D(G)$, then $\mathcal{C}$ contributes $+2^k$ if and only if c is even. That is, the contribution of $\mathcal{C}$ is $(-1)^c 2^k$. Thus,

$$\mathsf{Perm}(M) = 2^k \times [\#\text{Even-CC}(D(G)) - \#\text{Odd-CC}(D(G))]$$

where M is the adjacency matrix of H (*i.e.* $H = G_M$).

Now, if we start with a bipartite graph G, then the resulting graph H will be bipartite planar, and by the result of Section 5.1, $\mathsf{Perm}(M)$ can be computed in GapL. So, for plane drawings of bipartite graphs, Even-Odd Crossings Difference can be computed in $\mathsf{L}^{\mathsf{GapL}}$.

Remark 2. We can change the weights of all the edges out of vertex 2 (or vertex 5) in Figure 3 to weight $-1/2$ instead of -1. Then, since any cycle cover uses exactly one of these edges, the contribution per crossing becomes ± 1 instead of ± 2. This eliminates the 2^k factor in the expression above. On the other hand, it still doesn't change the upper bound from $\mathsf{L}^{\mathsf{GapL}}$ to GapL, because now the required resulting permanent is over rationals rather than integers.

6 Hardness of $\oplus\mathsf{LGGR}$ for $\oplus\mathsf{L}$

Although the permanent is $\#\mathsf{P}$-hard, the permanent mod 2 equals the determinant mod 2 and is thus complete for $\oplus\mathsf{L}$. A canonical $\oplus\mathsf{L}$-complete problem is counting the number of $s \rightsquigarrow t$ paths, mod 2, in a directed acyclic graph (DAG). We show that this remains $\oplus\mathsf{L}$ hard (under $\leq_m^{\log}$-reductions) even if the DAG is planar, further, even if it is a layered grid graph. This is in contrast to the situation for the decision version (reachability in a DAG), where the general case is NL-complete while its restriction to planar graphs is in $\mathsf{UL} \cap \text{co-}\mathsf{UL}$ [11]. (Planar Directed Reachability PDR is known to be L-hard under AC^0 many-one reductions, and is also logspace many-one equivalent to reachability in grid graphs GGR, but its exact complexity is still unknown. Reachability in layered grid graphs LGGR is not even known to be L-hard. The complexity of various versions of grid graph reachability is investigated in [2].)

Formally, we show:

Theorem 3. $\oplus\mathsf{L} \leq_m^{log} \oplus\mathsf{LGGR}$

The following chain of reductions establishes the result.

$\oplus$Paths-in-DAGs $\leq_m^{\log}$ $\oplus$Paths-in-Planar-DAGs: Without loss of generality, we can assume (as in Section 4.1) that G is layered, with s at the first layer and t at the last. We draw G on the plane as described in Section 3, with edge crossings. We then replace every crossing C in the drawing of G by the planarizing gadget H from Figure 1 to obtain a planar graph G_1. (Here, no negative weight edges are used. All edges have weight 1.) Observe that the parity of the number of paths between vertices a, b, c, d in the crossing C is preserved in H. (In C, there is exactly one $a \rightsquigarrow b$ and $c \rightsquigarrow d$ path, and no $a \rightsquigarrow d$ or $c \rightsquigarrow b$ path. In H, there is exactly one $a \rightsquigarrow b$ and $c \rightsquigarrow d$ path, and two $a \rightsquigarrow d$ or $c \rightsquigarrow b$ paths each.) Hence, $\#(G, s, t) \equiv \#(G_1, s, t) \pmod 2$.

$\oplus$-Paths-in-Planar-DAGs $\leq_m^{\log}$ $\oplus$Paths-in-x-Monotone-Grid-Graphs: We now embed G_1 into a grid in a layered fashion. In [4], a logspace procedure for embedding any planar graph into a grid graph (preserving reachability and even number of paths) was first described. This procedure when applied to a directed acyclic layered graph gives a grid graph in logspace. But the grid graph thus obtained is neither x-monotone nor y-monotone. In [12], a logspace procedure is described to embed any layered planar DAG into a grid in an x-monotone way; apply this to G_1 to obtain G_2. It is easy to see that this procedure preserves not only reachability, but also the exact count of the number of paths. That is, $\#(G_1, s, t) \equiv \#(G_2, s', t') \pmod 2$.

$\oplus$Paths-in-x-Monotone-Grid-Graphs $\leq_m^{\log}$ $\oplus$Paths-in-layered-grid-graphs $=$ $\oplus$LGGR: As mentioned in [2], an xy-monotone grid graph (*i.e.* a layered grid graph) can be obtained from any x-monotone grid graph via first-order projections, preserving reachability. This reduction was first sketched in [9]; a detailed description of this can also be found in [12]. Applying this to G_2 yields a layered grid graph G_3, and again, it is easy to see that the exact count of the number of paths is preserved. That is, $\#(G_2, s', t') \equiv \#(G_3, s'', t'') \pmod 2$.

7 (Unique) Perfect Matchings in Planar Bipartite Graphs

We now investigate the complexity of checking existence and unique existence of a perfect matching in a bipartite graph, B-PM and B-UPM respectively, when restricted to planar instances. Both B-PM and B-UPM are known to be NL-hard ([13, 23]), but B-UPM is believed to be easier since unlike B-PM, it is known to be in NC (in both $\mathsf{C_{=}L}$ and $\mathsf{NL}^{\oplus \mathsf{L}}$ [23]). We provide two further pieces of evidence that B-UPM may be easier by considering the planar restrictions of these problems, Planar-B-PM and Planar-B-UPM.

Firstly, we show that while both Planar-B-PM and Planar-B-UPM are L-hard, Planar-B-PM is hard for Planar Directed Reachability PDR, whereas Planar-B-UPM is hard only for co-Layered Grid Graph Reachability co-LGGR. (It is known that PDR is equivalent to co-PDR and to its restriction Grid Graph Reachability

GGR, by [4]). This former hardness can be viewed as a planarization of the result "Reachability reduces to B-PM". We do not know how to planarize the result "co-Reachability reduces to bipartite-UPM" from [23].

Secondly, we obtain an upper bound of $\oplus$L for Planar-B-UPM. This can be viewed as a planarization of the result "B-UPM is in Reach$^{\oplus\mathsf{L}}$" from [23]: our algorithm is a GGR$^{\oplus\mathsf{L}}$ algorithm, and since Section 6 shows that $\oplus$LGGR is hard for $\oplus$L, it is in fact in GGR$^{\oplus\mathsf{LGGR}}$.

We note, however, that the complexity of LGGR (and co-LGGR) is an interesting question in its own right. It is not known whether it is in L, or L- hard, or reducible to its complement co-LGGR. However, its best known upper bound is the same as that for PDR, namely UL $\cap$ co-UL.

Also, analogous to the equivalence of PDR and GGR, we show that Planar-B-PM and Planar-B-UPM are equivalent to testing existence or unique existence of perfect matchings in grid graphs, GGPM and GGUPM respectively.

We also consider the related problem of testing uniqueness of a minimum-weight perfect matching. In a bipartite graph with unary weights, this is known to be hard for NL and in L$^{\mathsf{C_=L}}$ and NL$^{\oplus\mathsf{L}}$ [23]. No better upper bound is known for the planar restriction, though the lower bound is also not known to hold. We show that GGR reduces to this planar restriction.

The results in this section can be summarized as follows.

Theorem 4. *1.* (L $\cup$ *co*-LGGR) $\leq_{proj}$ Planar-B-UPM $\equiv_{proj}$ GGUPM $\in \oplus$L
2. (L $\cup$ GGR) $\leq_{proj}$ Planar-B-PM $\equiv_{proj}$ GGPM
3. Testing uniqueness of a min-weight perfect matching in a planar bipartite graph with unary weights is hard for GGR*.*

See Figure 4 for a schematic view. The highlighted arrows in pairs denote "planarizing" results: (1) the two dotted arrows represent the known result "NL (Reachability) reduces to B-PM" and its planarized version "Planar Reachability reduces to Planar-B-PM"; and (2) the two dashed arrows represent the known result "testing unique existence of perfect matchings in bipartite graphs is in NL with $\oplus$L oracle, that is, reducible to Reach with $\oplus$L oracle" and its planarized version showing that if the graphs are planar then this test is reducible to planar Reach with the $\oplus$L oracle.

7.1 L$\leq_{proj}$ Planar-B-UPM; L$\leq_{proj}$ Planar-B-PM

We start with the problem of determining whether there is an $s \rightsquigarrow t$ path in a directed forest G; this is logspace-complete under projections ([14, 25]). Since L is closed under complement, Directed Forest Unreachability is also complete for L under projections. Given an instance (G, s, t), first construct the split graph G' as described in Section 2. Then define H_1 to be the undirected version of G' and H_2 to be $H_1 \cup \{(s, t)\}$. From the properties of the split graph, since G was a forest, H_1 and H_2 are planar bipartite. Also the construction involves simple projections; it is FO-uniform.

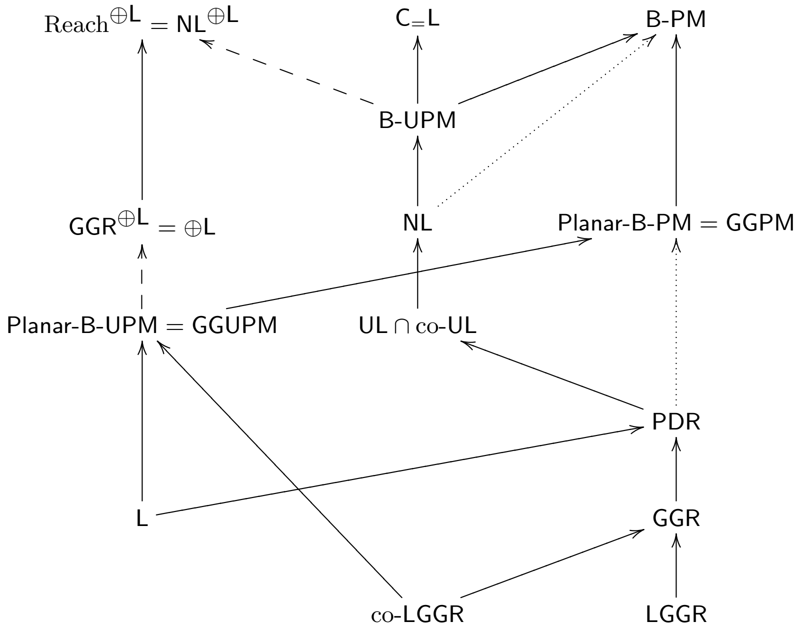

Fig. 4. Bipartite perfect matchings, Uniqueness, and Reachability: the overall picture

Now, as in [13, 23], for every $s \rightsquigarrow t$ path in G, the alternate edges of the corresponding path in H, along with edges of the form (v_{in}, v_{out}) for vertices v not on the path, form a perfect matching in H_1 and H_2. H_1 has no other matching, H_2 has one more matching, namely, the added (s, t) edge along with all the edges of the form (v_{in}, v_{out}). Thus H_1 has no perfect matching if and only if H_2 has exactly one perfect matching if and only if (G, s, t) is not in Forest-Reachability. This gives a projection from Forest-Unreachability to Planar-B-PM and Planar-B-UPM.

7.2 co-LGGR$\leq_{proj}$ Planar-B-UPM; GGR$\leq_{proj}$ Planar-B-PM

Consider the construction in Section 7.1, when applied to an arbitrary instance (G, s, t) of Reachability (instead of to a directed forest). To obtain a reduction to Planar-B-UPM, the resulting H_2 should be planar, bipartite, and have just one perfect matching more than the number of paths in G. If G has a bimodal embedding, then G' is planar bipartite and H_2 is bipartite. If the bimodal embedding of G has s and t on the same face, then H_2 is also planar. If G is acyclic, then every $s \rightsquigarrow t$ path in G gives rise to a distinct perfect matching in H_2, and the only other matching in H_2 is the canonical matching M described above. So, for any instance of Reachability satisfying the conditions of being planar bimodal acyclic, co-Reachability reduces to Planar-B-UPM. In particular, all these conditions are satisfied by instances of LGGR; hence co-LGGR$\leq_{proj}$ Planar-B-UPM.

If we want to test only the existence of a matching, we can afford to start with instances of GGR rather than LGGR. We assume that the instance G has s and t on the same face (as shown in [4], this is without loss of generality), and that the embedding is bimodal. (To achieve the latter, we ensure that every vertex v has degree at most 3: we replace v by a 3×4 sub-grid-cycle and attach the eight possible edges incident on v to different points on the cycle. This preserves reachability.) Now each $s \rightsquigarrow t$ path yields at least one perfect matching in H_1, while if G has no $s \rightsquigarrow t$ path, then H_1 has no perfect matching. Thus co-GGR reduces to Planar-B-PM. But it is known that co-GGR is equivalent to GGR [4]. Hence GGR reduces to Planar-B-PM.

7.3 Planar-B-UPM$\leq_m^{\log}$ GGUPM; Planar-B-PM$\leq_m^{\log}$ GGPM

We describe a parsimonious (in the number of perfect matchings) reduction from planar bipartite graphs to grid graphs. This implies both the claimed results.

Let G be the planar bipartite graph with bipartition (X, Y). Assume, without loss of generality, that every vertex has degree at most 3. (If not, then using the logspace construction described in Section 3 of [26], one can obtain such a graph preserving planarity, bipartiteness and number of perfect matchings).

Apply the grid embedding technique of [4], with the following modifications. Double the size of the coarse grid and then place the vertices of the bipartition X on the grid points $(2i, 2j)$ and those of Y on the grid points $(2i, 2j+1)$. Also let the size of the fine grid be $(2n+1) \times 2n$. This will ensure that every edge of the original graph has now become an odd length path. The rest of the construction is similar and can be done in L, giving a grid graph G' with the same number of perfect matchings as in G. (If edge $e = (u, v)$ was in the matching of G then the odd edges along the $u \rightsquigarrow v$ path are used in the matching of G', otherwise the even edges are put in the matching of G'.)

7.4 Planar-B-UPM is in $\oplus$L

In [23], an $\mathsf{NL}^{\oplus\mathsf{L}}$ algorithm for B-UPM is described. Given a bipartite graph G, it proceeds in two stages. In the first stage, an $\mathsf{L}^{\oplus\mathsf{L}}$ procedure either constructs some perfect matching M, or detects that G is not in B-UPM. In the second stage, an NL procedure, with oracle access to M, verifies that M is indeed unique.

We show below that if G is planar and bipartite, then the second stage can be performed in $\mathsf{L}^{\mathsf{GGR}}$. Since GGR is known to be in UL$\cap$co-UL [11] which is contained in $\oplus$L, and since $\oplus\mathsf{L}^{\oplus\mathsf{L}} = \mathsf{L}^{\oplus\mathsf{L}} = \oplus\mathsf{L}$ ([21]), it then follows that Planar-B-UPM is in $\oplus$L.

Given bipartite $G = (V, E)$ and a perfect matching M in it, consider the auxiliary directed graph H defined as follows: $H = (V, E')$ where $E' = \{\langle i, j \rangle \mid$ for some $k \in V, (i, k) \in M$ and $(k, j) \in E \setminus M\}$. As argued in [23], M is not unique in G if and only if there exists a directed cycle in H, the auxiliary graph. That is, $G \notin$ UPM if and only if there exists an edge (u, w) in H such that there exists a directed path from w to u in $H \setminus \{\langle u, w \rangle\}$. Thus the problem reduces to several reachability questions in H (one for each edge in H). We show below that if G is planar, then H is also planar. Since PDR is in UL∩co-UL by [11], it follows that testing uniqueness of M in G is in $\mathsf{L}^{\mathsf{UL}\cap\text{co-}\mathsf{UL}} = \mathsf{UL} \cap \text{co-}\mathsf{UL}$.

Lemma 1. *The auxiliary graph H of a planar bipartite graph G, with respect to any perfect matching M, is planar.*

Proof. Since G is bipartite, say with bipartitions V_A, V_B, H does not contain any edges with one endpoint each in V_A and V_B. Thus it suffices to prove that the sub-digraphs of H induced by V_A and V_B, say D_A and D_B respectively, are both planar. We show planarity for D_A, that for D_B follows by symmetry.

Let (u, v) be an edge in M such that $u \in V_A$ and $v \in V_B$. Then the outgoing edges from u in D_A are exactly the set of edges $\langle u, w\rangle$ such that $(w, v) \in E$ and $w \neq u$. Suppose $u = w_0, w_1, \ldots, w_{d-1}$ are the vertices adjacent to v in clockwise order, in the given planar embedding of G. Remove all the edges in G incident on v and instead join each pair of vertices $w_i, w_{(i+1) \bmod d}$ by a curve $C_i(v)$ not intersecting with the rest of the graph. See Figure 5.

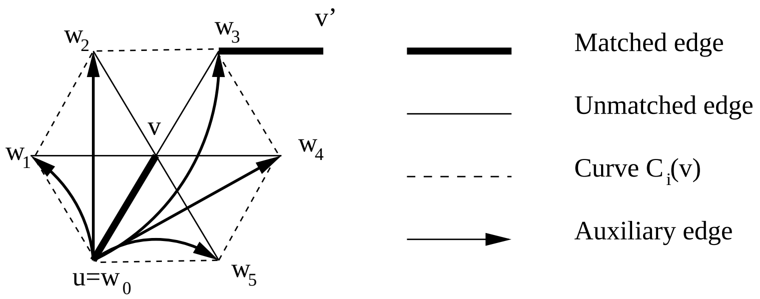

Fig. 5. Drawing the Auxiliary Edges Without Intersection

Then the auxiliary edges (u, w_j) $(1 \leq j \leq d-1)$ can all be drawn as non-intersecting curves within the closed region bounded by the curves C_i. Notice that $C_i(v)$ will be the same as $C_j(v')$ for v' at distance two from v in G, for some i, j — see the figure above. We can apply the same procedure for each vertex $v \in V_B$ without causing any intersection between the curves $C_i(v)$ and the edges of the digraph D_A and between the edges of D_A. Removing the curves $C_i(v)$ we get a planar embedding of D_A. □

7.5 Unique minimum weight Planar-B-UPM is hard for GGR

For the purpose of this section alone, the weight of a matching is the *sum* of the weights of its constituent edges.

Let (G, s, t) be the GGR instance; as discussed in Section 7.2, we can assume that G is bimodal and has s and t on the external face. We now assign weights to the edges of G according to the weighting scheme of [11] to get a graph G'; this weighting scheme has the property that $s \rightsquigarrow_G t \iff s \rightsquigarrow_{G'} t \iff$ the minimum weight $s \rightsquigarrow_{G'} t$ path is unique. Now construct $H = \text{Split}(G')$, copying the weight of an edge (u, v) in G' to the edge (u_{out}, v_{in}) of H and assigning weight zero to all the edges of the form (v_{in}, v_{out}). H is a planar bipartite graph and can be obtained via simple projections.

If $(G, s, t) \notin \mathsf{GGR}$, then it is easy to see that H has *no* perfect matching.

If $(G, s, t) \in \mathsf{GGR}$, then the unique minimum-weight path $\rho : s \rightsquigarrow_{G'} t$ can be extended to a perfect matching M_ρ in H, where $M_\rho = \{(u_{out}, v_{in}) \mid \langle u, v\rangle \in \rho_{G'}\} \cup \{(v_{in}, v_{out}) | v \in G' \text{and } v \notin \rho\}$ of the same weight. Since all (v_{in}, v_{out}) edges in H have weight 0, it is easy to see that this matching is the unique minimum-weight matching in H.

8 Discussion

In this paper, we have examined the complexity of computing the determinant or permanent of integer matrices when the associated graphs are planar, with or without additional conditions of bipartiteness or bimodality. We have also examined the complexity of testing existence and unique existence of perfect matchings in planar bipartite graphs.

Some of our results are expected, in that they were conjectured to be true and merely awaiting formal proof. But some are indeed quite surprising. For instance, the hardness of $\oplus\mathsf{LGGR}$ for $\oplus\mathsf{L}$ is unexpected given that the best known lower bound for LGGR is just NC^1.

In more recent work [17], it has been shown that Planar-B-UPM is in fact in the complexity class SPL, that is contained in $\oplus\mathsf{L}$. This thus improves our bound from Section 7.4.

Acknowledgement

The authors gratefully thank the referees for detailed comments that helped improve the presentation in the paper.

References

1. E. Allender. Arithmetic circuits and counting complexity classes. In J. Krajicek, editor, *Complexity of Computations and Proofs*, Quaderni di Matematica Vol. 13, pages 33–72. Seconda Universita di Napoli, 2004. An earlier version appeared in the Complexity Theory Column, SIGACT News 28, 4 (Dec. 1997) pp. 2-15.
2. E. Allender, D. A. M. Barrington, T. Chakraborty, S. Datta, and S. Roy. Planar and grid graph reachability problems. *Theory of Computing Systems*, pages 675–723, 2009.
3. E. Allender, R. Beals, and M. Ogihara. The complexity of matrix rank and feasible systems of linear equations. *Computational Complexity*, 8(2):99–126, 1999.

4. E. Allender, S. Datta, and S. Roy. The directed planar reachability problem. In *Proc. 25th FSTTCS, LNCS vol. 3821*, pages 238–249, 2005.
5. E. Allender and M. Mahajan. The complexity of planarity testing. *Information and Computation*, 189(1):117–134, 2004.
6. E. Allender, K. Rheinhardt, and S. Zhou. Isolation, matching and counting: uniform and nonuniform upper bounds. *Journal of Computer and System Sciences*, 59:164–181, 1999.
7. S. Arora and B. Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.
8. D. Barrington. Bounded-width polynomial size branching programs recognize exactly those languages in NC^1. *Journal of Computer and System Sciences*, 38:150–164, 1989.
9. D. A. M. Barrington. Grid graph reachability problems. Talk presented at Dogstuhl Seminar on Complexity of Boolean funcions, Seminar Number 02121, 2002.
10. C. Bourke, R. Tewari, and N. V. Vinodchandran. Directed planar reachability is in unambiguous logspace. In *Proceedings of 22nd IEEE Conference on Computational Complexity CCC*, pages 217–221, 2007.
11. C. Bourke, R. Tewari, and N. V. Vinodchandran. Directed planar reachability is in unambiguous log-space. *ACM Transactions on Computation Theory*, 1(1):1–17, 2009.
12. T. Chakraborty and S. Datta. One-input-face MPCVP is hard for L, but in LogDCFL. In *Proc. of 26th FST TCS Conference, LNCS vol. 4337*, pages 57–68, 2006.
13. A. Chandra, L. Stockmeyer, and U. Vishkin. Constant depth reducibility. *SIAM Journal on Computing*, 13(2):423–439, 1984.
14. S. A. Cook and P. McKenzie. Problems complete for L. *Journal of Algorithms*, 8:385–394, 1987.
15. C. Damm. $\mathrm{DET}=\mathrm{L}^{(\#\mathrm{L})}$. Technical Report Informatik–Preprint 8, Fachbereich Informatik der Humboldt–Universität zu Berlin, 1991.
16. S. Datta, R. Kulkarni, N. Limaye, and M. Mahajan. Planarity, determinants, permanents, and (unique) matchings. In *Proceedings of CSR, LNCS vol. 4649*, pages 115–126, 2007.
17. S. Datta, R. Kulkarni, and S. Roy. Deterministically isolating a perfect matching in bipartite planar graphs. In *Proceedings of STACS*, pages 229–240, 2008.
18. A. L. Delcher and S. R. Kosaraju. An NC algorithm for evaluating monotone planar circuits. *SIAM Journal of Computing*, 24(2):369–375, 1995.
19. M. E. Dyer and A. M. Frieze. Planar 3DM is NP-complete. *J. Algorithms*, 7(2):174–184, 1986.
20. K. Hansen. Constant width planar computation characterizes ACC^0. *Theory of Computing Systems*, 39(1):79–92, 2006.
21. U. Hertrampf, S. Reith, and H. Vollmer. A note on closure properties of logspace MOD classes. *Information Processing Letters*, 75(3):91–93, 2000.
22. W. Hesse, E. Allender, and D. Barrington. Uniform constant-depth threshold circuits for division and iterated multiplication. *Journal of Computer and System Sciences*, 65:695–716, 2002.
23. T. M. Hoang, M. Mahajan, and T. Thierauf. On the bipartite unique perfect matching problem. In *Proc. 33rd International Colloquium on Automata, Languages and Programming (ICALP)*, pages 453–464. Springer, 2006. LNCS 4051.
24. H. B. Hunt, III, M. V. Marathe, V. Radhakrishnan, and R. E. Stearns. The complexity of planar counting problems. *SIAM Journal on Computing*, 27(4):1142–1167, 1998.

25. N. Immerman. Languages which capture complexity classes. *SIAM J. Comput.*, 4:760–778, 1987.
26. R. Kulkarni, M. Mahajan, and K. R. Varadarajan. Some perfect matchings and perfect half-integral matchings in NC. *Chicago Journal of Theoretical Computer Science*, 2008(4), September 2008.
27. N. Limaye, M. Mahajan, and J. M. N. Sarma. Upper bounds for monotone planar circuit value and variants. *Computational Complexity*, page to appear, 2009.
28. M. Mahajan, P. R. Subramanya, and V. Vinay. The combinatorial approach yields an NC algorithm for computing Pfaffians. *Discrete Applied Mathematics*, 143(1-3):1–16, September 2004.
29. M. Mahajan and V. Vinay. Determinant: combinatorics, algorithms, complexity. *Chicago Journal of Theoretical Computer Science* `http://www.cs.uchicago.edu/publications/cjtcs`, 1997:5, Dec 1997.
30. B. Mohar and C. Thomassen. *Graphs on Surfaces.* John Hopkins University Press, Maryland, 2001.
31. C. M. Papadimitriou. *Computational Complexity.* Addison-Wesley, Reading, Massachusetts, 1994.
32. O. Reingold. Undirected *st*-connectivity in logspace. In *Proc. 37th STOC*, pages 376–385, 2005.
33. K. Reinhardt and E. Allender. Making nondeterminism unambiguous. *SIAM J. Comp.*, 29:1118–1131, 2000.
34. S. Toda. Counting problems computationally equivalent to the determinant. Technical Report CSIM 91-07, Dept of Comp Sc. & Information Mathematics, Univ. of Electro-Communications, Chofu-shi, Tokyo, 1991.
35. S. Vadhan. The complexity of counting in sparse, regular, and planar graphs. *SIAM Journal on Computing*, 31(2):398–427, 2001.
36. L. G. Valiant. The complexity of computing the permanent. *Theoretical Computer Science*, 8:189–201, 1979.
37. L. G. Valiant. Why is boolean complexity theory difficult? In M. S. Paterson, editor, *Boolean Function Complexity.* Cambridge University Press, 1992. London Mathematical Society Lecture Notes Series 169.
38. 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.
39. V. Vinay. *Semi-unboundedness and complexity classes.* PhD thesis, Indian Institute of Science, Bangalore, July 1991.
40. M. Xia and W. Zhao. #3-regular bipartite planar vertex cover is #P-complete. In *TAMC*, pages 356–364, 2006.
41. H. Yang. An NC algorithm for the general planar monotone circuit value problem. In *Proceedings of 3rd IEEE Symposium on Parallel and Distributed Processing*, pages 196–203, 1991.
42. V. Zankó. #P-completeness via many-one reductions. *Int. J. Found. Comput. Sci.*, 2(1):77–82, 1991.