Reachability in $O(\log n)$ Genus Graphs is in Unambiguous Logspace

Chetan Gupta
Indian Institute of Technology, Kanpur, India
gchetan@cse.iitk.ac.in

Vimal Raj Sharma
Indian Institute of Technology, Kanpur, India
vimalraj@cse.iitk.ac.in

Raghunath Tewari
Indian Institute of Technology, Kanpur, India
rtewari@cse.iitk.ac.in

Abstract

We show that given an embedding of an $O(\log n)$ genus graph G and two vertices s and t in G, deciding if there is a path from s to t in G is in unambiguous logarithmic space.

Unambiguous computation is a restriction of nondeterministic computation where the nondeterministic machine has at most one accepting computation path on each input. An important fundamental question in computational complexity theory is whether this is an actual restriction or are unambiguous computations as powerful as general nondeterminism. We investigate this problem in the domain of logarithmic space bounded computations, where the corresponding unambiguous and general nondeterministic classes are UL and NL respectively.

In 1997 Reinhardt and Allender showed that NL and UL are equal in a non-uniform model. More specifically they showed that if one can efficiently construct an $O(\log n)$-bit min-unique weight function for a graph, then these classes are equal unconditionally as well. In other words, they gave a UL algorithm to solve reachability in graphs with a min-unique weight assignment. Using this approach reachability in various classes of graphs such as planar graphs, constant genus graphs, minor free graphs, etc., have been shown to be in UL by devising min-unique weight functions for those classes.

In this paper we improve these results by constructing a min-unique weight function for $O(\log n)$ genus graphs. We define signature of a path in a graph as the parity of the number of crossings of that path with respect to each handle of the surface on which the graph is embedded. We construct our weight function in two steps. First we ensure that between any pair of vertices, amongst all paths having the same signature, the minimum weight path is unique. Now since in a genus g graph there are 2^{2g} many possible signatures, we use the hashing scheme of Fredman, Komlós and Szemerédi to isolate a unique minimum weight path among these 2^{2g} many paths isolated in the first step.

2012 ACM Subject Classification Theory of computation → Complexity classes

Keywords and phrases logspace unambiguity, high genus, path isolation

Digital Object Identifier 10.4230/LIPIcs.STACS.2019.34

Acknowledgements The authors would like to thank Ministry of Electronics and IT, India for supporting this research through the Visvesvaraya PhD and YFRF program. The third author would also like to thank DST for providing funding support.

1 Introduction

Deciding reachability between a pair of vertices in a graph is an important problem in computational complexity theory. Directed graph reachability characterizes the complexity of the class nondeterministic logspace (NL) and undirected graph reachability characterizes the

36th International Symposium on Theoretical Aspects of Computer Science (STACS 2019).
Editors: Rolf Niedermeier and Christophe Paul; Article No. 34; pp. 34:1–34:13
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

complexity of the class deterministic logspace (L). The latter follows due to a seminal result by Reingold in 2005 [24]. Several other variants of this problem characterize the complexity of other complexity classes [6, 13, 7].

Unambiguous computation is a natural restriction of nondeterministic computation where for every input the Turing machine can have at most one accepting computation path. In the domain of logarithmic space, this defines the class unambiguous logspace (UL) of languages for which there are nondeterministic logspace bounded Turing machines that have exactly one accepting path for every input in the language and zero accepting path otherwise. The class UL was introduced in [9] and subsequently its properties were also studied in the paper [4]. The relation between NL and UL was not well understood till that point. In 1997, Reinhardt and Allender showed that NL and UL are equal in a non-uniform setting [25]. Subsequently, it was shown that if deterministic linear space functions cannot be computed by $2^{\epsilon n}$ sized circuits then NL = UL [3]. Both these results gave evidence that most likely the classes are the same unconditionally. Recently directed graph reachability was shown to be decidable by an unambiguous algorithm running in polynomial time and using $O(\log^2 n)$ space [20]. The space bound was improved to $O(\log^{1.5} n)$ in a subsequent result [29].

A graph G is said to be *min-unique* with respect to a weight function w if for every pair of vertices in G there is at most one minimum weight path from one vertex to the other with respect to w. We will call such a weight function a *path isolating* weight function. Min-uniqueness has been studied in several papers [30, 16, 25]. Reinhardt and Allender showed that if graphs in a class of graphs are min-unique with respect to an $O(\log n)$ bit weight function then deciding reachability for that class of graphs is in UL [25]. They also gave a UL algorithm to check if a graph is min-unique.

The technique of assigning weights to isolate a combinatorial structure is not specific to the graph reachability problem and has been applied to other computational problems as well. In a sequence of results it has been shown that assigning an efficiently computable weight function to a graph class such that the minimum weight perfect matching is unique with respect to the weight function, results in an efficient parallel algorithm for computing matching for the respective class of graphs [11, 12, 5, 14]. Another area which has successfully used the idea of isolating structures to obtain better upper bounds is polynomial identity testing. More specifically, researchers have used basis isolating weight assignment to obtain polynomial and quasi-polynomial time algorithms for certain restrictions of the polynomial identity testing problem [1, 19, 18].

Observe that devising a UL algorithm for directed graph reachability would show that NL = UL, since directed graph reachability is complete for the class NL. Although the NL versus UL has been elusive so far, partial progress has been made towards solving this problem. For several classes of directed graphs, the reachability problem has been shown to be in UL – such as layered grid graphs [2], planar graphs [8], constant genus graphs [21, 12], graphs with polynomially many paths from the source to all other vertices [23], $K_{3,3}$-free and K_5-free graphs [27, 5]. The techniques involve either an efficient construction of a path isolating weight function or reduction to reachability in a graph class for which the problem is already known to be in UL.

Reachability in positive genus graphs is a natural extension of planar reachability. Allender et al. showed that reachability in 1 genus graphs can be reduced to planar reachability [2]. After planar reachability was shown to be in UL, reachability in constant genus graphs was reduced to reachability in planar graphs [21]. Later a path isolating weight function was also given for constant genus graphs [12]. Prior to our result, the best known nondeterministic space upper bound for reachability in non-constant genus graphs was nothing better than general directed graphs. The question of whether reachability in $\omega(1)$ genus graphs belongs to UL or not has been open for almost a decade.

1.1 Our Result

In this paper, we make progress towards understanding the space complexity of directed graph reachability and show the following result.

▶ **Theorem 1.** *Given a polygonal schema of an $O(\log n)$ genus directed graph G, deciding reachability in G is in* $\mathsf{UL} \cap \mathsf{coUL}$.

Given a genus g graph, in the first stage, we give an $O(\log n)$ bit weight function w_{pl} which is essentially the same weight function as defined in [26] and another weight function w_{len} which gives weight 1 to every edge in the graph. Weight function w_{len} ensures that minimum weight paths among all pairs of vertices are of minimum length as well. We then show that between every pair of vertices in the graph, the number of minimum weight "topologically unequivalent" paths is at most $2^{O(g)}$. For this, we define a notion called *signature* which allows us to classify topologically equivalent paths. We show that topologically equivalent paths are very similar to paths in planar graphs and therefore we can borrow the machinery for path isolation in planar graphs here as well. In the second stage, we use the hashing scheme of Fredman, Komlós and Szemerédi [15] to compute an $O(\log n + g)$ bit weight function w_{fks} with respect to which only one among the $2^{O(g)}$ many paths of the first stage gets the minimum weight value.

When g is $O(\log n)$ the number of such minimum weight paths produced in the first stage is at most polynomial in n. Thereafter by combining the weight functions w_{len}, w_{pl} and w_{fks} we get an $O(\log n)$ bit weight function with respect to which the graph is min-unique. We then apply Reinhardt and Allender's algorithm to get a UL algorithm for $O(\log n)$ genus reachability.

1.2 Organization of the Paper

The rest of the paper is organized as follows. In Section 2 we define the notations and framework of our problem. We discuss different representations of high genus graphs and how to efficiently obtain a representation that is suitable for our purpose. We also state results from earlier work that we use in this paper. In Section 3 we prove the main result by giving a min-unique weight function.

2 Preliminaries

Let $G = (V, E)$ be a directed graph on n vertices and m edges. Let uv denote an edge directed from u to v. A *weight function* is a map $w : E \to \mathbb{Z}$ which maps every edge in G to an integer. A weight function w is said to be *skew-symmetric* if for every edge uv, $w(uv) = -w(vu)$. For a set of edges S, $w(S) = \sum_{e \in S} w(e)$. We can think of different structures in a graph such as path, walk, cycle as sets of edges and define the weight of the structure accordingly.

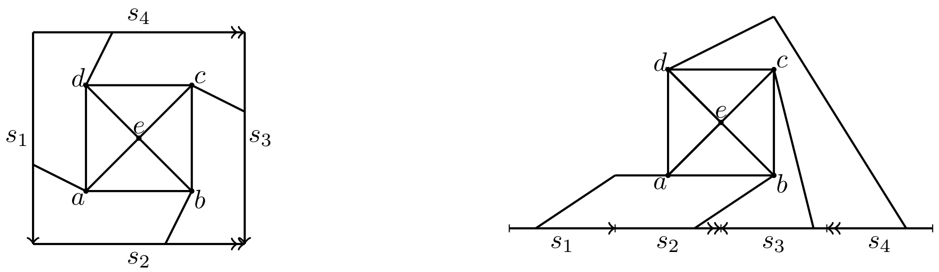
2.1 Representation of High Genus Graph

A genus g surface is a sphere with g handles on it. The *genus of a graph* is the minimum genus surface on which the graph can be embedded without any edge crossings. Such an embedding is also called a *2-cell embedding*. Since we are dealing with graphs embedded on surfaces, it is important to specify how the input graph is represented. Given a graph, computing its genus is NP-hard [28]. To the best of our knowledge, no PTAS is known either to compute the genus of a graph. So in accordance with the convention followed by earlier papers that deals with problems on bounded genus graphs, we also assume a suitable representation of the input graph [22, 17]. We use a representation similar to the one used by Mahajan and Varadarajan [22].

Given a genus g graph G we consider an embedding of G inside a polygon S with $4g$ sides, $s_1, s_2, \ldots, s_{4g}$. We refer to these as the *segments* of S. Moreover, we assume there is no vertex on the boundary of the polygon. The segments s_{4k+1} and s_{4k+2} are directed in anti-clockwise and segments s_{4k+3} and s_{4k+4} are directed in clockwise direction. The segments s_{4k+1} and s_{4k+3} form a pair together such that an edge can come into one of them and go out from another. Similarly, segments s_{4k+2} and s_{4k+4} form a pair. Also, if an edge is the jth edge crossing a segment s_i from head to tail then it will be the jth edge crossing the paired segment of s_i from tail to head. Pairs (s_{4k+1}, s_{4k+3}) and (s_{4k+2}, s_{4k+4}) together constitute the ith handle of the sphere. We assume that we are provided with the combinatorial embedding of the graph G inside S and an ordering of the edges crossing each segment s_i. We also assume without loss of generality that an edge can cross a segment of the polygonal schema at most once. This is because an edge crossing multiple segments can be subdivided into several edges in logspace so that reachability is still preserved. We call this representation the *polygonal schema* of G.

Let E_S be the set of edges in G that cross some segment s_i. Then observe that $G_{\text{planar}} = G \setminus E_S$ is a planar graph. A *piecewise straight line embedding* of a planar graph is an embedding where vertices are integral coordinates and an edge is a piecewise straight line segment connecting its two end points such that no two edges intersect. Given the combinatorial embedding of a planar graph a piecewise straight line embedding of it can be constructed in logspace such that each edge consists of at most 4 segments [26].

For a genus g graph G, a *flat schema* is an embedding of G such that the polygon S is represented as a straight line segment parallel to the x-axis, the internal planar graph G_{planar} is given as a piecewise straight line embedding and each edge in E_S is drawn as a piecewise straight line segment such that no two edges cross each other. Moreover, all vertices and points where an edge crosses a segment are integral coordinates. See Figure 1 for an example of a flat schema of K_5.



(a) Polygonal schema of K_5.

(b) Flat schema of K_5.

Figure 1 Embeddings of K_5.

Given a polygonal schema of G, we can compute a piecewise straight line embedding of G_{planar} in logspace. Now using a similar idea we draw each edge in E_S as a piecewise straight line segment from its end vertices to the corresponding segments of S. We summarize this process in Lemma 2.

▶ **Lemma 2.** *Given a polygonal schema of a graph G with $4g$ segments we can construct a flat schema of G with $4g$ segments, having polynomially bounded integer coordinates in logspace.*

The weight function w_{pl} that we define in Section 3 is similar to the weight function in [26]. That is why we need edges to be piecewise straight line segments. Also the flat schema embedding is necessary because we want all edges parallel to the x-axis to have weight 0 with respect to w_{pl}.

2.2 Previous Work

Consider a genus g graph G embedded on a surface of genus g say Γ. A simple cycle C in G is called a *separating cycle* if cutting along C divides the surface into at least two parts. Otherwise C is called a *non-separating cycle.* We state a characterization of these cycles from Lemma 4 of Cabello and Mohar [10] and Lemma 10 of Datta et al. [12].

▶ **Theorem 3** ([10, 12]). *Consider a polygonal schema of a genus g graph. A cycle C in G is said to be surface separating if and only if C crosses each segment of the polygonal schema an even number of times. Moreover, if C is surface separating then with respect to each segment s_i, the cycle C alternates between coming into s_i and going out of it (if C crosses s_i at all).*

We next state the popular hashing result by Fredman, Komlós and Szemerédi.

▶ **Theorem 4** ([15]). *Let $S = \{x_1, x_2, \ldots, x_k\}$ be a set of $n-$bit integers. Then there exists a $O(\log n + \log k)$ bit prime number p so that for all $x_i \neq x_j \in S$, $x_i \bmod p \neq x_j \bmod p$.*

In Theorem 5 we state a slightly modified version of Reinhardt and Allender's result that would be useful for our purpose.

▶ **Theorem 5** ([25]). *There is a nondeterministic logspace Turing machine M that takes a tuple $\langle G, s, t, w\rangle$ as input where G is a directed graph on n vertices, s and t are vertices in G and w is an $O(\log n)$ bit edge weight function and outputs the following along a unique computation path while all other computation paths halt and reject:*

- Not Min-unique *if G is not min-unique with respect to w,*
- Yes *if G is min-unique with respect to w and there is a path from s to t in G, and*
- No *if G is min-unique with respect to w and there is no path from s to t in G.*

Finally, in Theorem 6 we state the relation between the area of a simple cycle in a planar graph and weight of the cycle with respect to a suitable weight function as shown by Tewari and Vinodchandran.

▶ **Theorem 6** ([26]). *Given a piece-wise straight line embedding of a planar graph G, there exists a logspace computable weight function w such that for any cycle C in G, we have $w(C) = 2 \cdot \mathrm{Area}(C)$ if C is a counter-clockwise cycle and $w(C) = -(2 \cdot \mathrm{Area}(C))$ if C is a clockwise cycle, where $\mathrm{Area}(C)$ is the area of the region enclosed by C.*

3 Isolating Paths in High Genus Graphs

In this section we show that graphs of logarithmic genus are min-unique with respect to an $O(\log n)$-bit weight function that can be computed by an unambiguous logspace machine. Using this weight function in combination with Theorem 5 we get a $\mathsf{UL} \cap \mathsf{coUL}$ algorithm for directed graph reachability in $O(\log n)$ genus graphs. Theorem 7 is the main technical result of this paper where we show the existence and computability of such a weight function.

▶ **Theorem 7.** *Given a genus g directed graph $G = (V, E)$ in terms of its flat schema, there exists an $O(\log n + g)$ bit weight function $w : E \to \mathbb{Z}$, such that for every $u, v \in V$, there exists a unique minimum weight path from u to v with respect to w, if v is reachable from u. Moreover, there is a nondeterministic $O(\log n + g)$ space algorithm that given G as input, outputs the weight function w along a unique computation path while all other paths halt and reject.*

Let $\mathcal{S} = \{s_1, s_2, \ldots, s_{4g}\}$ be the set of segments of the flat schema of G. We define a skew-symmetric weight function w_{pl} that gives non-zero weight to every surface separating cycle in G. For edges which do not cross any segment of the flat schema (we refer to them as *planar edges*), w_{pl} is same as the weight function defined in [26], and for edges which do cross some segment of the flat schema (we refer to them as *crossing edges*) we modify the weight function to be the sum of the weights of the two line segments of the edge. Formally, for an edge $e = uv$ we define the $w_{\text{pl}}(e)$ as:

$$w_{\text{pl}}(e) = \begin{cases} (y_{v_e} - y_{u_e})(x_{v_e} + x_{u_e}) & \text{if } e \text{ is a planar edge} \\ (y_{u'_e} - y_{u_e})(x_{u'_e} + x_{u_e}) + (y_{v_e} - y_{v'_e})(x_{v_e} + x_{v'_e}) & \text{if } e \text{ is a crossing edge} \end{cases}$$

where (x_{u_e}, y_{u_e}) and (x_{v_e}, y_{v_e}) are the coordinates of u and v respectively and $(x_{u'_e}, y_{u'_e})$ and $(x_{v'_e}, y_{v'_e})$ are the coordinates of intersection points of edge e with segments s_i and s_j respectively, assuming that edge e comes into s_i and goes out of s_j.

We also define another weight function w_{len} that assigns value one to every edge in the graph. That is $w_{\text{len}}(e) = 1$ for every edge $e \in G$. Let $w_{\text{comb}} = w_{\text{len}} \cdot n^{k_1} + w_{\text{pl}}$ be the weight function defined by combining w_{pl} and w_{len} for a large enough constant k_1. As a result the minimum weight path with respect to w_{comb} also has the minimum length.

We first show that every surface separating cycle has non-zero weight with respect to w_{pl}. The idea is to decompose every surface separating cycle into a set of planar cycles having the same orientation such that the weight of the original cycle is the sum of the weights of the planar cycles.

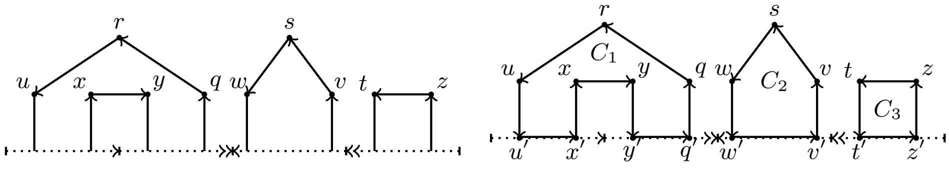
▶ **Lemma 8.** *Let C be a simple surface separating cycle of length at least 3 in G, then $w_{pl}(C) \neq 0$.*

Proof. A surface separating cycle can be of two types – one which does not intersect with any segment of the flat schema and the one which does. If C does not intersect any segment of the flat schema then C is a planar cycle. Hence $w_{\text{pl}}(C) \neq 0$ by [26].

Now consider the case where some edges of C cross the flat schema. From Theorem 3 we know that since C is a surface separating cycle, therefore, C alternates between going out and coming into the segments of the flat schema. Without loss of generality assume that the first edge of C crossing the flat schema going left to right, is coming into it. The other case is analogous.

For every edge $e = uv$ which crosses the boundary S of the flat schema we subdivide e into two directed edges $u_e u'_e$ and $v'_e v_e$, such that u'_e is the point at which uv comes into some segment s_i and v'_e is the point at which uv goes out of some segment s_j. Let C' be the cycle corresponding to C formed by this subdivision. By definition of w_{pl} we have $w_{\text{pl}}(uv) = w_{\text{pl}}(u_e u'_e) + w_{\text{pl}}(v'_e v_e)$ and hence $w_{\text{pl}}(C) = w_{\text{pl}}(C')$.

Let $x_1, x_2, \ldots x_{2t}$ be the set of intersection points of C' and S ordered from left to right. Add t dummy directed edges from x_{2i-1} to x_{2i} for all $1 \leq i \leq t$. This decomposes C' into a set of disjoint planar cycles $C_1, C_2, \ldots, C_k$ such that each C_i has the same orientation (counter-clockwise, since we assume the first edge is coming into S). See Figure 2 for an example.



(a) Surface separating cycle C.

(b) Decomposition of C into C_1, C_2 and C_3.

Figure 2 Decomposing a surface separating cycle into planar cycles.

By Theorem 6, $w_{\text{pl}}(C_i) = 2 \cdot \text{Area}(C_i)$ for each i. Moreover, since the C_i's have all the edges of C' plus some horizontal edges (the dummy edges) of zero weight, therefore $w_{\text{pl}}(C') = \sum_{i=1}^{k} w_{\text{pl}}(C_i) = 2 \cdot \sum_{i=1}^{k} \text{Area}(C_i)$. Therefore $w_{\text{pl}}(C) \neq 0$. ◀

We now show that the number of minimum weight paths with respect to w_{comb}, between any pair of vertices is at most 2^{2g}. We define classes of paths based on the number of times a path intersects each segment of the flat schema, and show that in each such class there is at most one minimum weight path.

Given a polygonal schema of a genus g graph G, by Lemma 2 we assume that we are provided with a flat schema of G having $4g$ segments. Let $\mathcal{T} = \{T_1, T_2, \ldots, T_{2g}\}$ be the set of segments of the flat schema such that no two elements of $\mathcal{T}$ are pairs of each other.

For a path P in G, define the *signature* of P, denoted as $\text{sign}(P)$, as a binary string $s = s_1 s_2 \ldots s_{2g}$ where $s_i = 1$ if P crosses T_i an odd number of times and $s_i = 0$ if P crosses T_i an even number of times. Clearly, the total number of different signatures are 2^{2g}. This definition can be similarly extended to cycles and walks as well.

For $0 \leq i \leq 2^l - 1$, let $\text{bin}(i)$ be the l-bit string that denotes the binary representation of i (if the binary representation has lesser than l bits then we prefix it with appropriate number of zeroes to make it l-bit long). Now consider $l = 2g$. For every pair of vertices u and v, we define 2^{2g} classes of paths $K_0^{uv}, K_1^{uv}, \ldots, K_{2^{2g}-1}^{uv}$ as follows:

$$K_i^{uv} = \{P \mid P \text{ is path from } u \text{ to } v \text{ and } \text{sign}(P) = \text{bin}(i)\}.$$

Note that if $P = P_1 P_2 \ldots P_k$ be a partition of a path P into subpaths, then $\text{sign}(P) = \text{sign}(P_1) \oplus \text{sign}(P_2) \oplus \ldots \text{sign}(P_k)$, where $\oplus$ is the bitwise XOR operator.

For a directed path P from x to y, let P^r represent the directed path from vertex y to x obtained by reversing the edges along the path P. Note that $\text{sign}(P) = \text{sign}(P^r)$.

▶ **Theorem 9.** *Let $G = (V, E)$ be a genus g graph embedded on a flat schema having $4g$ segments. Let u and v be two vertices in G and i be a non-negative integer less than or equal to $2^{2g} - 1$. Then in every class K_i^{uv} there exists at most one minimum weight path from u to v with respect to w_{comb}, that is, the weight function w_{comb} is min-isolating for each set K_i^{uv}.*

Proof. Assume that P_1 and P_2 are two minimum weight paths in K_i^{uv} with respect to w_{comb}. Then $w_{\text{pl}}(P_1) = w_{\text{pl}}(P_2)$ and $w_{\text{len}}(P_1) = w_{\text{len}}(P_2)$. Consider two cases – when P_1 and P_2 have common intermediate vertices and when they do not.

Case 1: P_1 and P_2 do not have any common intermediate vertices. We will show that P_1 and P_2^r together form a surface separating cycle. Let $C = P_1 P_2^r$ be the directed cycle formed by taking P_1 followed by P_2^r. Since P_1 and P_2 do not have any common intermediate vertices therefore C is a simple cycle. Recall that w_{pl} is a skew-symmetric weight function so $w_{\text{pl}}(P_2^r) = -w_{\text{pl}}(P_2)$. Therefore,

$$\begin{aligned} w_{\text{pl}}(C) &= w_{\text{pl}}(P_1) + w_{\text{pl}}(P_2^r) \\ &= w_{\text{pl}}(P_1) - w_{\text{pl}}(P_2) \\ &= 0 \qquad \text{(since } P_1 \text{ and } P_2 \text{ have the same minimum weight)} \end{aligned}$$

Also since P_1 and P_2 belong to K_i^{uv} we have that $\text{sign}(P_1) = \text{sign}(P_2) = \text{sign}(P_2^r)$. Therefore we get that $\text{sign}(C) = 0$ (the all zeroes vector). By Theorem 3 we have that C is a surface separating cycle and thus by Lemma 8 $w_{\text{pl}}(C)$ cannot be zero. Thus we get a contradiction. Therefore Case 1 cannot occur.

Case 2: P_1 and P_2 have common intermediate vertices. Note that at any common intermediate vertex, the paths P_1 and P_2 can either cross each other or tangentially touch each other without crossing. We refer to the former as *crossing vertex* and the latter as *grazing vertex.*

We will show that the closed walk formed by P_1 and P_2^r reduces to a surface separating simple cycle such that the weight of the closed walk is almost equal to that of the cycle.

▶ **Lemma 10.** *Let P_1 and P_2 be two minimum weight paths from u to v with $u_1, u_2, \dots u_l$ being the set of common intermediate vertices. Then $u_1, u_2, \dots u_l$ must occur in the same order in both paths P_1 and P_2.*

Proof. Lemma is trivially true when $l = 1$. So, let $l > 1$. Suppose u_i occurs before u_j in P_1 and u_j occurs before u_i in P_2, for $i < j$. Let a, b and c be the lengths of path P_1 from u to u_i, u_i to u_j and u_j to v respectively. Similarly, let d, e and f be the lengths of path P_2 from u to u_j, u_j to u_i and u_i to v respectively. Since $w_{\text{len}}(P_1) = w_{\text{len}}(P_2)$ we have

$$a + b + c = d + e + f. \tag{1}$$

If $d < a + b$ then taking P_2 from u to u_j and P_1 from u_j to v gives us a shorter length path from u to v than either P_1 or P_2. Similarly, if $d > a + b$ we can construct a shorter length from u to v as well. Hence we can assume that

$$d = a + b. \tag{2}$$

Using analogous argument we can assume

$$f = b + c. \tag{3}$$

Now adding Equations 2 and 3 we have

$$a + 2b + c = d + f. \tag{4}$$

Now since b and e are non zero, Equations 1 and 4 contradict each other. Hence $u_1, u_2, \dots u_l$ occur in the same order in paths P_1 and P_2. ◀

▶ **Lemma 11.** *Let P_1 and P_2 be two paths in K_i^{uv} having crossing vertices $v_1, v_2, \dots v_k$, such that these vertices divide P_1 and P_2 into $k+1$ sub-paths $P_1^1, P_1^2, \dots P_1^{k+1}$ and $P_2^1, P_2^2, \dots P_2^{k+1}$ respectively (as shown in Figure 3). Then the paths $P' = P_1^1 P_2^2 \dots P_i^{k+1}$ and $P'' = P_2^1 P_1^2 \dots P_j^{k+1}$ (where $i = 1$ and $j = 2$ if k is even and $i = 2$ and $j = 1$ if k is odd) belong to the same class.*

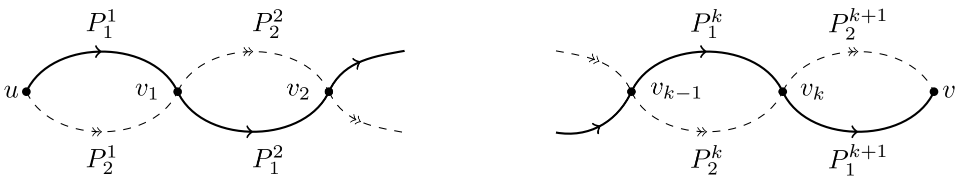


Figure 3 Crossings of paths P_1 (bold line) and P_2 (dashed line) at k many points.

The intuition of Lemma 11 is that if P_1 and P_2 cross each other then the two paths obtained by taking the "above" and "below" portions of these two paths have the same signature.

Proof. Since P_1 and P_2 belong to same class, we have

$$\begin{aligned} \text{sign}(P_1) &= \text{sign}(P_2) \\ \text{sign}(P_1^1) \oplus \text{sign}(P_1^2) \oplus \ldots \oplus \text{sign}(P_1^{k+1}) &= \text{sign}(P_2^1) \oplus \text{sign}(P_2^2) \oplus \ldots \oplus \text{sign}(P_2^{k+1}) \end{aligned}$$

We know that if a, b, c, d are binary strings of equal length then $a \oplus b = c \oplus d \Leftrightarrow a \oplus c = b \oplus d$. Therefore by rearranging the terms we get

$$\begin{aligned} \text{sign}(P_1^1) \oplus \text{sign}(P_2^2) \oplus \ldots \oplus \text{sign}(P_i^{k+1}) &= \text{sign}(P_2^1) \oplus \text{sign}(P_1^2) \oplus \ldots \oplus \text{sign}(P_j^{k+1}) \\ \text{sign}(P') &= \text{sign}(P'') \end{aligned}$$

Hence P' and P'' belong to the same class. ◀

Note that P' and P'' need not belong to the same class as P_1 and P_2. We define $P_i^{j,k} (j < k)$ as a shorthand for path $P_i^j P_i^{j+1} \ldots P_i^k$.

▶ **Lemma 12.** *Let P_1 and P_2 are two minimum weight paths in K_i^{uv} having crossing vertices $v_1, v_2, \ldots v_k$, then $w_{pl}(P_1^i) = w_{pl}(P_2^i)$, for all $i, 1 \leq i \leq k+1$. Additionally, for the closed walk $C' = P'(P'')^r$ we have that $w_{pl}(C') = 0$ and $\text{sign}(C') = 0$ (where P' and P'' are as defined in Lemma 11).*

Proof. Assume that there exists some j $(1 \leq j \leq k+1)$ such that j is the smallest index, where $w_{\text{pl}}(P_1^j) \neq w_{\text{pl}}(P_2^j)$. Without loss of generality assume that $w_{\text{pl}}(P_1^j) < w_{\text{pl}}(P_2^j)$. Now consider path $\widetilde{P} = P_2^{1,j-1} P_1^j P_2^{j+1,k+1}$. $\widetilde{P}$ is a path from u to v and by construction $w_{\text{pl}}(\widetilde{P}) < w_{\text{pl}}(P_2)$. This is a contradiction since P_2 is a minimum weight path from u to v. Therefore for all i we have $w_{\text{pl}}(P_1^i) = w_{\text{pl}}(P_2^i)$.

Now,

$$\begin{aligned} w_{\text{pl}}(C') &= w_{\text{pl}}(P') + w_{\text{pl}}((P'')^r) \\ &= w_{\text{pl}}(P_1^1) + w_{\text{pl}}(P_2^2) + w_{\text{pl}}(P_1^3) + \ldots + w_{\text{pl}}(P_i^{k+1}) + \\ &\quad w_{\text{pl}}((P_2^1)^r) + w_{\text{pl}}((P_1^2)^r) + w_{\text{pl}}((P_2^3)^r) + \ldots + w_{\text{pl}}((P_j^{k+1})^r) \\ &= (w_{\text{pl}}(P_1^1) + w_{\text{pl}}((P_2^1)^r)) + (w_{\text{pl}}(P_2^2) + w_{\text{pl}}((P_1^2)^r)) + (w_{\text{pl}}(P_1^3) + w_{\text{pl}}((P_2^3)^r)) + \\ &\quad \ldots + (w_{\text{pl}}(P_i^{k+1}) + w_{\text{pl}}((P_j^{k+1})^r)) \\ &= (w_{\text{pl}}(P_1^1) - w_{\text{pl}}(P_2^1)) + (w_{\text{pl}}(P_2^2) - w_{\text{pl}}(P_1^2)) + (w_{\text{pl}}(P_1^3) - w_{\text{pl}}(P_2^3)) + \\ &\quad \ldots + (w_{\text{pl}}(P_i^{k+1}) - w_{\text{pl}}(P_j^{k+1})) \\ &= 0. \end{aligned}$$

By Lemma 11 we have that P' and P'' belong to the same class. Hence $\text{sign}(C') = \text{sign}(P') \oplus \text{sign}(P'') = 0$. ◀

We now argue that there is a simple cycle (say $\widehat{C}$) such that C' and $\widehat{C}$ are infinitesimally separated. Hence their signatures are the same. However the weight function w_{pl} depends on the coordinates of an edge, therefore $w_{\text{pl}}(C')$ and $w_{\text{pl}}(\widehat{C})$ are nearly the same. This implies that $w_{pl}(\widehat{C})$ is close to zero. Which leads to a contradiction as we will show that $|w_{pl}(\widetilde{C})| > |w_{pl}(\widehat{C})|$ where $\widetilde{C}$ is one of the planar cycles in which $\widehat{C}$ can be decomposed. Hence P_1 and P_2 cannot be two minimum weight paths in K_i^{uv}.

Consider a graph $\widehat{G}$ that is similar to G except that in $\widehat{G}$ we split each common intermediate vertex u_i (both crossing and grazing vertices) of the paths P_1 and P_2, into two vertices u_i' and u_i'', such that u_i' and u_i'' are δ distance apart (see Figure 4). If $e = xu_i$ was an edge in P_1(or P_2) then we will have the edge $e' = xu_i'$ (or $e' = xu_i''$) in $\widehat{G}$. Let $N = cn^k$ be an upper bound on the coordinates of the embedding of G, where c and k are constants. Then by definition of w_{pl}, we have $|w_{\text{pl}}(e) - w_{\text{pl}}(e')| \leq 4N\delta + \delta^2$. Let us define $f(\delta) := 4N\delta + \delta^2$.

Let $v_1, v_2, \ldots, v_k$ be the crossing vertices of P_1 and P_2, and $Q^{1'}, Q^{2'}, \ldots, Q^{(k+1)'}$ be the paths from u to v_1', v_1' to v_2' and so on till v_k' to v respectively, such that the paths $Q^{1'}, Q^{2'}, \ldots, Q^{(k+1)'}$ correspond to the paths $P_1^1, P_2^2 \ldots, P_i^{k+1}$ respectively (see Figure 4). Note that the paths $Q^{i'}$ and P_j^i (where j is 1 if i is odd and 2 otherwise) can have at most n edges that differ and their weights w_{pl} differ by at most $f(\delta)$ for each such edge. Therefore $|w_{\text{pl}}(P_j^i) - w_{\text{pl}}(Q^{i'})| \leq n \cdot f(\delta)$. Similarly let $Q^{1''}, Q^{2''}, \ldots, Q^{(k+1)''}$ be the paths from u to v_1'', v_1'' to v_2'' and so on till v_k'' to v respectively, such that the paths $Q^{1''}, Q^{2''}, \ldots, Q^{(k+1)''}$ correspond to the paths $P_2^1, P_1^2 \ldots, P_j^{k+1}$ respectively. By an analogous argument we have, $|w_{\text{pl}}(P_j^i) - w_{\text{pl}}(Q^{i''})| \leq n \cdot f(\delta)$ (where j is 2 if i is odd and 1 otherwise).

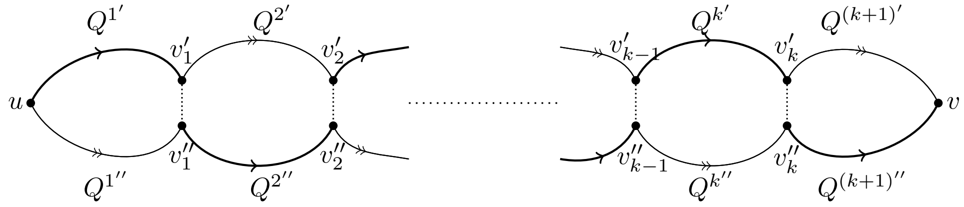


Figure 4 Splitting vertices to form the graph $\widehat{G}$ from G.

Now $Q' = Q^{1'}Q^{2'} \ldots Q^{(k+1)'}$ and $Q'' = Q^{1''}Q^{2''} \ldots Q^{(k+1)''}$ are paths from u to v (corresponding to the paths P' and P'' respectively) that do not cross each other, as shown in Figure 4. Observe that $\text{sign}(Q') = \text{sign}(P')$ since the difference between the coordinates of vertices along Q' and P' is less than 1, therefore the number of crossings with respect to each segment of the flat schema remains the same. Similarly, $\text{sign}(Q'') = \text{sign}(P'')$. Now consider the simple cycle $\widehat{C} = Q'(Q'')^r$. By Lemma 12, $\text{sign}(\widehat{C}) = 0$. Hence $\widehat{C}$ is a surface separating cycle by Theorem 3.

$$\begin{aligned}
|w_{\text{pl}}(\widehat{C}) - w_{\text{pl}}(C')| &= |(w_{\text{pl}}(Q') + w_{\text{pl}}((Q'')^r)) - (w_{\text{pl}}(P') + w_{\text{pl}}((P'')^r))| \\
&= |\Big(\Big(w_{\text{pl}}(Q^{1'}) + \ldots + w_{\text{pl}}(Q^{(k+1)'})\Big) - \Big(w_{\text{pl}}(Q^{1''}) + \ldots + w_{\text{pl}}(Q^{(k+1)''})\Big)\Big) \\
&\quad - \Big(\Big(w_{\text{pl}}(P_1^1) + \ldots + w_{\text{pl}}(P_i^{(k+1)})\Big) - \Big(w_{\text{pl}}(P_2^1) + \ldots + w_{\text{pl}}(P_j^{(k+1)})\Big)\Big)| \\
&\leq \Big|w_{\text{pl}}(Q^{1'}) - w_{\text{pl}}(P_1^1)\Big| + \ldots + \Big|w_{\text{pl}}(Q^{(k+1)'}) - w_{\text{pl}}(P_i^{(k+1)})\Big| \\
&\quad + \Big|w_{\text{pl}}(Q^{1''}) - w_{\text{pl}}(P_2^1)\Big| + \ldots + \Big|w_{\text{pl}}(Q^{(k+1)''}) - w_{\text{pl}}(P_j^{(k+1)})\Big| \\
&\leq 2(k+1)nf(\delta)
\end{aligned}$$

Now since by Lemma 12 $w_{\text{pl}}(C') = 0$, therefore we can choose δ small enough (say less than $1/100N^3$) so that we get $|w_{\text{pl}}(\widehat{C})| < 1/3$.

Without loss of generality assume that C' crosses some segment of the flat schema. If not then both P_1 and P_2 would not be crossing any segment of the polygon and hence with respect to w_{pl} both cannot be minimum weight paths [26]. Since C' crosses some segment, therefore, $\widehat{C}$ also must cross the same segment. Since $\widehat{C}$ is a surface separating cycle therefore by Lemma 8, $\widehat{C}$ can be decomposed into planar cycles such that the weight of $\widehat{C}$ is equal to sum of the weights of the planar cycles with respect to w_{pl}. Moreover, the weight of each planar cycle has the same sign and each planar cycle has a dummy edge (an edge that is incident on a segment of the flat schema). Let $\widetilde{C}$ be one such planar cycle, and consider a triangulation of $\widetilde{C}$ (by thinking of $\widetilde{C}$ as a polygon). There exists some triangle say $T = (a, x, y)$ in this triangulation that contains the dummy edge xy of $\widetilde{C}$ as one of its sides. Now $||x - y|| \geq 1$ since vertices in C' were integral and in $\widehat{C}$, x and y were not shifted. Moreover, a cannot be a vertex that is δ close to any segment of the flat schema. This is because for every vertex v that lies at the intersection of cycle C' and the side S of the flat schema, v was not split when forming the cycle $\widehat{C}$. Hence the distance of a from the line joining x and y is at least $1 - \delta$. Therefore the area of the triangle $\text{Area}(T) > 1/2 - 1/200n$. Now, $\text{Area}(T) \leq \text{Area}(\widetilde{C}) \leq |w_{\text{pl}}(\widetilde{C})| \leq |w_{\text{pl}}(\widehat{C})| \leq 1/3$, where the second inequality follows from Theorem 6. This contradicts that P_1 and P_2 are two minimum weight paths in G with respect to w_{comb}.

Therefore the class K_i^{uv} has at most one minimum weight path from u to v with respect to w_{comb}. This completes the proof of Theorem 9. ◀

For a fixed pair of vertices u, v, the number of classes K_i^{uv} is at most 2^{2g}. Since by Theorem 9 there is at most one minimum weight path from u to v in each class K_i^{uv}, therefore we have the following the result.

▶ **Theorem 13.** *Let $G = (V, E)$ be a genus g graph embedded on a flat schema having $4g$ segments. Then there exists at most 2^{2g} minimum weight paths between any pair of vertices in G, with respect to weight function w_{comb}.*

Now we are ready to prove Theorem 7 which says that there is a weight function with respect to which there is at most one minimum weight path between any pair of vertices in G.

Proof of Theorem 7. Let M_{uv} be the set of all minimum weight paths from u to v with respect to w_{comb} and let $M = \bigcup_{(u,v) \in V^2} M_{uv}$. By Theorem 13, $|M_{uv}|$ is at most 2^{2g}. Hence $|M| \leq n^2 \cdot 2^{2g}$.

Assume that edges of the graph are labeled $e_1, e_2, \ldots, e_m$. Define a weight function w_{index} such that it assigns weight 2^i to the edge e_i. It is easy to see that every path in graph gets unique weight with respect to w_{index}. Thus every path in M also gets a unique weight with respect to w_{index}. However w_{index} is an exponential valued weight function.

Now by Theorem 4 there exists an $O(\log n + g)$ bit prime p such that with respect to weight function $w_{\text{fks}} := (w_{\text{index}}) \bmod p$, every path in M gets a unique weight. Therefore with respect to the weight function $w := w_{\text{comb}} \cdot n^{k_2} + w_{\text{fks}}$, where k_2 is a sufficiently large constant, the minimum weight path between every pair of vertices in graph is unique. Note that w_{comb} and w_{fks} are $O(\log n)$ bit and $O(\log n + g)$ bit weight functions respectively.

Computing w_{comb} can be done in logspace since it is a simple function of the coordinates of the end points of an edge. To compute w_{fks} one needs to find the appropriate prime whose existence is shown in Theorem 4. For each prime, we check if G is min-unique with respect to the corresponding weight function and if not we move to the next prime. This can be done by a nondeterministic $O(\log n + g)$ space algorithm along a unique computation path as shown in [20]. ◀

Proof of Theorem 1. Now given a graph G on n vertices and two vertices s and t in G we cycle through all primes less than n', and for each prime, we compute the weight function w given in Theorem 7. Using Theorem 5 we check if G is min-unique with respect to w and if so we check if there is a path from s to t in G. If G is not min-unique with respect to w then we move to the next prime. Theorem 4 guarantees that there is an $n' = n^{O(1)}$ and a prime less than n' such that G is min-unique with respect to the corresponding prime. Hence along a unique computation path, we finally have *Yes* or *No* answer depending on whether s is reachable from t or not respectively, while all other paths halt and reject. ◀

References

1 Manindra Agrawal, Rohit Gurjar, Arpita Korwar, and Nitin Saxena. Hitting-Sets for ROABP and Sum of Set-Multilinear Circuits. *SIAM J. Comput.*, 44(3):669–697, 2015. doi:10.1137/140975103.

2 Eric Allender, David A. Mix Barrington, Tanmoy Chakraborty, Samir Datta, and Sambuddha Roy. Planar and Grid Graph Reachability Problems. *Theory of Computing Systems*, 45(4):675–723, 2009. doi:10.1007/s00224-009-9172-z.

3 Eric Allender, Klaus Reinhardt, and Shiyu Zhou. Isolation, Matching, and Counting: Uniform and Nonuniform Upper Bounds. *Journal of Computer and System Sciences*, 59:164–181, 1999.

4 Carme Àlvarez and Birgit Jenner. A Very Hard Log-space Counting Class. *Theoretical Computer Science*, 107:3–30, 1993.

5 Rahul Arora, Ashu Gupta, Rohit Gurjar, and Raghunath Tewari. Derandomizing Isolation Lemma for $K_{3,3}$-free and K_5-free Bipartite Graphs. In *33rd Symposium on Theoretical Aspects of Computer Science, STACS 2016, February 17-20, 2016, Orléans, France*, pages 10:1–10:15, 2016.

6 David A. Mix Barrington. Bounded-width polynomial-size branching programs recognize exactly those languages in NC^1. *Journal of Computer and System Sciences*, 38:150–164, 1989.

7 David A. Mix Barrington, Chi-Jen Lu, Peter Bro Miltersen, and Sven Skyum. Searching constant width mazes captures the AC^0 hierarchy. In *15th International Symposium on Theoretical Aspects of Computer Science (STACS)*, Volume 1373 in Lecture Notes in Computer Science, pages 74–83. Springer, 1998.

8 Chris Bourke, Raghunath Tewari, and N. V. Vinodchandran. Directed Planar Reachability Is in Unambiguous Log-Space. *ACM Transactions on Computation Theory*, 1(1):1–17, 2009. doi:10.1145/1490270.1490274.

9 Gerhard Buntrock, Birgit Jenner, Klaus-Jörn Lange, and Peter Rossmanith. Unambiguity and fewness for logarithmic space. In *Proceedings of the 8th International Conference on Fundamentals of Computation Theory (FCT'91)*, Volume 529 Lecture Notes in Computer Science, pages 168–179. Springer-Verlag, 1991.

10 Sergio Cabello and Bojan Mohar. Finding Shortest Non-Separating and Non-Contractible Cycles for Topologically Embedded Graphs. *Discrete & Computational Geometry*, 37(2):213–235, February 2007. doi:10.1007/s00454-006-1292-5.

11 Samir Datta, Raghav Kulkarni, and Sambuddha Roy. Deterministically Isolating a Perfect Matching in Bipartite Planar Graphs. In *STACS 2008, 25th Annual Symposium on Theoretical Aspects of Computer Science, Bordeaux, France, February 21-23, 2008, Proceedings*, pages 229–240, 2008. doi:10.4230/LIPIcs.STACS.2008.1346.

12 Samir Datta, Raghav Kulkarni, Raghunath Tewari, and N.V. Vinodchandran. Space complexity of perfect matching in bounded genus bipartite graphs. *Journal of Computer and System Sciences*, 78(3):765–779, 2012. In Commemoration of Amir Pnueli. doi:10.1016/j.jcss.2011.11.002.

13 Kousha Etessami. Counting quantifiers, successor relations, and logarithmic space. *Journal of Computer and System Sciences*, 54(3):400–411, June 1997.

14 Stephen A. Fenner, Rohit Gurjar, and Thomas Thierauf. Bipartite Perfect Matching is in quasi-NC. *CoRR*, 2016. arXiv:1601.06319.

15 Michael L. Fredman, János Komlós, and Endre Szemerédi. Storing a Sparse Table with 0(1) Worst Case Access Time. *J. ACM*, 31(3):538–544, June 1984. doi:10.1145/828.1884.

16 Anna Gál and Avi Wigderson. Boolean complexity classes vs. their arithmetic analogs. *Random Struct. Algorithms*, 9(1-2):99–111, 1996. doi:10.1002/(SICI)1098-2418(199608/09)9:1/2\%3C99::AID-RSA7\%3E3.0.CO;2-6.

17 Anna Galluccio and Martin Loebl. On the Theory of Pfaffian Orientations. I. Perfect Matchings and Permanents. *Electr. J. Comb.*, 6, 1999. URL: http://www.combinatorics.org/Volume_6/Abstracts/v6i1r6.html.

18 Rohit Gurjar, Arpita Korwar, and Nitin Saxena. Identity Testing for Constant-Width, and Commutative, Read-Once Oblivious ABPs. In *31st Conference on Computational Complexity, CCC 2016, May 29 to June 1, 2016, Tokyo, Japan*, pages 29:1–29:16, 2016. doi:10.4230/LIPIcs.CCC.2016.29.

19 Rohit Gurjar, Arpita Korwar, Nitin Saxena, and Thomas Thierauf. Deterministic Identity Testing for Sum of Read-once Oblivious Arithmetic Branching Programs. In *Proceedings of the 30th Conference on Computational Complexity*, CCC '15, pages 323–346, Germany, 2015. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.CCC.2015.323.

20 Vivek Anand T. Kallampally and Raghunath Tewari. Trading Determinism for Time in Space Bounded Computations. In *41st International Symposium on Mathematical Foundations of Computer Science, MFCS 2016, August 22-26, 2016 - Kraków, Poland*, pages 10:1–10:13, 2016. doi:10.4230/LIPIcs.MFCS.2016.10.

21 Jan Kynčl and Tomáš Vyskočil. Logspace Reduction of Directed Reachability for Bounded Genus Graphs to the Planar Case. *ACM Transactions on Computation Theory*, 1(3):1–11, 2010. doi:10.1145/1714450.1714451.

22 Meena Mahajan and Kasturi R. Varadarajan. A New NC-algorithm for Finding a Perfect Matching in Bipartite Planar and Small Genus Graphs (Extended Abstract). In *Proceedings of the Thirty-second Annual ACM Symposium on Theory of Computing*, STOC '00, pages 351–357, New York, NY, USA, 2000. ACM. doi:10.1145/335305.335346.

23 Aduri Pavan, Raghunath Tewari, and N. V. Vinodchandran. On the power of unambiguity in log-space. *Computational Complexity*, 21(4):643–670, 2012. doi:10.1007/s00037-012-0047-3.

24 Omer Reingold. Undirected connectivity in log-space. *Journal of the ACM*, 55(4), 2008. doi:10.1145/1391289.1391291.

25 Klaus Reinhardt and Eric Allender. Making Nondeterminism Unambiguous. *SIAM J. Comput.*, 29(4):1118–1131, 2000. doi:10.1137/S0097539798339041.

26 Raghunath Tewari and N. V. Vinodchandran. Green's theorem and isolation in planar graphs. *Inf. Comput.*, 215:1–7, 2012. doi:10.1016/j.ic.2012.03.002.

27 Thomas Thierauf and Fabian Wagner. Reachability in $K_{3,3}$-free Graphs and K_5-free Graphs is in Unambiguous Log-Space. In *17th International Conference on Foundations of Computation Theory (FCT)*, Lecture Notes in Computer Science 5699, pages 323–334. Springer-Verlag, 2009.

28 Carsten Thomassen. The Graph Genus Problem is NP-Complete. *J. Algorithms*, 10(4):568–576, 1989. doi:10.1016/0196-6774(89)90006-0.

29 Dieter van Melkebeek and Gautam Prakriya. Derandomizing Isolation in Space-Bounded Settings. In *32nd Computational Complexity Conference, CCC 2017, July 6-9, 2017, Riga, Latvia*, pages 5:1–5:32, 2017. doi:10.4230/LIPIcs.CCC.2017.5.

30 Avi Wigderson. $NL/poly \subseteq \oplus L/poly$ (Preliminary Version). In *Proceedings of the Ninth Annual Structure in Complexity Theory Conference, Amsterdam, The Netherlands, June 28 - July 1, 1994*, pages 59–62, 1994. doi:10.1109/SCT.1994.315817.