ANNALES DE LA FACULTÉ DES SCIENCES DE TOULOUSE
Mathématiques

Martin Dyer and Haiko Müller

Graph classes and the switch Markov chain for matchings

Tome XXIV, nº 4 (2015), p. 885-933.

<http://afst.cedram.org/item?id=AFST_2015_6_24_4_885_0>

Annales de la Faculté des Sciences de Toulouse Vol. XXIV, n° 4, 2015
pp. 885-933

Graph classes and the switch Markov chain for matchings

Martin Dyer[(1)], Haiko Müller[(1)]

Résumé. — Diaconis, Graham et Holmes [8] ont étudié les applications statistiques du comptage et de l'échantillonnage des appariements parfaits dans certaines classes de graphes. Ils ont proposé une chaîne de Markov simple, appelée ici la chaîne "switch", pour engendrer aléatoirement un appariement presque uniforme pour les graphes appartenant à ces classes. Nous étudions ces classes en détail en les justifiant du point de vue de la théorie des graphes. Nous montrons que l'ergodicité des chaînes des classes de [8] se déduit de celle d'une classe plus large. Nous nous intéressons également à la complexité calculatoire du temps de mélange de la chaîne switch et nous la déterminons pour toutes les classes de [8] sauf une, celle correspondant aux graphes monotones de Diaconis, Graham et Holmes. Nous ébauchons une approche pour montrer une convergence en temps polynomial de la chaîne switch pour les graphes monotones. Elle dépend d'une conjecture intéressante mais non-prouvée concernant les cycles hamiltoniens des graphes monotones.

Abstract. — Diaconis, Graham and Holmes [8] studied the statistical applications of counting and sampling perfect matchings in certain classes of graphs. They proposed a simple Markov chain, called the switch chain here, to generate a matching almost uniformly at random for graphs in these classes. We examine these graph classes in detail, and show that they have a strong graph-theoretic rationale. We consider the ergodicity of the switch chain, and show that all the classes in [8] inherit their ergodicity from a larger class. We also study the computational complexity of the mixing time of the switch chain, and show that this has already been resolved for all but one of the classes in [8], that which Diaconis, Graham and Holmes called monotone graphs. We outline an approach to showing polynomial time convergence of the switch chain for monotone graphs. This is shown to rely upon an interesting, though unproven, conjecture concerning Hamilton cycles in monotone graphs.

1. School of Computing, University of Leeds, Leeds, LS2 9JT, UK.

The work described here was supported in part by the EPSRC grant "Computational Counting" (ref. EP/I012087/1).

1. Introduction

The computational problems of (approximately or exactly) evaluating the *permanent*, and sampling perfect matchings (almost) uniformly at random from a graph, are well known in Computer Science, and elsewhere. In [8], Diaconis, Graham and Holmes discussed the applications of the permanent to Statistics. They studied permanents of 0-1 matrices arising naturally in these applications, which they called *truncated* or *interval-restricted*.

These matrices can be viewed as the biadjacency matrices of bipartite graphs. Then the truncated matrices are those which have the property that their columns can be permuted to give the *consecutive ones* property on rows. That is, no two ones in any row are separated by one or more zeros. Diaconis, Graham and Holmes [8] considered two types of truncation : "one-sided", where the consecutive ones appear at the left of each row, and "two-sided", where the consecutive ones can appear at any position in each row. Within the two-sided case, they considered two subcases. The first is where the rows and columns can be permuted so that both rows and columns have the consecutive ones property. The second is a subclass of this, where the consecutive ones have a "staircase" presentation, which we will describe later. In this case, they called the underlying graph *monotone*.

Diaconis, Graham and Holmes proposed a simple Markov chain for sampling perfect matchings in a graph, which we will call the *switch* chain, and they conjectured that it would mix rapidly for these truncated matrices. The convergence of the switch chain for these cases was subsequently studied in the PhD theses of Matthews [27] and Blumberg [3].

In this paper, we show that the matrices considered by Diaconis, Graham and Holmes [8] correspond to an ascending sequence of natural graph classes, in which the switch chain is ergodic, and we identify the largest class in this sequence. We examine the mixing time behaviour of the switch chain for graphs from these classes, extending the work of [8], [3] and [27].

For the necessary background information on Markov chains, see [1, 18, 24]. For the relevant graph-theoretic background, see [5, 14, 33, 37].

1.1. Notation and definitions

Let $\mathbb{N} = \{1, 2, \ldots\}$ denote the natural numbers, and $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$. If $n \in \mathbb{N}$, let $[n] = \{1, 2, \ldots, n\}$ and, if $n_1, n_2 \in \mathbb{N}_0$, let $[n_1, n_2] = \{n_1, n_1 + 1, \ldots, n_2\}$.

We will use the notation $[n]' = \{1', 2', \ldots, n'\}$ and $[n_1, n_2]' = \{n_1', (n_1+1)', \ldots, n_2'\}$. Here the prime serves only to distinguish i from i'. Ordering and arithmetic for $[n]'$ elements follows that for $[n]$. Thus, for example, $1' < 2'$ and $1' + 2' = 3'$.

A graph $G = (V, E)$ is *bipartite* if its vertex set $V = [m] \cup [n]'$ and there is no (undirected) edge $(v, w) \in E$ such that $v, w \in [m]$ or $v, w \in [n]'$. Thus V comprises two independent sets $[m]$ and $[n]'$. Bipartite graphs $G_1 = ([m] \cup [n]', E_1)$ and $G_2 = ([m] \cup [n]', E_2)$ are *isomorphic* if there are permutations σ of $[m]$ and τ of $[n]'$ such that $(j, k') \in E_1$ if and only if $(\sigma_j, \tau_{k'}) \in E_2$.

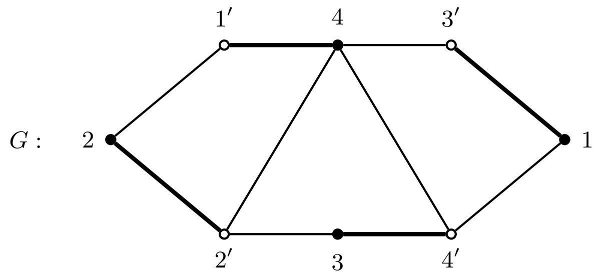
Let $G = ([m] \cup [n]', E)$ be a bipartite graph. We consider $[m]$ and $[n]'$ to have the usual linear ordering, and we will abuse notation by denoting these ordered sets simply by $[m]$ and $[n]'$. Then let $A(G)$ denote the $m \times n$ *biadjacency matrix* of G, with rows indexed by $[m]$ and columns by $[n]'$, such that $A(i, j') = 1$ if $(i, j') \in E$, and $A(i, j') = 0$ otherwise. We will use the graph and matrix terminology interchangeably. For example, we refer to rows and columns of G, or edges in $A(G)$.

The neighbourhood in G of a vertex $v \in [m] \cup [n]'$ will be denoted by $\mathcal{N}(v)$. To avoid trivialities, we will assume that G has no isolated vertices, unless explicitly stated otherwise.

A *matching* in a bipartite graph $G = ([m] \cup [n]')$ is a set of independent edges, that is, no two edges in the set share an endpoint. A *perfect* matching is a set of edges such that every vertex of G lies in exactly one edge. For a bipartite graph $G = ([m] \cup [n]', E)$ this requires $m = n$, and n independent edges in E. In particular, G can have no isolated vertices. We will call a bipartite graph with $m = n$ *balanced*, though we may omit this restriction when it is clear from the context.

Equivalently, a perfect matching may be viewed as n independent 1's in the $n \times n$ 0-1 matrix $A(G)$. Thus a perfect matching M has edge set $\{(i, \pi_i') : i \in [n]\}$, where π is a permutation of $[n]$. Equivalently, M has edge set $\{(\sigma_j, j') : j \in [n]\}$, where σ is a permutation of $[n]$. Note that $\sigma = \pi^{-1}$ as elements of the symmetric group S_n. We may identify the matching M with the permutations π and σ. An example is shown in Fig. 1 below.

The total number of perfect matchings in a bipartite graph G is the *permanent*, which we denote by $\mathrm{per}(A)$ when $A = A(G)$.



$$A(G) : \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{cccc} 1' & 2' & 3' & 4' \\ 0 & 0 & \boxed{1} & 1 \\ 1 & \boxed{1} & 0 & 0 \\ 0 & 1 & 0 & \boxed{1} \\ \boxed{1} & 1 & 1 & 1 \end{array}$$

Figure 1. — Bipartite graph with perfect matching $\sigma = (3241)$, $\pi = (4213)$.

We will also require the following graph-theoretic definition. The *pathwidth* $\mathrm{pw}(G)$ of a graph G was introduced by Robertson and Seymour [29]. A pair (X, P) is a *path decomposition* of a graph $G = (V, E)$ if $P = ([l], F)$ is a path, with $F = \{(i, i+1) : i \in [l-1]\}$, and X maps *nodes* (vertices) of P to subsets of V, called *bags*, such that

(a) for each vertex $v \in V$ there is a node $i \in [l]$ such that v is in the bag $X(i)$,
(b) for each edge $(u, v) \in E$ there is a node $i \in [l]$ such that u and v are in the bag $X(i)$,
(c) For each vertex $v \in V$ the set of nodes $\{i \in [l] : v \in X(i)\}$ is connected in P.

Note that, subject to (a) and (b), (c) is equivalent to : for all i, j, k with $i < j < k$ we have $X(i) \cap X(k) \subseteq X(j)$.

The *width* of the path decomposition (X, P) is $\max\{|X(i)| - 1 : i \in [l]\}$ and the *pathwidth* of G, denoted by $\mathrm{pw}(G)$, is the minimum width of any path decomposition of G.

An obvious, but useful, property of pathwidth is that if G^* is any subgraph of G, then $\mathrm{pw}(G^*) \leq \mathrm{pw}(G)$.

There is an alternative view of this quantity, which is useful. Any linear order of the vertices of a graph G is called a *layout*. Suppose we visit the vertices in the order of the layout. Then the maximum number of already-visited vertices which have an unvisited neighbour is called the *vertex separation* of the layout. The minimum value of the vertex separation over all layouts is called the *vertex separation number* of G, $\mathrm{vs}(G)$. Formally, let $G = ([n], E)$ and let S_n be the symmetric group on $[n]$. Then

$$\mathrm{vs}(G) \;=\; \min_{\sigma \in S_n} \max_{j \in [n]} \left| \{ i \leq j : \exists k > j \text{ with } (\sigma_i, \sigma_k) \in E \} \right| .$$

Vertex separation was studied by Ellis, Sudborough and Turner [10] who showed, in particular, that $\mathrm{vs}(T) = O(\log n)$ for any n-vertex tree T. They gave an $O(n)$ time algorithm for determining $\mathrm{vs}(T)$, and an $O(n \log n)$ algorithm for computing the optimal layout. Skodinis [30] improved the latter to $O(n)$.

Kinnersley [22] showed that $\mathrm{vs}(G) = \mathrm{pw}(G)$ for any graph G. Hence we will use $\mathrm{pw}(G)$ rather than $\mathrm{vs}(G)$ for this quantity. For graphs with a constant degree bound, Makedon and Sudborough [26] showed that pathwidth is related by a constant factor to other measures of graph width, such as *cutwidth* and *bandwidth*.

1.2. Computing the permanent

The permanent has been studied extensively in Combinatorics and Computer Science. Valiant showed that computing the permanent *exactly* is #P-complete for a general 0-1 matrix [36]. No algorithm running in sub-exponential time is known for the exact evaluation of the permanent of 0-1 matrices.

Jerrum, Sinclair and Vigoda [20] showed that the permanent has a *fully polynomial randomised approximation scheme* (FPRAS), using an algorithm for randomly sampling perfect matchings. This improved a Markov chain algorithm of Jerrum and Sinclair [19]. The algorithm of [20] is simple, but involves polynomially many repetitions of a polynomial-length sequence of related Markov chains. The best bound known for the running time of this algorithm is $O(n^7 \log^4 n)$, due to Bezáková, Štefankovič, Vazirani and Vigoda [2].

Jerrum, Valiant and Vazirani [21] showed that sampling almost uniformly at random and approximate counting have equivalent computational complexity for many combinatorial problems, including the permanent.

From the viewpoint of theoretical Computer Science, these results entirely settle the question of sampling and counting perfect matchings in bipartite graphs. However, simpler methods have been proposed for special cases of this problem, and here we consider one such proposal.

1.3. The switch chain

Diaconis, Graham and Holmes [8] proposed the following Markov chain for sampling perfect matchings from a balanced bipartite graph $G = ([n] \cup [n]', E)$ almost uniformly at random, which we will call the *switch chain*. A transition of the chain will be called a *switch*. Diaconis, Graham and Holmes [8] called this a "transposition". The switch chain generalises the transposition chain for generating random permutations.

Switch chain

Let the perfect matching M_t at time t be the permutation π of $[n]$.

(1) Set $t \leftarrow 0$, and let M_0 be any perfect matching of G.

(2) Choose $i, j \in [n]$, uniformly at random, so (i, π'_i), $(j, \pi'_j) \in M_t$.

(3) If $i \neq j$ and $(i, \pi'_j), (j, \pi'_i)$ are both in E,
set $M_{t+1} \leftarrow M_t \setminus \{(i, \pi'_i), (j, \pi'_j)\} \cup \{(i, \pi'_j), (j, \pi'_i)\}$.

(4) Otherwise, set $M_{t+1} \leftarrow M_t$.

(5) Set $t \leftarrow t + 1$. If $t < t_{\max}$, repeat from step (2). Otherwise, stop.

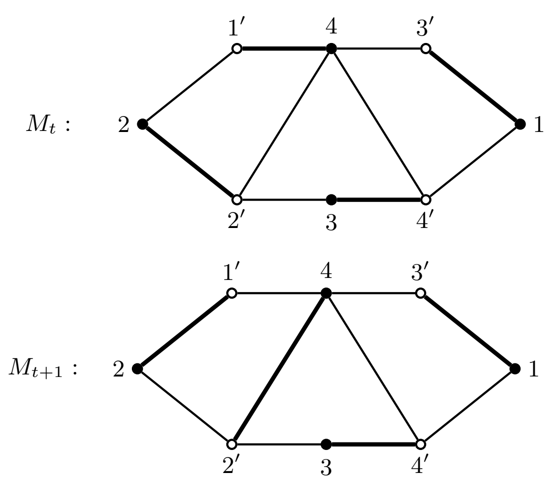


Figure 2

Note that the switch chain is invariant under isomorphisms of G, so properties of the chain can be investigated from the viewpoint of graph theory. An example of a switch is shown below, with the edges $(4, 1')$, $(2, 2')$ being switched for $(4, 2')$, $(2, 1')$

2. Graph classes

Here we consider graph classes which are equivalent to the matrix classes considered by Diaconis, Graham and Holmes. We examine other related classes in sections 2.6 and 2.7. Many of these classes lead to certain orderings of the rows and/or columns of the biadjacency matrix, which exhibit particular properties. These orderings can always be found by a fast algorithm, in most cases with $O(n)$ time complexity. Unless stated otherwise, we will assume that the biadjacency matrix is presented with this ordering. For example, in section 2.1, we consider Γ-free orderings, so we would assume that the biadjacency matrix is presented with a Γ-free ordering.

2.1. Chordal bipartite graphs

The first question we might ask about the switch chain is : for which class of graphs is it ergodic ? We wish to have a graph-theoretic answer to this question, so that we can recognise membership of graphs in the class. Therefore, we restrict attention to *hereditary* graph classes, that is, those for which all (vertex) induced subgraphs of every graph in the class are in the class.

There is a further technical reason for preferring hereditary graph classes. We then have *self-reducibility* for most problems in #P, including the permanent. This property implies the equivalence between sampling and counting referred to in Section 1.2. See [21].

To answer the ergodicity question, we will use the term “hereditary” in a slightly weaker sense, where appropriate. Since the switch chain is defined only for balanced bipartite graphs, we are not interested in induced subgraphs which are not balanced. Therefore we will say that a class of balanced bipartite graphs is hereditary if every balanced induced subgraph of a graph in the class is also in the class.

Diaconis, Graham and Holmes [8] observed that the switch chain is not ergodic for all balanced bipartite graphs. They gave the example :

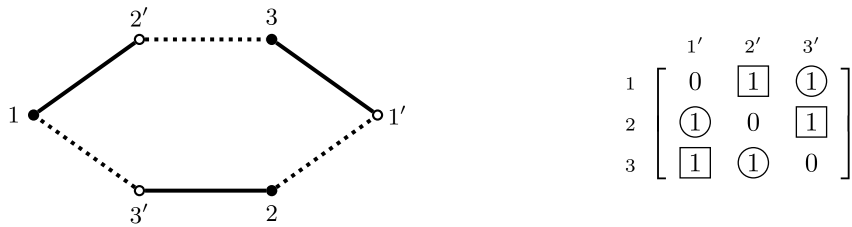


Figure 3

This graph has two perfect matchings, but the switch chain cannot move between them. This is because the graph is a chordless 6-cycle. In fact, this property characterises the class of graphs for which the switch chain is not ergodic, as we now show.

Definition 2.1. — *A graph G is* chordal bipartite *if and only if it has no chordless cycle of length other than four.*

The class of chordal bipartite graphs is clearly hereditary. Note that the definition of chordal bipartite graphs is an "excluded subgraph" characterisation. To show that the switch chain is ergodic for this class, we require the following "excluded submatrix" characterisation.

A Γ (Gamma) in a 0-1 matrix is an induced submatrix of the form

$$\Gamma \; : \quad \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}.$$

A matrix is called Γ-free if it has no such induced submatrix. Then Lubiw [25] gave the following characterisation.

Theorem 2.2 (Lubiw). — *A bipartite graph is chordal bipartite if and only if it is isomorphic to a graph G such that $A(G)$ is Γ-free.*

Moreover, Lubiw [25] showed that this characterisation can be used to recognise chordal bipartite graphs in $O(|E| \log |E|)$ time. This was subsequently improved to $O(|E|)$ time by Uehara [35]. For the switch chain we then have the following characterisation of the bipartite graphs on which it is ergodic :

Lemma 2.3. — *The largest hereditary class of bipartite graphs for which the switch chain is ergodic is the class of balanced chordal bipartite graphs. In this class, if $G = ([n] \cup [n]', E)$, the diameter of the chain is at most n.*

Proof.— Clearly any graph with an induced cycle of length greater than 4 cannot be in the class, so we need only show ergodicity for chordal bipartite graphs. The chain is aperiodic, since there is a loop probability at least $1/n$ at each step, from choosing $i = j$ in step (2). Thus we must show that the chain is irreducible. From Theorem 2.2, we may suppose that $A(G)$ is given with a Γ-free presentation.

Let $\mathcal{G} = (\Omega, \mathcal{E})$ be the transition graph of the switch chain, with Ω the set of perfect matchings in G, and $\mathcal{E}$ the set of transitions. We must show that $\mathcal{G}$ is connected, and has diameter n. Let π and σ be any two perfect matchings in G, and let $\operatorname{dist}(\pi, \sigma) = |\{i : \pi'_i \neq \sigma'_i\}|$. Note that $\operatorname{dist}(\pi, \sigma) \leq n$, and $\operatorname{dist}(\pi, \sigma) = 0$ implies $\pi = \sigma$.

Let k be the smallest index such that $\pi'_k \neq \sigma'_k$ and, without loss of generality, suppose $\pi'_k > \sigma'_k$. Then there exists $\ell > k$ such that $\pi'_\ell = \sigma'_k$, and hence $\pi'_\ell \neq \sigma'_\ell$. Then we have (k, σ'_k), (k, π'_k), $(\ell, \sigma'_k) \in E$, $\ell > k$ and $\pi'_k > \sigma'_k$.

$$\begin{array}{c} \\ k \\ \ell \end{array} \begin{array}{cc} \scriptstyle \pi'_\ell = \sigma'_k & \scriptstyle \pi'_k \\ \left[\begin{array}{c} 1 \\ 1 \end{array} \right. & \left. \begin{array}{c} 1 \\ ? \end{array} \right] \end{array}$$

The Γ-free property of $A(G)$ then implies $(\ell, \pi'_k) \in E$. Thus we have (k, π'_k), $(\ell, \pi'_\ell) \in \pi$ and (k, π'_ℓ), $(\ell, \pi'_k) \in E$. Therefore $\tau \in \Omega$ and $(\pi, \tau) \in \mathcal{E}$, where

$$\tau = \pi \setminus \{(k, \pi'_k), (\ell, \pi'_\ell)\} \cup \{(k, \pi'_\ell), (\ell, \pi'_k)\}.$$

Note that $\tau'_i = \pi'_i$ for $i \neq k, \ell$. However, $\pi'_k \neq \sigma'_k$, but $\tau'_k = \pi'_\ell = \sigma'_k$. Also $\pi'_\ell \neq \sigma'_\ell$, but $\tau'_\ell = \pi'_k = \sigma'_\ell$ if $\pi'_k = \sigma'_\ell$. Thus $\operatorname{dist}(\pi, \sigma) - 2 \leq \operatorname{dist}(\tau, \sigma) \leq \operatorname{dist}(\pi, \sigma) - 1$. Hence there is a path of at most n edges in $\mathcal{G}$ between any pair of matchings π, σ. Thus the diameter of $\mathcal{G}$ is at most n. □

Computing the permanent exactly is known to be #P-complete for the class of chordal bipartite graphs [28], though this result does not even cover the case of chordal bipartite graphs of bounded degree. The complexity of exact computation of the permanent is unknown for all the subclasses of chordal bipartite graphs considered below. That is, with the exception of chain graphs, which we examine in Section 2.5.

2.2. Convex graphs

The largest class of graphs considered in Diaconis, Graham and Holmes [8] were those with "two-sided restrictions". These are bipartite graphs G for which $A(G)$ has the *consecutive ones* property. These have been called *convex* graphs in the graph theory literature. They were introduced by Glover [13], who gave a simple greedy algorithm for finding a maximum matching in such a graph. The consecutive ones property can be recognised in $O(|E|)$ time by the well-known algorithm of Booth and Lueker [4].

Definition 2.4. — *A bipartite graph is* convex *if it is isomorphic to a graph* $G = ([m] \cup [n]', E)$ *such that* $\mathcal{N}(i)$ *is an interval* $[\alpha'_i, \beta'_i] \subseteq [n]'$ *for all* $i \in [m]$.

Note that this property remains true under an arbitrary permutation of $[m]$. Then

Lemma 2.5. — *Convex graphs are a proper hereditary subclass of chordal bipartite graphs.*

Proof.— It is easy to see that the class Convex is hereditary. To see that it is a subclass of chordal bipartite graphs, we permute rows so that $\beta'_i \le \beta'_j$ when $i < j$. Now we can show that $A(G)$ is Γ-free. If not, there is a Γ in some rows i, j and columns k', ℓ'.

$$\begin{array}{c} \\ i \\ j \end{array} \begin{array}{c} \begin{array}{cc} k' & \ell' \end{array} \\ \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \end{array}$$

We have $i < j$ but, since the rows of $A(G)$ have consecutive ones, $\beta'_i \ge \ell' > \beta'_j$. This contradicts our ordering of the rows of $A(G)$. To see that it is a proper subclass, note that there are at most $n!\binom{n}{2}^n = 2^{O(n \log n)}$ labelled convex graphs with n rows and columns, whereas Spinrad [31] has shown that there are $2^{\Theta(n \log^2 n)}$ chordal bipartite graphs. (Spinrad also gives in [31, Ex. 9.16(c)] an explicit example of a graph that is chordal bipartite but not convex.) □

It is possible to give excluded subgraph and excluded submatrix characterisations of convex graphs, but we will not explore this here, since they are not easy to describe, and appear to have little algorithmic application. See [34] for details.

Blumberg [3] gave an $r^{O(r)} n^4$ bound on the mixing time of the switch chain for convex graphs with $r = \max_{i \in [n]} \deg(i)$. This is a hereditary subclass of convex graphs, since it is easy to see that graphs with bounded row- or column-degree form a hereditary subclass of any hereditary class. We will give an algorithm for exact counting and sampling in this subclass of convex graphs. First we will show that these graphs have bounded column degree.

Lemma 2.6. — *Let $G = ([n] \cup [n]', E)$ be a convex graph containing a perfect matching. Let $r = \max_{i \in [n]} \deg(i)$ and $c = \max_{j \in [n]} \deg(j')$. Then we have $c \leq 2r - 1$.*

Proof.— Let M be any perfect matching of G. We first permute the rows of $A(G)$ so that M is the diagonal of A, i.e. $M \leftarrow \{(i, i') : i \in [n]\}$. To bound c, consider any edge $(i, j') \in E$. Since G is convex, and $(i, i') \in E$, we have $i', j' \in [\alpha'_i, \beta'_i]$ and so $|i - j| \leq r - 1$. Hence $i \in [j - r + 1, j + r - 1] \cap [n]$, and so $\mathcal{N}(j') \subseteq [j - r + 1, j + r - 1] \cap [n]$. Therefore we have $c \leq 2r - 1$. □

It is known that there is an exact algorithm for computing the permanent which is linear in n for all graphs of bounded *treewidth* [7, Theorem 1]. Convex graphs with $r = \max_{i \in [n]} \deg(i)$ have treewidth at most $2r - 1$. Unfortunately, the general algorithm of Courcelle, Makowsky and Rotics [7] is superexponential in the treewidth. An algorithm of Fürer [12, Theorem 3], for counting independent sets in graphs of bounded treewidth, could also be applied, since the treewidth of the line graph of a convex graph can be bounded by $8r^2$. (We will not prove these facts about treewidth here, since we do not use them, but see [17], for example.) Combined with Fürer's algorithm, this produces an algorithm for the permanent which is linear in n, but exponential in r^2.

However, we will not use either of these approaches, since the following dynamic programming algorithm has better time complexity for the problem at hand.

Lemma 2.7. — *Let $G = ([n] \cup [n]', E)$ be a convex graph containing a perfect matching, and let $r = \max_{i \in [n]} \deg(i)$. Then, for any subgraph G^* of G, the permanent of $A(G^*)$ can be evaluated exactly in time $O(r^{2r} n)$. Hence the permanent can be evaluated in polynomial time for all convex graphs with degree bound $O(\log n / \log\log n)$.*

Proof.— Let $A = A(G^*)$. The algorithm uses triangular windows W_i of width $2r + 1$ and height $2r + 1$, with corners at $A(i, (i - r)')$, $A(i, (i + r)')$ and $A(i + 2r, (i + r)')$. Note, from Lemma 2.6, that W_i cuts G as shown below. Moreover, for every edge of G there is an index i such that the corresponding entry of A appears in the window W_i.

At iteration i of the algorithm, a *subperfect* matching Q will be a matching of G^*, such that

(a) Every row $j \leq i$ has a matching edge ;
(b) Every column $j' \leq \min\{(i + r)', n'\}$ has a matching edge ;
(c) No row $j > i + 2r$ has a matching edge ;

(d) No column $j' > (i+r)'$ has a matching edge.

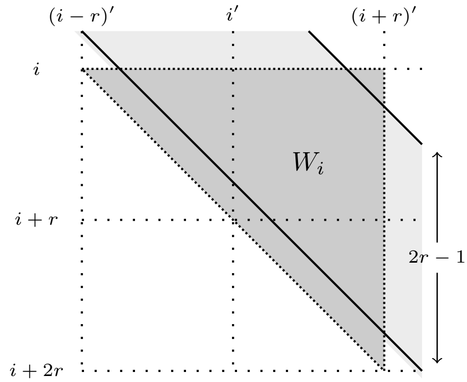


Figure 4. — The sliding window

Note that a any truncation of a perfect matching is subperfect, but a subperfect matching cannot always be extended to a perfect matching of G^*. We consider the set

$$S_i \;=\; \{M : M = Q \cap W_i \text{ and } Q \text{ is a subperfect matching}\}\,.$$

Note that $|S_i| < (2r)!$, since each column of W_i is either empty or contains a unique edge in any of positions $1, 2, \ldots, j$, for $j = 1, 2, \ldots, 2r-1$. For $M \in S_i$, let

$$N_i(M) = \left|\{Q : Q \cap W_i = M\}\right|,$$

be the number of subperfect matchings represented by M. Initially, $i = 1$ and S_1 will be the set of all matchings in W_1 such that every vertex $j' \leq (r+1)'$ has a matching edge. When $i = n-r$, all the subperfect matchings represented in W_{n-r} will be perfect matchings, and so we will have

$$\mathrm{per}(A) \;=\; \textstyle\sum_{M \in S_{n-r}} N_{n-r}(M)\,.$$

We must show how to update the M and $N_i(M)$ from W_i to W_{i+1}. Let $W_i^* = W_i \cap W_{i+1}$.

First we remove row i. We remove all $M \in S_i$ such that row i contains no matching edge, since they cannot correspond to a subperfect matching at iteration $(i+1)$. Then we delete the matching edge in row i from M, for all $M \in S_i$. This will produce a set S_i^* of matchings in W_i^*,

$$S_i^* \;=\; \{M : M = Q \cap W_i^* \text{ and } Q \text{ is a subperfect matching}\}\,.$$

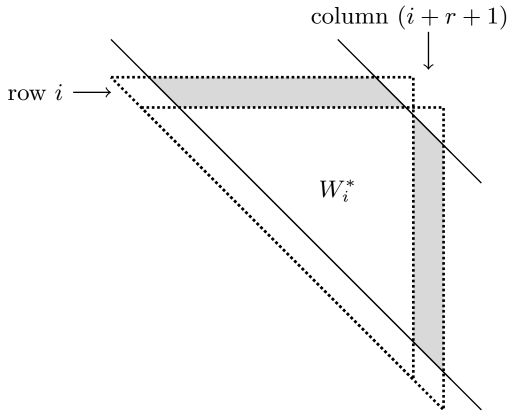


Figure 5. — Moving the window

We must now add column $(i+r+1)'$ to W_{i+1}. For all $M^* \in S_i^*$, we attempt to augment each M^* with a matching edge e in column $(i+r+1)'$. Note that e must be in W_{i+1}, and e can be in any row which has no matching edge in M^*. If no such row exists, we delete M^* from S_i^*, since it cannot correspond to a subperfect matching at iteration $i+1$. Otherwise, for each possible choice of e, we add $M = M^* \cup \{e\}$ to S_{i+1}, and set

$$N_{i+1}(M) \;=\; \sum \{N_i(M^*) : M^* \in S_i,\; M^* \cap W_{i+1} = M \cap W_i\}$$

This completes the description of the algorithm.

The operations in the update require $O(r|S_i|)$ time, except for the removal of duplicates, which can be implemented in $O(|S_i| \log |S_i|) = O(r^2|S_i|)$ time. Therefore, since

$$r^2|S_i| \;\leq\; r^2(2r)! \;\sim\; 2\sqrt{\pi}\, r^{5/2}(2r/e)^{2r} \;=\; O(r^{2r}),$$

using Stirling's formula, and $O(n)$ updates must be performed, the overall time complexity of the algorithm is $O(r^{2r}n)$. This is polynomial in n if $r = O(\log n / \log\log n)$. □

We can extend the algorithm of Lemma 2.7 to sample a matching uniformly at random. To do this, we must retain the sets S_i and the counts $N_i(M)$ ($M \in S_i$) used in the permanent evaluation. Then the sampling algorithm is a standard dynamic programming traceback through $S_{n-r}, \dots, S_i, \dots, S_1$, using the $N_i(M)$ to select matchings with the correct probability. See [9] for a more complete description of similar uses of traceback sampling. The time complexity for sampling a single matching is $O\big(\sum_i |S_i|\big) = O\big((2r)!n\big)$.

Thus dynamic programming seems superior to any known method of Markov chain sampling for convex graphs with small degree bound, at least if the chain is to be run for its a guaranteed mixing time.

2.3. Biconvex graphs

Diaconis, Graham and Holmes [8] considered the following subclass of convex graphs.

DEFINITION 2.8. — *A graph* $G = ([m] \cup [n]', E)$ *is* biconvex *if it is convex and* $\mathcal{N}(j')$ *is an interval* $[\alpha_{j'}, \beta_{j'}] \subseteq [n]$ *for all* $j' \in [n]'$.

Thus $A(G)$ has the consecutive ones property for both rows and columns.

LEMMA 2.9. — *Biconvex graphs are a proper hereditary subclass of convex graphs.*

Proof.— It is easy to see that the class BICONVEX is a hereditary subclass of CONVEX. To see that it is a proper subclass, consider the example :

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1' & 2' & 3' & 4' \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \end{array}.$$

In a biconvex ordering, row 2 must be adjacent to row 1 and row 3, or columns $1'$ and $2'$ cannot be convex. But row 4 must also be adjacent to row 2, or column $3'$ cannot be convex. These conditions clearly cannot be satisfied simultaneously. □

As with convex graphs, it is possible to give excluded subgraph and excluded submatrix characterisations of biconvex graphs. Since these are a little easier to describe than for convex graphs, we will give an excluded subgraph characterisation. Tucker shows [34, Theorem 10] that a bipartite graph is biconvex if and only if it does not contain the graphs I_n for $n \geq 1$, II_1, II_2, III_1, III_2 and III_3 as induced subgraph. Here I_n is a chordless cycle C_{2n+4}, II_1 is the triomino and III_1 is the tripod.

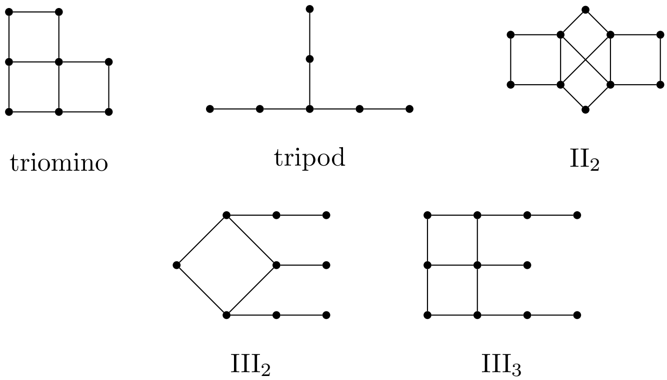


Figure 6. — The triomino, the tripod and the graphs II_2, III_2 and III_3

We know that the switch chain converges eventually on biconvex graphs, but how quickly is this guaranteed to occur ? Unfortunately, the convergence may be exponentially slow. Both Matthews [27] and Blumberg [3] gave the following examples $\mathcal{G}_k = \big([n] \cup [n]', \mathcal{E}_k\big)$, where $n = 2k - 1$:

$$(i, j') \in \mathcal{E}_k \iff \begin{cases} 1 \le i < k \text{ and } k' \le j' \le (k+i)'; \\ i = k \text{ and } 1' \le j' \le n'; \\ k < i \le n \text{ and } (i-k)' \le j' \le k'. \end{cases}$$

For example, $\mathcal{G}_4$ is

$$A(\mathcal{G}_4) \;=\; \begin{array}{c} \\ 1\\2\\3\\4\\5\\6\\7 \end{array} \begin{array}{c} \begin{array}{ccccccc} 1' & 2' & 3' & 4' & 5' & 6' & 7' \end{array} \\ \left[\begin{array}{ccccccc} 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{array}\right] \end{array}$$

Let π be any perfect matching. Then choosing $\pi'_k \le k'$ forces $\pi'_i = (k+i)'$ for $i \in [k-1]$, and similarly choosing $\pi'_k \ge k'$ forces $\pi'_{k+i} = i'$ for $i \in [k-1]$. Thus the set of perfect matchings of $\mathcal{G}_k$ is $S_1 \cup S_2$, where $S_1 = \{\pi : \pi'_k \le k'\}$ and $S_2 = \{\pi : \pi'_k \ge k'\}$.

Clearly $S_1 \cap S_2 = \{\pi : \pi'_k = k'\} = \{\sigma\}$, for a single matching σ. Moreover, it is not difficult to show that there are 2^{k-1} ways to extend a partial matching π with $\pi'_i = (k+i)'$ for $i \in [k-1]$ to a perfect matching. One way is to note that the submatrix induced by rows $[k, n]$ and columns $[k]'$ is a so-called chain graph, for which the permanent is easy to compute :

see Section 2.5 and the formula presented there. Thus $|S_1 \cap S_2| = 1$ and $|S_1| = |S_2| = 2^{k-1}$, and hence $|S_1 \cup S_2| = 2^k - 1$.

Therefore, if the switch chain is started at a random matching in S_1, it will need $\Omega(2^n)$ time before it reaches σ, and it cannot enter S_2 before this occurs. This gives an $\Omega(2^n)$ lower bound on the mixing time of the chain. This argument can be made completely rigorous, see [3] or [27], but we will not do so here.

2.4. Monotone graphs

Diaconis, Graham and Holmes [8] considered a subclass of biconvex graphs, which they called *monotone*, and showed that the switch chain is ergodic on monotone graphs. However, note that Lemma 2.3 gives a stronger result for the larger class of chordal bipartite graphs. Diaconis, Graham and Holmes [8] conjectured further that the switch chain mixes rapidly for the class MONOTONE.

DEFINITION 2.10. — *A bipartite graph $G = ([m] \cup [n]', E)$ is* monotone *if it is isomorphic to a convex graph such that $\alpha'_i \le \alpha'_j$ and $\beta'_i \le \beta'_j$ for all $i, j \in [m]$ with $i < j$.*

First we show that, if G is row-monotone, it is also column-monotone.

LEMMA 2.11. — *A monotone graph is biconvex, and $\alpha_{i'} \le \alpha_{j'}$, $\beta_{i'} \le \beta_{j'}$ if $i', j' \in [n]'$ and $i' < j'$.*

Proof.— For $j \in [n]$, let $s = \min\{i \in \mathcal{N}(j')\}$ and $t = \max\{i \in \mathcal{N}(j')\}$. If $s < i < t$, then $j' \ge \alpha'_t \ge \alpha'_i$ and $j' \le \beta'_s \le \beta'_i$, so $j' \in [\alpha'_i, \beta'_i] = \mathcal{N}(i)$ and hence $i \in \mathcal{N}(j')$. Thus $\mathcal{N}(j')$ is the interval $[s, t]$, so we may take $\alpha_{j'} = s$, $\beta_{j'} = t$. Hence $\alpha_{i'} = \min\{k : i' \in [\alpha'_k, \beta'_k]\}$ and $\alpha_{j'} = \min\{k : j' \in [\alpha'_k, \beta'_k]\}$, so $i' < j'$ implies $\alpha_{i'} \le \alpha_{j'}$. Similarly $i' < j'$ implies $\beta_{i'} \le \beta_{j'}$. □

Next we show a "forbidden submatrix" characterisation of monotone graphs, extending that of Lubiw [25] for chordal bipartite graphs.

LEMMA 2.12. — *A bipartite graph is monotone if and only if it is isomorphic to a graph G such that $A(G)$ has none of the following as an induced 2×2 submatrix :*

$$\ulcorner \text{ (Gamma) } : \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}, \ \lrcorner \text{ (backwards L) } : \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}, \ / \text{ (slash) } : \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

Proof.— Suppose G is monotone, but A contains $\ulcorner$ or $\boldsymbol{/}$ in rows i and j, with $i < j$. Then row-convexity implies $\beta'_i > \beta'_j$, a contradiction. Similarly, if $A(G)$ contains a $\lrcorner$, then row-convexity implies $\alpha'_i > \alpha'_j$, again a contradiction. Thus, if G is a monotone graph, $A(G)$ cannot contain $\ulcorner$, $\lrcorner$ or $\boldsymbol{/}$.

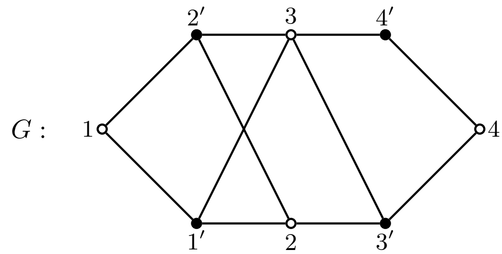
Now assume $A(G)$ contains no $\ulcorner$, $\lrcorner$ or $\boldsymbol{/}$. Suppose $\mathcal{N}(i)$ is not an interval, so there exist $j' < k' < l'$ so that $(i, j'), (i, l') \in E$, but $(i, k') \notin E$. Since $\mathcal{N}(k') \neq \emptyset$, there exists $s \in [n]$ such that $(s, k') \in E$. If $s < i$, then $A(G)$ contains the first configuration below, which is either a $\ulcorner$ or a $\boldsymbol{/}$, a contradiction. If $s > i$, then $A(G)$ contains the second configuration below, which is either a $\lrcorner$ or a $\boldsymbol{/}$, also a contradiction.

$$\begin{matrix} & \begin{matrix} j' & k' \end{matrix} \\ \begin{matrix} s \\ i \end{matrix} & \begin{bmatrix} ? & 1 \\ 1 & 0 \end{bmatrix} \end{matrix} \qquad\qquad \begin{matrix} & \begin{matrix} k' & l' \end{matrix} \\ \begin{matrix} i \\ s \end{matrix} & \begin{bmatrix} 0 & 1 \\ 1 & ? \end{bmatrix} \end{matrix}$$

Therefore suppose that $i < j$, but $\alpha'_i > \alpha'_j$. Then $A(G)$ contains the first configuration below, which is a $\lrcorner$ or $\boldsymbol{/}$, a contradiction. Similarly, if $\beta'_i > \beta'_j$, $A(G)$ contains the second configuration below, which is a $\ulcorner$ or $\boldsymbol{/}$, again a contradiction. Hence G is monotone.

$$\begin{matrix} & \begin{matrix} \alpha'_j & \alpha'_i \end{matrix} \\ \begin{matrix} i \\ j \end{matrix} & \begin{bmatrix} 0 & 1 \\ 1 & ? \end{bmatrix} \end{matrix} \qquad\qquad \begin{matrix} & \begin{matrix} \beta'_j & \beta'_i \end{matrix} \\ \begin{matrix} i \\ j \end{matrix} & \begin{bmatrix} ? & 1 \\ 1 & 0 \end{bmatrix} \end{matrix} \qquad \square$$

A *bipartite permutation graph* is a *permutation graph* which is also bipartite. A graph $G = (V, E)$ is a permutation graph if there are permutations π, σ of V so that $(\pi_i, \pi_j) \in E$ if and only if $\pi_i < \pi_j$ and $\sigma_i > \sigma_j$. This can be given a *crossing* presentation, where π, σ are on parallel lines, and connected by lines (v, v), for all $v \in V$. Then $(v, w) \in E$ if and only if corresponding lines (v, v) and (w, w) cross. Spinrad, Brandstädt and Stewart [32] studied this class of graphs, and gave $O(|E|)$ time algorithms for recognising membership in the class, and for constructing a crossing representation. An example is shown in Figs. 7 and 8 below.



$$A(G): \quad \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1' & 2' & 3' & 4' \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \end{array}$$

Figure 7. — A bipartite permutation graph

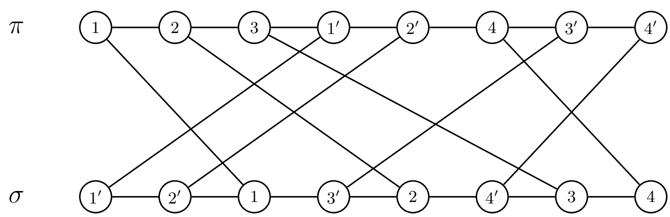


Figure 8. — Crossing representation of the graph in Fig. 7

Our reason for introducing this class of graphs is that the bipartite permutation graphs are precisely the monotone graphs.

Lemma 2.13. — *A graph is monotone if and only if it is a bipartite permutation graph.*

Proof.— The condition of Lemma 2.12 is equivalent to the following. If $(i,k'),(j,\ell') \in E$ with $i < j$ and $k' > \ell'$, then (i,ℓ'), $(j,k') \in E$. The conclusion now follows from the characterisation of bipartite permutation graphs given in [32], in particular Definition 3 and Theorem 1. □

Note that Lemma 2.12 is not a “forbidden subgraph” characterisation in the usual graph-theoretic sense. However, such a characterisation is known.

Lemma 2.14. [23, Lemma 1.46]. — *A graph is monotone if and only if it is chordal bipartite (i.e. it has no chordless cycle of length other than 4), and it contains none of the three graphs shown in Fig. 9 as an induced subgraph.*

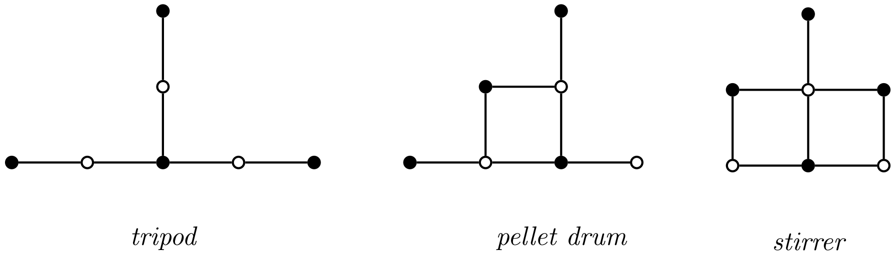


Figure 9. — The tripod, the pellet drum and the stirrer

For example, the graph G given in Fig. 10 contains the pellet drum as a subgraph.

$$A(G) = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}\!\!\begin{array}{c} \begin{array}{ccccc} 1' & 2' & 3' & 4' & 5' \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix} \end{array} \qquad \text{induced subgraph}: \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array}\!\!\begin{array}{c} \begin{array}{ccc} 3' & 4' & 5' \end{array} \\ \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \end{array}$$

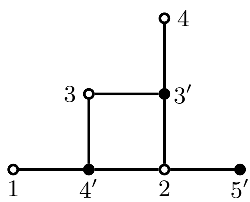


Figure 10. — A biconvex graph containing a pellet drum

Lemma 2.15. — *Monotone graphs are a proper hereditary subclass of biconvex graphs.*

Proof.— The hereditary property follows easily from the definitions. The inclusion follows from Lemma 2.11, and strict inclusion follows from the example of Fig. 10. □

To apply the switch chain to a monotone graph, we need to know whether it contains any perfect matching. If it does, we need to identify one efficiently, in order to start the chain. However, these are easy questions.

Lemma 2.16. — *A monotone graph $G = ([n] \cup [n]', E)$ contains a perfect matching if and only if it contains the* diagonal matching $\delta = \{(i, i') : i \in [n]\}$.

Proof.— We prove this by induction on n. If $n = 1$, $E = \{(1, 1')\}$, and there is nothing to prove. So, suppose $n > 1$. Clearly $(1, 1') \in E$, or else either 1 or $1'$ is an isolated vertex, and hence G has no perfect matching. We will show that there is a perfect matching M^* which contains $(1, 1')$. Therefore, suppose that M is any perfect matching in G, with $(1, 1') \notin M$. Then $(1, j'), (i, 1') \in M$ for some $i \geq 2$, $j' \geq 2'$, and we have $(1, 1') \in E$. Hence $(i, j') \in E$, or else $A(G)$ would contain a Γ.

$$\begin{array}{c} \\ 1 \\ i \end{array} \begin{array}{cc} 1' & j' \\ \left[\begin{array}{c} 1 \\ \boxed{1} \end{array}\right. & \left.\begin{array}{c} \boxed{1} \\ ? \end{array}\right] \end{array}$$

Thus $M^* = M \setminus \{(1, j'), (i, 1')\} \cup \{(1, 1'), (i, j')\}$ is a perfect matching containing the edge $(1, 1')$. Now we use induction on the graph G^* given by deleting 1 and $1'$ from G, which contains the perfect matching $M^* \setminus \{(1, 1')\}$. □

We will be particularly interested in *Hamiltonian* monotone graphs. Towards that end, we consider the graph illustrated below, the *ladder* L_n.

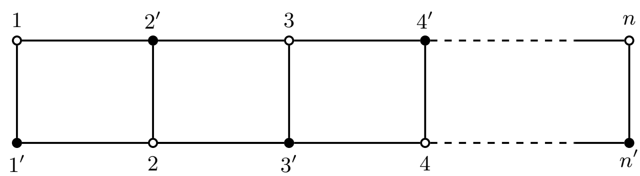


Figure 11. — The ladder

Lemma 2.17. — *L_n is a Hamiltonian monotone graph.*

Proof.— Clearly L_n is bipartite, and $\mathcal{N}(1)$, $\mathcal{N}(2)$, $\mathcal{N}(3)$, $\ldots$, $\mathcal{N}(n-1)$, $\mathcal{N}(n)$ are, respectively,

$$\{1', 2'\},\ \{1', 2', 3'\},\ \{2', 3', 4'\},\ \ldots\ , \{(n-2)', (n-1)', n'\},\ \{(n-1)', n'\},$$

so are non-empty intervals satisfying the required ordering conditions. Finally, L_n has the Hamilton cycle $1' \to 2 \to 3' \to \cdots \to n' \to n \to \cdots \to 3 \to 2' \to 1 \to 1'$. □

We have the following easy criterion for Hamiltonicity of a monotone graph.

Lemma 2.18. — *A monotone graph G is Hamiltonian if and only if it contains the ladder as a spanning subgraph.*

Proof.— If G has a spanning ladder, the Hamilton cycle in the ladder is also a Hamilton cycle in G, and so G is Hamiltonian.

If $G = ([m] \cup [n]', E)$ is Hamiltonian, it has a perfect matching, so $m = n$ and G contains the diagonal matching δ, from Lemma 2.16. We will show by induction that G contains L_n, and so has a spanning ladder. The base case is $n = 2$. Then G must be a 4-cycle, so $G = L_2$.

If $n > 2$, consider any Hamilton cycle H in G. Vertices 1 and $1'$ lie on this cycle. There are two cases :

(a) The cycle H contains the edge $(1, 1')$. Let $j' \neq 1'$ be adjacent on H to 1, and $i \neq 1$ be adjacent on H to $1'$. Since $i, j \geq 2$, biconvexity implies $(1, 2') \in E$ and $(2, 1') \in E$. Thus the three edges $(1, 1')$, $(1, 2')$, $(2, 1')$ of L_n are in E. Also $(i, j') \in E$, since G is Γ-free. Hence $i \to j' \to \cdots \to i$ is a Hamilton cycle H^* in the monotone graph G^* obtained by deleting 1 and $1'$ from G.

$$\begin{array}{c} \\ 1 \\ i \end{array} \begin{array}{cc} 1' & j' \\ \hline 1 & \mathbf{1} \\ \mathbf{1} & 1 \end{array}$$

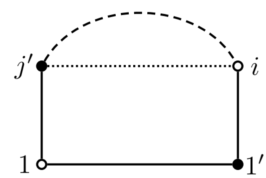


Figure 12

(b) The cycle H does not contain the edge $(1, 1') \in E$. Let j', l' be the vertices of H adjacent to 1, and i, k the vertices of H adjacent to $1'$, so that H contains paths $i \to j'$ and $k \to l'$, avoiding 1 and $1'$. Now, since G is Γ-free, (i, j'), (i, l'), (k, j'), $(k, l') \in E$. Since $(1, 1') \in E$, and $(1, j') \in E$ for $j \geq 2$, convexity implies that $(1, 2') \in E$. Similarly, since $(1, 1'), (i, 1') \in E$, with $i \geq 2$, convexity implies that $(2, 1') \in E$. Thus the three edges $(1, 1')$, $(1, 2')$, $(2, 1')$ of L_n are in E. Also $i \to j' \to k \to l' \to i$ is a Hamilton cycle H^* in the monotone graph G^* obtained by deleting 1 and $1'$ from G.

$$\begin{array}{c} \\ 1 \\ i \\ k \end{array} \begin{array}{ccc} 1' & j' & l' \\ \hline 1 & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & 1 & 1 \\ \mathbf{1} & 1 & 1 \end{array}$$

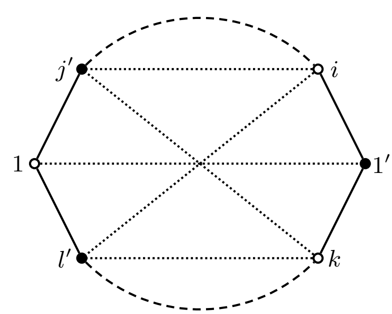


Figure 13

In both cases, the edges $(1,1')$, $(1,2')$, $(2,1')$ of L_n are in E, and we have a Hamiltonian monotone graph G^* with bipartition $[2,n] \cup [2,n]'$. It now follows by induction that G contains L_n. □

2.5. Chain graphs

Diaconis, Graham and Holmes called the simplest class of graphs they considered "one-sided restriction" graphs. These are usually called *chain graphs* in the graph theory literature [38]. They are a proper subclass of monotone graphs, which we consider in Section 2.4, and hence of chordal bipartite graphs.

DEFINITION 2.19. — *A graph $G = ([m] \cup [n]', E)$ is a chain graph if it is isomorphic to a monotone graph with $\mathcal{N}(i) = [a_i]'$ for all $i \in [m]$, and $a_1 \le a_2 \le \cdots \le a_m$.*

Hence chain graphs are a subclass of monotone graphs, given by taking $\alpha'_i = 1$, $\beta'_i = a_i$, for all $i \in [n]$. The following easy fact is then true.

LEMMA 2.20. — *$\mathcal{N}(j') = [b_j, m]$ for all $j' \in [n]'$ with $b_1 \ge b_2 \ge \cdots \ge b_n$.*

Proof.— Since $a_i \le a_{i+1}$, $(i,j') \in E$ implies $(i+1,j') \in E$. Let $b_j = \min\{i : a_i \ge j\}$. □

Chain graphs have a simple excluded subgraph characterisation. A graph is a chain graph if and only if it does not contain $2K_2$, the graph comprising two disjoint edges shown in Fig. 14, as an induced subgraph.

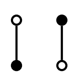


Figure 14. — The graph $2K_2$

Note that the three excluded subgraphs of Fig. 9 contain $2K_2$ as an induced subgraph, giving another proof that all chain graphs are monotone graphs.

Diaconis, Graham and Holmes [8] observed that there is a "classical" explicit formula for the permanent of a chain graph G. Of course, we must have $m = n$. Then, if $A = A(G)$,

$$\mathrm{per}(A) = \begin{cases} 0, & \text{if } a_i < i \text{ for any } i \in [n]\,; \\ \prod_{i=1}^{n}(a_i - i + 1), & \text{otherwise.} \end{cases}$$

For example, if

$$A: \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1' & 2' & 3' & 4' \end{array} \\ \left[\begin{array}{cccc} 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{array}\right] \end{array}, \quad \text{then } \operatorname{per}(A) = 2(3-1)(3-2)(4-3) = 4.$$

After noting that the first a_1 columns of A are all 1's, and hence identical, this formula can be proved by an easy induction on the row order. The proof method can also be used to sample a perfect matching uniformly at random in $O(n)$ time.

It is also possible to count and sample matchings of any given size in a chain graph in polynomial time. For positive integers m and n let $G = (V, E)$ be a chain graph, as defined above, with with $V = [m] \cup [n]'$. Let $M(i, s)$ be the number of matchings of size exactly s in the subgraph of G_i induced by $[i] \cup [n]'$. Then

$$\begin{aligned} M(i,s) &= 0 \text{ for all } s > i, \\ M(1,s) &= \begin{cases} 1, & \text{if } s = 0, \\ a_1, & \text{if } s = 1, \end{cases} \end{aligned}$$

and we have

$$M(i,s) = M(i-1,s) + (a_i - s + 1)M(i-1,s-1). \tag{2.1}$$

The first term on the right counts matchings of size s in G_i with i unmatched. The second term counts all matchings of size s in G_i with i is matched, as follows. Since G_i is a chain graph, each matching of size $(s-1)$ in G_{i-1} can be extended to a matching of size s in G_i, with i matched, in exactly $(a_i - s + 1)$ ways. Clearly we can compute $M(i, s)$ for all $i, j \in [n]$ in $O(n^2)$ time using (2.1).

As an example, we can recover the formula above for the permanent of a chain graph :

$$\begin{aligned} M(s,s) &= M(s-1,s) + (a_s - s + 1)M(s-1,s-1) \\ &= (a_s - s + 1)M(s-1,s-1), \quad \text{since } M(s-1,s) = 0, \\ &= (a_s - s + 1)(a_{s-1} - s + 2) \cdots (a_2 - 1)a_1 \quad \text{using induction and} \\ &\qquad\qquad M(1,1) = a_1. \end{aligned}$$

Matthews [27] showed, using a coupling argument, that the mixing time of the switch chain for chain graphs is bounded by $O(n^3 \log n)$. Blumberg [3] gave a detailed study of the eigenvalues of the transition matrix of the switch chain for this class, based on earlier work of Hanlon [16].

These results clearly have little computational application, but establishing the mixing time of the switch chain for graphs in the the class Chain is far from trivial. For example, there are chain graphs for which the original Jerrum and Sinclair [19] Markov chain has exponential mixing time. Consider the graph G for which $A(G)$ is lower triangular, so $A(i,j)=1$ if $i \le j$, $A(i,j)=0$ otherwise. Then $\mathrm{per}(A)=1$, from the formula above, but the graph G^* given by deleting vertices 1 and n' has $\mathrm{per}(A^*)=2^{n-3}$ by the same formula, where $A^*=A(G^*)$. Thus G has one perfect matching, but an exponential number of near-perfect matchings. Therefore the algorithm of [19] will need exponential time to sample a perfect matching almost uniformly.

2.6. Other graph classes

We have seen that the hereditary graph classes considered by Diaconis, Graham and Holmes [8] form an ascending sequence :

Chain ⊂ Monotone ⊂ Biconvex ⊂ Convex ⊂ Chordal bipartite.

The question arises as to whether this sequence is in any sense complete, or whether there are other intermediate classes. Unfortunately, the answer is that we can define an infinite number of intermediate classes by considering a suitable set of forbidden subgraphs. The following construction is quite general.

Let $\mathcal{C}$ be a hereditary graph class characterised by a set of minimal forbidden subgraphs $\mathrm{Forb}(\mathcal{C})$. It is well known, and easy to show, that any hereditary graph class has such a characterisation. However, $\mathrm{Forb}(\mathcal{C})$ may be infinite. For example Forb(Chordal bipartite) is the set Cycles of odd cycles, and even cycles of length greater than 4. For the subclasses of Chordal bipartite considered here, we will have Cycles $\subseteq \mathrm{Forb}(\mathcal{C})$. However, this need not be the case. For example Complete Bipartite ⊂ Chordal bipartite, but Forb(Complete Bipartite) $=\{K_1+K_2, C_3\}$, where K_1+K_2 is an isolated vertex plus a disjoint edge, and C_3 is a triangle.

Suppose we have classes $\mathcal{C}_1$, $\mathcal{C}_2$ with $\mathcal{C}_1 \subset \mathcal{C}_2$. Choose two graphs $F_1, F_2 \in \mathcal{C}_2 \setminus \mathcal{C}_1$ such that F_1 is a proper induced subgraph of F_2. Consider the (unique) maximal class $\mathcal{C}$ with $\mathrm{Forb}(\mathcal{C}) \subseteq \mathrm{Forb}(\mathcal{C}_2) \cup \{F_2\}$. Then $\mathcal{C} \subset \mathcal{C}_2$, since $F_2 \in \mathcal{C}_2 \backslash \mathcal{C}$, and $\mathcal{C}_1 \subset \mathcal{C}$, since $F_1 \in \mathcal{C} \backslash \mathcal{C}_1$. By iterating this construction, we can create an infinite chain of different classes between any two graph classes such that $\mathcal{C}_1 \subset \mathcal{C}_2$.

We might ask where the boundary for polynomial time exact computation of the permanent, or for polynomial time mixing of the switch chain, occur in this sequence. Unfortunately, graph classes constructed simply by giving forbidden subgraphs do not usually seem to define graphs with useful structure. And, without exploitable structure, establishing boundaries for computational properties seems very difficult.

However, some of these classes do possess structure. We will illustrate with a class which lies strictly in the gap between Chain and Monotone. Since exact counting is in polynomial time for Chain, but apparently is not for Monotone, we might ask where this new class lies with respect to this dichotomy. We consider this question below, and hence show that the boundary for polynomial time exact counting lies strictly above Chain.

Suppose we choose F_1 to be the path P_5 of length 4, and F_2 to be the graph E given by adding a pendant edge to the middle vertex of P_5.

Figure 15

Clearly $\mathrm{P}_5 \notin$ Chain, since deleting the middle vertex gives $2K_2$, but $\mathrm{E} \in$ Monotone, since it does not contain any of the subgraphs in Fig. 9.

Definition 2.21. — *A monotone graph $G = ([n] \cup [n'], E)$ is* E-free *if and only if it does not contain the graph* E*, shown in Fig. 15, as an induced subgraph.*

So Forb(E-free) $\subseteq$ {E} $\cup$ Forb(Monotone) and hence we have

$$\text{Chain} \subset \text{E-free} \subset \text{Monotone} \subset \cdots .$$

Note that E is, in fact, an induced subgraph of all three graphs in Fig. 9, as indicated below in Fig. 16, where the isolated vertex in each graph is considered as having been deleted. Consequently, we can take Forb(E-free) = {E} $\cup$ Cycles. We will consider the class E-free below.

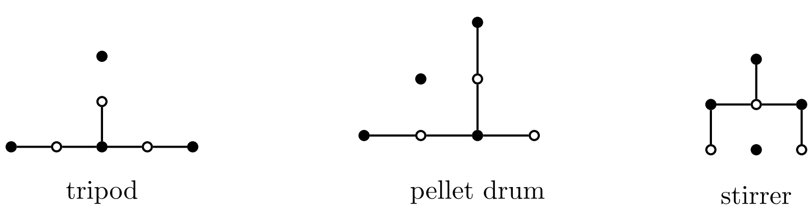


Figure 16. — Induced E's

2.7. E-free graphs

We will first consider the structure of graphs in the class E-FREE.

Let S_i $(i \in [l])$ be disjoint independent sets. Form a graph $G = (V, E)$, with $V = \bigcup_{i=1}^{l} S_i$ and $E = \{(u, v) : u \in S_i,\ v \in S_{i+1}\ \ (i \in [l-1])\}$. We will call G a *complete layered* graph, with *layers* S_i $(i \in [l])$.

Let $G(V, E)$ be a monotone graph with $V = [m] \cup [n]'$. We say that G is *aligned* if every $i \in [m]$ is adjacent to $1'$ or n', and every $j' \in [n]'$ is adjacent to 1 or m.

LEMMA 2.22. — *Every connected* E*-free graph that contains an induced path on* 7 *vertices is a complete layered graph.*

Proof.— Two vertices u and v are *false twins* if they have the same neighbourhood $\mathcal{N}(u) = \mathcal{N}(v)$. A connected graph is a complete layered graph if and only if it can be reduced to a path by identifying false twins. This operation preserves E-freeness. Hence it suffices to show that every connected E-free graph without false twins is indeed a path.

Let G be a connected E-free chordal bipartite graph that contains an induced path $P = (x_0, x_1, x_2, x_3, x_4, x_5, x_6)$. We consider a vertex v of G that does not belong to P but is adjacent to at least one vertex of P. If v has 3 or 4 neighbours on P then we may observe that G contains an E, see Figure 17. Thus every vertex of G has at most two neighbours on P.

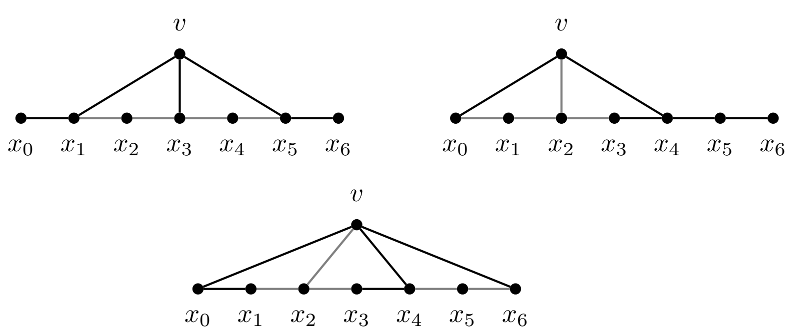


Figure 17. — The vertex v has three or four neighbours on the path P

Next we assume that v has exactly two neighbours x_i and x_j on P with $i < j$. In this case $(v, x_i, x_{i+1}, \ldots, x_j)$ is a cycle, and we conclude $j = i + 2$ since G is monotone. By symmetry we may assume $i \in \{0, 1, 2\}$. Since v and x_{i+1} are not false twins, one of them has a private neighbour, say

$u \in \mathcal{N}(x_{i+1}) \setminus \mathcal{N}(v)$. If x_{i+1} is the only neighbour of u on P then u, v, x_{i+1}, x_{i+2}, x_{i+3} and x_{i+4} induce an E in G. Otherwise u has exactly two neighbours on P. If u is adjacent to x_{i-1} then again u, v, x_{i+1}, x_{i+2}, x_{i+3} and x_{i+4} induce an E in G, otherwise we swap the rôles of u and v and find an E in G, see Figure 18. Consequently no vertex outside P has two or more neighbours on the path P.

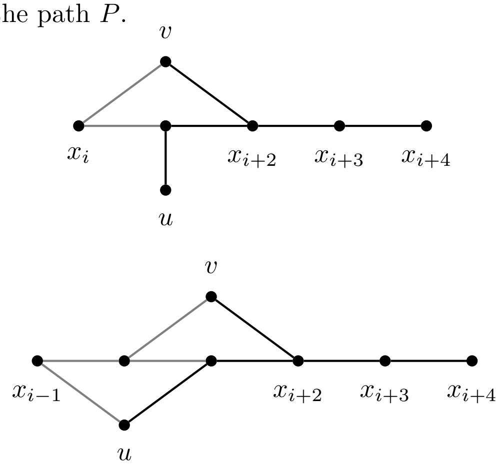


Figure 18. — The vertex v has exactly two neighbours on the path P

In the remaining case v has exactly one neighbour x_i on P. For $i \in \{2,3,4\}$ we have an E. If $i = 1$ then v and x_0 cannot be false twins, so there is a private neighbour, say $u \in \mathcal{N}(x_0) \setminus \mathcal{N}(v)$. Since x_0 is the only neighbour of u on P we can replace P by the path $(u, x_0, x_1, x_2, x_3, x_4, x_5)$ to handle this case. A symmetric argument deals with the case $i = 5$.

Thus the neighbour of v on P must be an endpoint of P. Since this is the case for every path P on 7 vertices and every vertex v with a neighbour on P, the entire graph G is a path. □

Lemma 2.23. — *Every connected* E*-free graph that does not contain an induced path on* 7 *vertices is an aligned graph.*

Proof.— First we show that all $i \in [m]$ are adjacent to $1'$ or n'. Otherwise let $j_1' = \min \mathcal{N}(i) - 1$ and $j_3' = \max \mathcal{N}(i) + 1$. Next let $i_1 = \max \mathcal{N}(j_1')$ and $i_3 = \min \mathcal{N}(j_3')$. If i_1 and i_3 have a common neighbour j' then $\{i_1, i, i_3, j_1', j', j_3'\}$ induces an E in G, see the left hand matrix below.

If $\mathcal{N}(i_1) \cap \mathcal{N}(i_3) = \emptyset$ then vertices $j' \in \mathcal{N}(i_1) \cap \mathcal{N}(i)$ and $j_2' \in \mathcal{N}(i) \cap \mathcal{N}(i_3)$ exist by connectivity, and $(j_1', i_1, j', i, j_2', i_3, j_3')$ is an induced path on 7 vertices in G.

A symmetric argument implies that every $j' \in [n]'$ is adjacent to 1 or m.

$$\begin{array}{c} \\ i_1 \\ \\ i \\ \\ i_3 \\ \\ \end{array}\begin{array}{c}
\begin{array}{ccccccc} & j_1' & & j' & & j_3' & \end{array} \\
\left[\begin{array}{ccccccccc}
 & \vdots & & \vdots & & \vdots & \\
\cdots & 1 & \cdots & 1 & \cdots & 0 & \cdots \\
 & \vdots & & \vdots & & \vdots & \\
\cdots & 0 & \cdots & 1 & \cdots & 0 & \cdots \\
 & \vdots & & \vdots & & \vdots & \\
\cdots & 0 & \cdots & 1 & \cdots & 1 & \cdots \\
 & \vdots & & \vdots & & \vdots &
\end{array}\right]\end{array}
\qquad
\begin{array}{c} \\ 1 \\ \\ k \\ k+1 \\ \\ m \end{array}\begin{array}{c}
\begin{array}{cccccc} 1' & & l' & (l+1)' & & n' \end{array} \\
\left[\begin{array}{cccccc}
1 & \cdots & 1 & 0 & \cdots & 0 \\
\vdots & & \vdots & \vdots & & \vdots \\
1 & \cdots & 1 & 1 & \cdots & 0 \\
0 & \cdots & 1 & 1 & \cdots & 1 \\
\vdots & & \vdots & \vdots & & \vdots \\
0 & \cdots & 0 & 1 & \cdots & 1
\end{array}\right]\end{array}$$

□

Lemma 2.24. — *For every aligned graph $G = ([m] \cup [n]', E)$ vertices $k \in [m]$ and $l' \in [n]'$ exist such that the subgraphs induced by $[k] \cup [l]'$ and $[k+1, m] \cup [l+1, n]'$ are complete bipartite and the subgraphs induced by $[1, k] \cup [l+1, n]'$ and $[k+1, m] \cup [1, l]'$ are chain graphs.*

Proof.— Let $k = \max \mathcal{N}(1')$ and $l' = \max \mathcal{N}(1)$. Since A, the bipartite adjacency matrix of G, is $\ulcorner$-free the vertices k and l' are adjacent as well. That is, $[k] \cup [l']$ induces a complete bipartite subgraph of G, see the right matrix above.

Furthermore $A(k+1, 1') = 0$, therefore $A(k+1, n') = 1$. Similarly, $A(1, (l+1)') = 0$ and therefore $A(m, (l+1)') = 1$ and $A(m, n') = 1$. Since A does not contain a $\lrcorner$, $A(k+1, (l+1)') = 1$, which means $[k+1, n] \cup [l+1, m]'$ also induces a complete bipartite subgraph of G.

Finally, the subgraphs of G induced by $[1, k] \cup [l+1, n]'$ and $[k+1, n] \cup [1, l]'$ are chain graphs, since G is monotone. □

Thus all E-free graphs are complete layered graphs or aligned graphs.

We will now show that the permanent of an E-free graph can be computed exactly in polynomial time. First, we show that there is a formula for complete layered graphs.

Lemma 2.25. — *Let G be a complete layered graph with layers of sizes $n_1, n_2, \ldots n_l$, and let $m_0 = 0$ and $m_i = n_i - m_{i-1}$ for $i = 1, 2, \ldots, l$. If there is an index $i < l$ such that m_i is negative or $m_l \neq 0$ then G has no perfect matching. Otherwise G has exactly $\prod_{i=2}^{l} \frac{n_i!}{m_i!}$ perfect matchings.*

Proof.— If G has a perfect matching then each layer S_i splits into parts L_i and R_i of vertices matched to vertices in S_{i-1} and S_{i+1}, respectively. Now

$L_1 = \emptyset$ and $|R_{i-1}| = |L_i|$ imply $m_i = |R_i|$ for all i, and $R_l = \emptyset$ implies $m_l = 0$. Therefore our conditions are necessary.

On the other hand, if, for all i, $m_i \geq 0$ and $m_l = 0$ then the the layers S_i split consistently into L_i and R_i, that is, such that $|R_{i-1}| = |L_i|$. There are $\binom{n_i}{m_i}$ different sets $R_i \subseteq S_i$ with $|R_i| = m_i$. The choice of R_i fixes $L_i = S_i \setminus R_i$. The subgraph of G induced by $R_{i-1} \cup L_i$ has $m_{i-1}!$ perfect matchings because it is complete bipartite with $m_{i-1} = n_i - m_i$. Hence G has $\prod_{i=2}^{l} \frac{n_i!}{m_i!}$ perfect matchings in total. □

Finally, we show that the permanent can be computed for aligned graphs.

Lemma 2.26. — *Let $G = ([n] \cup [n]', E)$ be an aligned graph, with biadjacency matrix A, then* per*(A) can be computed in $O(n^2)$ arithmetic operations on numbers of size $O(n \log n)$.*

Proof.— Let G have the biadjacency matrix A as shown below. The edges of G split into four sets E_{11}, E_{12}, E_{21} and E_{22}, that are the edge sets of four subgraphs of G induced by $V_{11} = [k] \cup [l]'$, $V_{12} = [k] \cup [l+1, n]'$, $V_{21} = [k+1, m] \cup [l]'$ and $V_{22} = [k+1, m] \cup [l+1, n]'$. The graphs $G_{11} = (V_{11}, E_{11})$ and $G_{22} = (V_{22}, E_{22})$ are complete bipartite graphs, and $G_{12} = (V_{12}, E_{12})$ and $G_{21} = (V_{21}, E_{21})$ are chain graphs.

$$A = \begin{array}{c} \\ 1 \\ \\ k \\ k+1 \\ \\ n \end{array} \begin{array}{c} \begin{array}{cccccc} 1' & & l' & (l+1)' & & n' \end{array} \\ \left[\begin{array}{cccccc} 1 & \cdots & 1 & 0 & \cdots & 0 \\ \vdots & & \vdots & \vdots & & \vdots \\ 1 & \cdots & 1 & 1 & \cdots & 0 \\ 0 & \cdots & 1 & 1 & \cdots & 1 \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \cdots & 0 & 1 & \cdots & 1 \end{array}\right] \end{array}$$

Let P be a perfect matching of G. Then

$$\begin{aligned} |P \cap E_{11}| + |P \cap E_{12}| &= k & \qquad |P \cap E_{12}| + |P \cap E_{22}| &= n - l \\ |P \cap E_{11}| + |P \cap E_{21}| &= l & \qquad |P \cap E_{21}| + |P \cap E_{22}| &= n - k \end{aligned}$$

Let $|P \cap E_{11}| = s$. Then $|P \cap E_{12}| = k - s$, $|P \cap E_{21}| = l - s$ and $|P \cap E_{22}| = n - k - l + s$, so $\max(0, k + l - n) \leq s \leq \min(k, l)$. Denote the number of matchings of size s in a chain graph G by $M(G, s)$. We say in section 2.5 that $M(G, s)$ can be computed for all s in $O(n^2)$ time. Then we can form perfect matchings in G by independently choosing a matching of size $k - s$ in G_{12}, a matching of size $l - s$ in G_{21}, and completing these by choosing arbitrary matchings in complete bipartite subgraphs of the appropriate size in G_{11}, G_{22}.

Hence the number of perfect matchings of G is

$$\mathrm{per}(A) \;=\; \sum_{s=\max(0,k+l-n)}^{\min(k,l)} M(G_{12}, k-s) \cdot M(G_{21}, l-s) \cdot s! \cdot (n-k-l+s)!\,,$$

It is easy to see that the sum can be computed in the claimed number of arithmetic operations, with numbers which are at most $n!$. □

Thus we have shown that E-free is a strict superclass of Chain for which exact evaluation of the permanent remains polynomial time.

3. Analysis of the switch chain

We have shown that

$$\text{Chain} \subset \text{E-free} \subset \text{Monotone} \subset \text{Biconvex}$$

$$\subset \text{Convex} \subset \text{Chordal bipartite}.$$

We know from Lemma 2.3 that the switch chain is ergodic for graphs in all these classes, and has diameter at most n. Note that we consider the switch chain to be ergodic on any bipartite graph for which the set of perfect matchings is empty, and this can be recognised in polynomial time.

Here we will consider the mixing time of the chain for these classes. We have seen that the switch chain may have exponential mixing time in the class Biconvex, and that the permanent can evaluated easily in the classes Chain and E-free. Therefore, it remains only to analyse the mixing time of the chain for the class Monotone.

3.1. Canonical paths and flows

Although there are other approaches to bounding the mixing time of Markov chains, here we will attempt only to apply the canonical paths approach of Jerrum and Sinclair [19]. For any symmetric Markov chain, this may be described briefly as follows.

Suppose the problem size is n. The method requires constructing a path of transitions of the chain $X = Z_1 \to Z_2 \to \cdots \to Z_\ell = Y$, between every pair of states X and Y in the state space Ω of the chain, such that the path length ℓ is at most polynomial in n, and every such path has the following, much more demanding, *canonical* property.

For any transition $Z_i \to Z_{i+1}$ on the path from X to Y, there must exist an *encoding* $W \in \Omega$, such that, given W and g other bits of additional information, we can identify X and Y uniquely. We will refer to the additional information as "guesses".

Then the mixing time T_{mix} of the chain can be bounded by $2^{O(g)}\text{poly}(n)$. Ideally, we seek $g = O(\log n)$, to give a polynomial bound on the mixing time.

3.2. Quadrangulations

We seek a canonical path of transitions of the switch chain between perfect matchings X and Y in a graph G in some hereditary subclass of Chordal bipartite. Since $X \oplus Y$ can be partitioned into alternating cycles, the canonical path from X to Y will be constructed by processing the individual cycles in $X \oplus Y$ one by one, with cycles being treated in increasing order of the value of their minimum vertex i in the $[n]$ ordering. If H is any individual cycle in this decomposition, we need only consider the case where $X \cup Y$ is a Hamilton cycle H in a smaller graph $G[H]$ in the same hereditary class. In the remainder of this section, we will simply write G rather than $G[H]$.

We wish to transform (X, Y) through successive pairs of perfect matchings

$$(X, Y) = (X_1, Y_1),\ (X_2, Y_2),\ (X_3, Y_3),\ \ldots,\ (X_k, Y_k) = (Y, X),$$

where X_{i+1} is obtained from X_i using a single move of the switch chain. Thus Y_i may be regarded as the encoding for X_i, or vice versa. We will make these switches in some subgraph Q of G such that $X \cup Y = H \subseteq Q \subseteq G$. Then the guesses we require are all edges of H which are not edges of $X_i \cup Y_i$. To ensure that X and Y are connected by the switch chain, Q should be a chordal bipartite graph, by Lemma 2.3. Subject to this restriction, we will also require that Q has as few edges as necessary.

Let H be a spanning subgraph of a chordal bipartite graph G, formally $H \subseteq G$. A chordal bipartite graph Q is a G-*quadrangulation* of H if $H \subseteq Q \subseteq G$. This quadrangulation is *minimal* if, for every G-quadrangulation Q' of H, $Q' \subseteq Q$ implies $Q' = Q$. In what follows H is a Hamiltonian cycle of G, and "quadrangulation" means "minimal G-quadrangulation". A *quadrangle* is any cycle of length 4 in G.

We will interchange, in some order, edges of the quadrangles of Q between the two matchings X and Y. To avoid confusion, we use the term "switch" for the transformation interchanging two matching edges of a quadrangle for two non-matching edges. The term "exchange" will be used for interchanging the edges of any alternating cycle between the two matchings X and Y. Our objective is to exchange the entire Hamilton cycle H.

During this process, $X_i \cup Y_i$ will essentially be a set of alternating cycles, inheriting quadrangulations from Q. Some will have already been exchanged and some remain to be exchanged. Thus the guesses will include an edge of H for every quadrangle of Q which is not in the quadrangulations of these alternating cycles. Using Lemma 3.1 below, it is easy to prove that only one edge of such each quadrangle must be guessed, since the other can then be deduced easily. Then we can use the matchings X_i, Y_i and the guesses to reconstruct the original Hamilton cycle $H = X \cup Y$.

Exchanging a quadrangulation simultaneously gives a (canonical) path from X to Y and a path from Y to X. We will denote these two paths by $X \to Y$ and $Y \to X$. Thus exchanging a quadrangle involves switching it twice, once for $X \to Y$ and once for $Y \to X$. A quadrangle which has been switched only once will be called *open*. Our first attempt at an encoding will be to perform the $X \to Y$ switch on each quadrangle in pathwidth order, and then perform the $Y \to X$ switch on each quadrangle as soon as possible, in order to minimise the number of open quadrangles. Then X_i and Y_i will be the current states of the X and Y matchings. The guesses will be one edge of each open quadrangle.

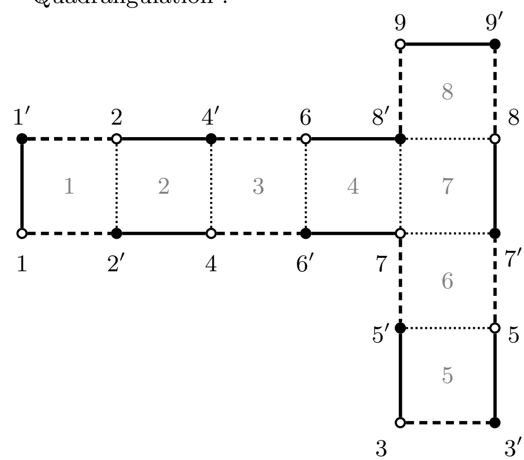
First we prove an important property of quadrangulations.

Lemma 3.1. — *Let H be a Hamiltonian cycle of a chordal bipartite graph $G = ([n] \cup [n]', E)$. Every quadrangulation Q of H is an outerplanar graph with the edges of H on the outer face.*

Proof.— We apply induction on n. If $n = 2$, then H is a quadrangle, so $Q = H$, and we are done. Otherwise, since G is chordal bipartite, H has a chord $e = (i, j')$ in E. So $H \cup \{e\}$ is a planar graph with two internal faces. Let H_1 and H_2 be their bounding cycles, and let Q_1 and Q_2 be the restrictions of Q to the vertex sets of H_1 and H_2. Since Q is a minimal quadrangulation of H, Q_1 and Q_2 are minimal too, and no edge of Q has endpoints both in H_1 and in H_2, unless one of these endpoints is i or j'. That is, Q is the union of Q_1 and Q_2. By induction hypothesis, both Q_i are outerplanar graphs with the edges of H_i on the outer face. Since e is an edge of both H_1 and H_2, Q is outerplanar with the edges of H on its outer face. $\square$

$$A(G) = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{array} \begin{array}{c} \begin{array}{ccccccccc} 1' & 2' & 3' & 4' & 5' & 6' & 7' & 8' & 9' \end{array} \\ \left[\begin{array}{ccccccccc} \boxed{1} & \textcircled{1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \textcircled{1} & 1 & 1 & \boxed{1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & \textcircled{1} & 1 & \boxed{1} & 0 & 0 & 0 & 0 \\ 0 & \boxed{1} & 1 & 1 & 1 & \textcircled{1} & 0 & 0 & 0 \\ 0 & 0 & \boxed{1} & 1 & 1 & 1 & \textcircled{1} & 0 & 0 \\ 0 & 0 & 0 & \textcircled{1} & 1 & 1 & 1 & \boxed{1} & 0 \\ 0 & 0 & 0 & 0 & \textcircled{1} & \boxed{1} & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \boxed{1} & 1 & \textcircled{1} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \textcircled{1} & \boxed{1} \end{array} \right] \end{array}$$

Quadrangulation :



Dual tree :

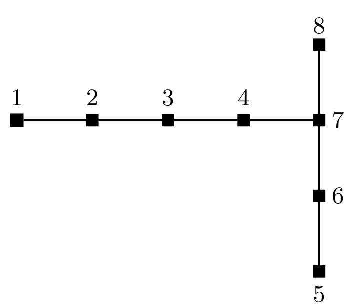


Figure 19. — A Hamilton cycle with a quadrangulation

A quadrangulation Q of H is a 2-connected outerplanar graph, so its *weak dual* is a tree T, which we call the *dual tree* of Q. (See, for example, [11, Observation 2].) Then T has $(n-1)$ vertices, which correspond to the quadrangles of Q, and its $(n-2)$ edges correspond to the internal edges of Q. A simple example is given in Fig. 19.

We apply terminology for quadrangles of Q to vertices of T, and vice versa. Thus, for example, we call a quadrangulation *linear* if T is a path.

If we switch the quadrangles of Q in some order, this order corresponds to a layout of the dual tree T, as described in section 1.1. The maximum number of quadrangles separating the exchanged part of Q from the remaining part is the vertex separation of the layout, and the minimum value of this quantity over all layouts is the pathwidth of T, $\mathrm{pw}(T)$. It follows that an optimum order in which to exchange the quadrangles is a layout which determines $\mathrm{pw}(T)$. For brevity, we will usually write $\mathrm{pw}(Q)$ rather than $\mathrm{pw}(T)$, though these quantities may differ. However, this abuse of notation causes no difficulties, since

LEMMA 3.2. — $\mathrm{pw}(T)+1 \leq \mathrm{pw}(Q) \leq 2\mathrm{pw}(T)+1$.

Proof.— Q is a 2-connected outerplanar graph, and T is its weak dual. The conclusion now follows using Lemma 1 and Theorem 1 from [11], and Theorem 4 from [6]. □

Thus, if we can find an encoding that guesses only $g = O(\mathrm{pw}(Q))$ edges, the mixing time of the switch chain can be bounded by $O(n^g)$. Since we know that $\mathrm{pw}(T) = O(\log n)$ for any n-vertex tree T, this will immediately give an $n^{O(\log n)}$ bound on the mixing time, as obtained by Matthews [27].

However, we might achieve a better bound on mixing time by using a different quadrangulation and/or layout from that used by Matthews [27]. In fact, Matthews chose a fixed layout with vertex separation $\Omega(\log n)$, independent of Q. Since any tree has pathwidth $O(\log n)$, this choice is clearly the worst case.

Therefore, the central issue is to establish the worst possible pathwidth for a quadrangulation of a Hamilton cycle in a monotone graph. However, there is a difficulty that we must resolve first. We need to be able to exchange a quadrangulation Q using only $O(\mathrm{pw}(Q))$ guesses. The solution to this problem is not completely straightforward.

3.3. Exchanging a quadrangulation

Given that we switch quadrangles in pathwidth order, we need an encoding which needs only $O(\mathrm{pw}(Q))$ guesses, in order to reconstruct H. However, we know from Lemma 3.1 that all chordal bipartite graphs allow quadrangulations of H. Since $\mathrm{pw}(Q) = O(\log n)$, if such an encoding always exists, there would be an $n^{O(\log n)}$ mixing time for chordal bipartite graphs. But there is an exponential lower bound on mixing time for the smaller class of biconvex graphs. This apparent contradiction implies, of course, that the necessary encoding cannot always exist. So we must investigate when a suitable encoding can be guaranteed to exist.

Define a *good* quadrangle of a quadrangulation to be one having exactly two non-adjacent edges in H, and a *leaf* quadrangle to be one having three edges in H. A leaf quadrangle corresponds to a leaf of the dual tree. Any other quadrangle will be called *bad*. Bad quadrangles are of two types : *junction* quadrangles, which have at most one edge in H, and *skew* quadrangles, which have exactly two adjacent edges in H. Junction quadrangles correspond to vertices in the dual tree with degree three or four.

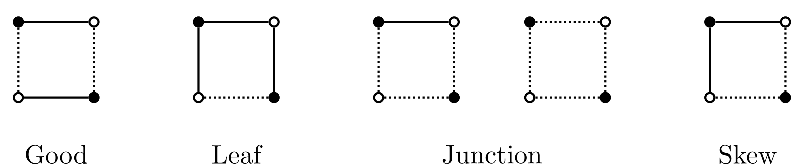


Figure 20

A good or leaf quadrangle in a quadrangulation can always be switched, in view of Lemma 3.3 below. A bad quadrangle can be switched only if at least one of its neighbouring quadrangles has been switched.

Lemma 3.3. — *Let H be an alternating cycle in a chordal bipartite graph $G = ([n] \cup [n]', E)$. If two non-adjacent edges of H are edges of a good or leaf quadrangle in a quadrangulation of H, then they belong to the same matching.*

Proof. — From Lemma 3.1, a quadrangulation is outerplanar with bounding cycle H. Suppose $\vec{H}$ is either orientation of H. If $\vec{H}$ is traversed in the direction of its orientation, every row vertex is preceded by an edge of the first matching and followed by an edge of the second. Also, since G is bipartite, all its edges connect a row vertex to a column vertex. So, if any good quadrangle has one edge in each matching, the edges of the quadrangle, together with H, form a subdivision of K_4, as illustrated below. Since an outerplanar graph cannot contain a subdivision of K_4 [5, p. 117], the quadrangulation cannot be outerplanar, a contradiction :

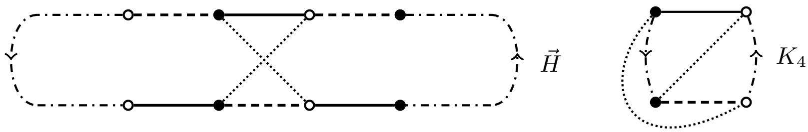


Figure 21

□

We must ensure that the number of guesses does not become large at any point during the exchange of the quadrangulation. The pathwidth of the quadrangulation is the principal obstruction to achieving this, but unfortunately there is another : any long path of bad quadrangles in the dual tree T.

A *bad* l-path in T is a path $(u_1, \ldots, u_l)$ such that every vertex u_i is a bad quadrangle, for $i \in [l]$. To exchange a bad path, we must switch every quadrangle in the path twice, but we can only switch u_i when either u_{i-1} or u_{i+1} has been switched. Hence, in exchanging a bad l-path, there is a stage at which at least l quadrangles are open.

Define an *ℓ-good* quadrangulation to be one such that there is no such bad l-path in T for any $l > \ell$. Note that the ladder is the only 0-good quadrangulation, since it is the only quadrangulation with no bad quadrangles. An ℓ-bad vertex v will be such that any ℓ-path with endpoint v in T is a bad path.

Lemma 3.4. — *An ℓ-good dual tree T with pathwidth p can be exchanged so that there are never more than $(\ell + 2)p$ open quadrangles. If T contains an ℓ-bad vertex v, then there are at least ℓ open vertices immediately after the first switch of v.*

Proof.— We assume the vertex order determining the pathwidth of T, and we switch quadrangles in this order. At any point in this numbering, we have at most p separating vertices $k_1, k_2, \ldots, k_p$. Each separating vertex k_i ($i \in [p]$) is the endpoint of a path of vertices numbered at most k_i, though these paths are not necessarily disjoint. There are at most $\ell + 2$ open vertices in each of these paths, and so there can be at most $(\ell + 2)p$ in total. (The additional 2 is because switching a path requires having two open quadrangles.) If v is ℓ-bad, we must switch all vertices along some bad

ℓ-path before we can switch v. When v is switched, all vertices in this ℓ-path are open. $\square$

Unfortunately, ℓ-good quadrangulations are not sufficient if we wish to have $\ell = O(\mathrm{pw}(Q))$, even for chain graphs. Consider the minimal chain graph F containing the Hamilton cycle :

$$H: \quad 1' \to 1 \to 2' \to 2 \to 3' \to \cdots (n-1)' \to (n-1) \to n' \to n \to 1'.$$

We will call this the *standard fan*. The biadjacency matrix $A(F)$ of F is :

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ \vdots \\ \vdots \\ n \end{array} \begin{array}{c} \begin{array}{ccccccc} 1' & 2' & 3' & 4' & \cdots\cdots & \cdots & n' \end{array} \\ \begin{bmatrix} \mathbf{1} & \mathbf{1} & 0 & 0 & \cdots & 0 & 0 \\ 1 & \mathbf{1} & \mathbf{1} & 0 & \cdots & 0 & 0 \\ 1 & 1 & \mathbf{1} & \mathbf{1} & \cdots & 0 & 0 \\ 1 & 1 & 1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & 1 & 1 & \cdots & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & 1 & 1 & 1 & \cdots & 1 & \mathbf{1} \end{bmatrix} \end{array},$$

where the Hamilton cycle H is is shown in heavy bold font. Then H has linear quadrangulations (where the dual tree is a path), for example :

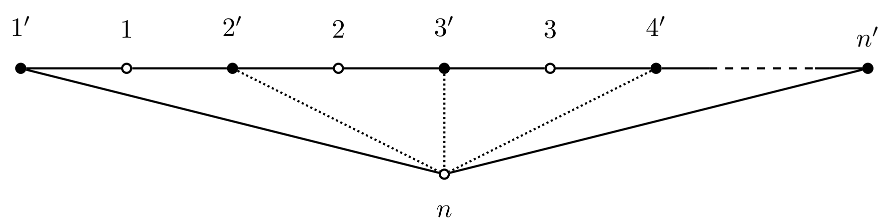


Figure 22. — Quadrangulation of the standard fan

All quadrangles are bad except for the two leaves, and the quadrangulation is not ℓ-good for any $\ell = o(n)$. Moreover, there are $\Omega(n)$-bad quadrangles, so the quadrangulation cannot be exchanged without having $\Omega(n)$ open quadrangles at some stage.

Note that it is not simply the large degree of vertex n in the quadrangulation that gives rise to this problem. The Hamilton cycle H has a linear quadrangulation with all vertex degrees at most four :

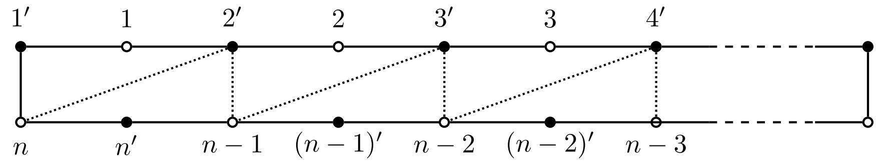


Figure 23. — Alternative quadrangulation of the standard fan

But again all quadrangles are bad except for the two leaves, the quadrangulation is not ℓ-good for any $\ell = o(n)$, and there are $\Omega(n)$-bad quadrangles. We can improve these bounds if we allow quadrangulations with larger pathwidth, but we cannot achieve $\ell = O(1)$.

Lemma 3.5. — *The standard fan has no ℓ-good quadrangulation, for any $\ell = o(\log n)$, and this bound is tight.*

Proof.— A good quadrangle corresponds to a submatrix of $A(G)$ of the form :

$$\text{(a)} : \quad \begin{bmatrix} \mathbf{1} & 1 \\ 1 & \mathbf{1} \end{bmatrix} \qquad \text{or} \qquad \text{(b)} : \quad \begin{bmatrix} 1 & \mathbf{1} \\ \mathbf{1} & 1 \end{bmatrix}.$$

Clearly A has no submatrix of type (a), and its only submatrices of type (b) are

$$\begin{array}{c} \\ k \\ n \end{array} \begin{array}{c} \begin{array}{cc} 1' & j' \end{array} \\ \begin{bmatrix} 1 & \mathbf{1} \\ \mathbf{1} & 1 \end{bmatrix} \end{array} \qquad \big(\max\{2, k\} \le j \le \min\{k+1, n\},\ k \in [n-1]\big).$$

Since all these quadrangles share the edge $(n, 1')$, and this is an edge of H, at most one of them can appear in a quadrangulation. Therefore there can be at most one good quadrangle in a quadrangulation. The remaining $(n-2)$ quadrangles are either leaves or bad. If we switch the good quadrangle, we will have a dual forest with two components. Each of these components contains only leaves and bad quadrangles and one of them, T' say, has size at least $n/2$.

The tree T' has maximum degree 4, and at least $n/2$ vertices, so it must have diameter at least $\Omega(\log n)$, using the Moore bound [15, p. 311]. Thus there must be a path of length $\Omega(\log n)$ in T', and hence in T, containing only bad quadrangles.

However, there is always an $O(\log n)$-good quadrangulation of any monotone graph, using Matthews' "binary tree" construction [27]. Since the diameter of the binary tree is $O(\log n)$, all bad paths have length $O(\log n)$. □

In particular, Lemma 3.5 implies that Matthews' approach [27] to analysing the mixing time of the switch chain on monotone graphs cannot yield a bound better than $n^{O(\log n)}$, even for chain graphs.

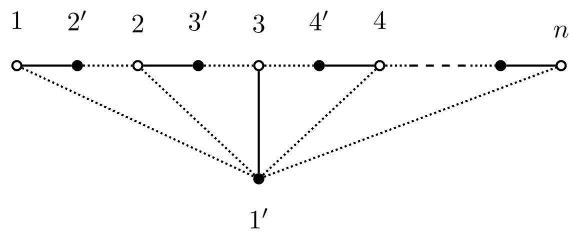
3.4. Exchanging the standard fan

Since there is no $O(1)$-good quadrangulation of a standard fan, we must find a different encoding. For any matching on the canonical path, we need the encoding to be a matching which allows us to reconstruct the original fan, using a small number of guesses. The encoding must be a perfect matching, since there can be exponentially more near-perfect matchings than perfect matchings. The method we will use has some similarities to that used by Blumberg [3] for bounded-degree convex graphs.

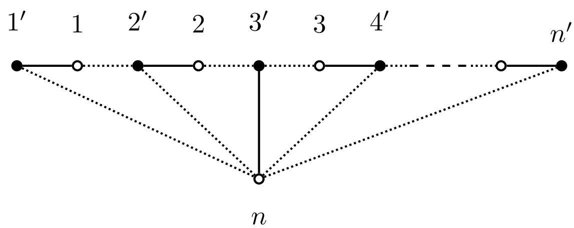
The fan has a natural ordering on the row and columns vertices, $1'$, 1, $2'$, 2, …, n', n. Let us suppose that $X = \{(1,1'), (2,2'), \ldots, (n,n')\}$ and $Y = \{(1,2'), (2,3'), \ldots, (n,1')\}$. In the canonical path construction, $X \subseteq X_G$ and $Y \subseteq Y_G$, for matchings X_G, Y_G of the whole graph G. If we switch from X to Y in the natural order (the left-to-right order in Fig. 24), we generate a sequence of isolated 2-cycles. Since there may be many other 2-cycle components in the cycle decomposition of $X_G \cup Y_G$, there may be ambiguity as to which of them are in $X \cup Y$. So reconstructing $X \cup Y$ may require guessing many edges.

A solution is to construct the path $X \to Y$ by switching in the natural linear order and to construct the path $Y \leftarrow X$ by switching in the reverse linear order. Then the state Y_i in the $Y \leftarrow X$ path which contains the edge (i', n) will be the encoding for the state X_i in the $X \to Y$ path which contains the edge $(1', i)$. Thus X_i contains the edges from Y on vertices $1, 2', \ldots, i'$, the edges from X on vertices $(i+1)', (i+1), \ldots, n$, and the edge $(1', i)$. Similarly Y_i contains the edges from X on vertices $1', 1, \ldots, (i-1)$, the edges from Y on vertices $i, (i+1)', \ldots, n'$, and the edge (n, i'). Thus all the edges of $X \cup Y$ appear in $X_i \cup Y_i$, with the exception of (i, i') and $(1', n)$. We can regard Y_i as the encoding for X_i, or vice versa. Hence we can reconstruct $X \cup Y$ by guessing only the two edges (i, i') and $(1', n)$. In fact, we need only guess the edge (i, i'). Then i is matched by $1'$ in X_i, and i' is matched by n in Y_i, so we can deduce the edge $(1', n)$.

In fact, even guessing (i, i') can be simplified. In the canonical path argument, we can identify the switch which led to X_i. This switched the edges (i, i'), $(i-1, 1')$ with the edges $(i-1, i')$, $(i, 1')$ in the matching X_{i-1}. So we need only guess one bit, to determine which of the two switched edges was (i, i').



X_3 : switching $X \to Y$



Y_3 : switching $Y \leftarrow X$

Figure 24

Note that the exchange involves switching two different quadrangulations of the fan. This is necessary, since we know that the exchange cannot be done with a single quadrangulation.

Once we know that this encoding exists, we can return to the single quadrangulation viewpoint. The canonical path simply switches $X \to Y$ from left to right, and then restores the quadrangulation by switching $Y \leftarrow X$ from right to left. The encodings for the Y switches are constructed analogously to those for the X switches. The case $n = 5$ is shown in Fig. 25. Note that, after switching $X \to Y$, there are $\Omega(n)$ open quadrangles. The revised method of encoding deals successfully this undesirable property of the quadrangulation.

To generalise this construction, let us define a cycle H in a bipartite graph $G = ([n] \cup [n]', E)$ to be a *good fan* if there is an edge $(p, q') \in H$ such that $(p, j') \in E$ for all $j' \in H \cap [n]'$, and $(i, q') \in E$ for all $i \in H \cap [n]$. The edge (p, q') will be called the *pivot edge*. A good fan can be doubly quadrangulated in the same fashion as the standard fan, and exchanged using the same encoding.

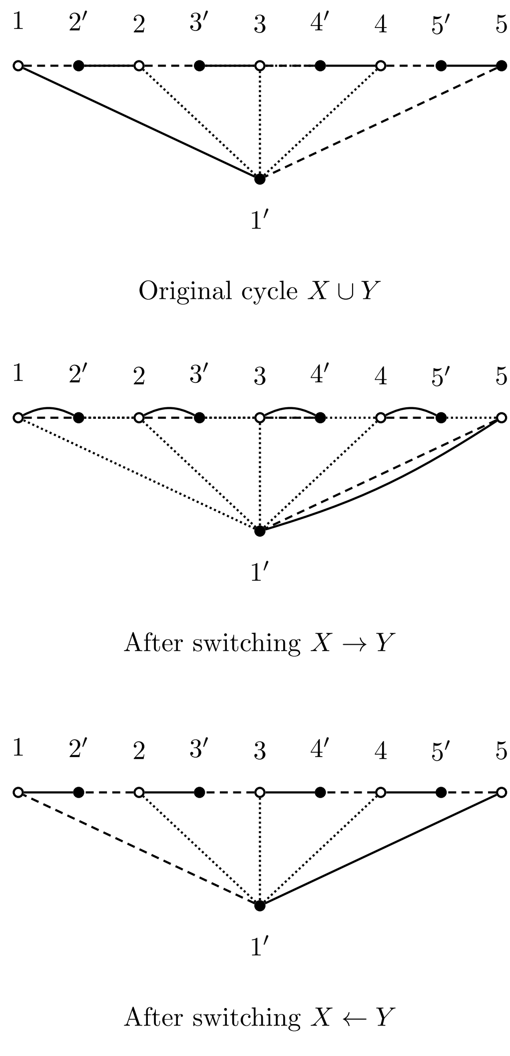


Figure 25. — Exchanging a standard fan

If a quadrangulation contains more than one good fan, they can be exchanged provided that the path between them in the dual tree T contains at least one good quadrangle. This enables us to "isolate" each good fan, so that it can be dealt with independently of the others. Such a good quadrangle will be called a *separating* quadrangle.

Thus, to switch a good fan in a quadrangulation, we must switch two separating quadrangles to isolate it, then switch the fan as described above.

The encoding is the union of the encodings for the three fragments. Then, given this encoding, we need to guess two edges in addition to the guesses for switching the fan.

3.5. Good quadrangulations

In the light of the discussion above, we make the following definition. A quadrangulation Q of a Hamilton cycle H in a monotone graph G is a *good* quadrangulation if it has the following properties.

(a) It comprises only good quadrangles, good fans and junction quadrangles.

(b) Every good fan and junction quadrangle is adjacent only to good quadrangles.

Thus the quadrangulation of the standard fan in Fig. 22 is a good quadrangulation, whereas that of Fig. 23 is not.

A good quadrangulation Q of H allows us to isolate good fans and junction quadrangles so that they can be exchanged. If there is such a good quadrangulation, then there exists an encoding such that H can be exchanged in pathwidth order, using $O(\mathrm{pw}(Q))$ guesses. If $\mathrm{pw}(Q) = O(1)$, then we will have polynomial mixing time for the switch chain in chain graphs.

The construction of Matthews [27] gives a good quadrangulation with pathwidth $O(\log n)$ for any Hamilton cycle in a monotone graph. We are currently not able to prove a better general bound in general. However, our approach gives, for example, good quadrangulations with pathwidth 1 for the ladder and the standard fan, whereas Matthews' approach always gives $\Omega(\log n)$.

3.6. An example with pathwidth 2

We are unable to produce examples where the best quadrangulation of a Hamilton cycle in a monotone graph has large pathwidth. In fact, we have no evidence that the best quadrangulation has pathwidth more than 2. However, we can give an example where the pathwidth of the dual tree is 2.

The example is an E-free graph $G = (V, E)$, a complete layered graph with vertex set $V = [21] \cup [21]'$, partitioned into layers S_i for $i = 1, 2, \ldots, 15$:

$$\begin{array}{lll} S_1 = \{1'\} & S_2 = \{1, 2\} & S_3 = \{2', 3'\} \\ S_4 = \{3, 4, 5\} & S_5 = \{4', 5', 6', 7'\} & S_6 = \{6, 7, 8, 9\} \\ S_7 = \{8', 9', 10', 11'\} & S_8 = \{10, 11, 12, 13\} & S_9 = \{12', 13', 14', 15'\} \\ S_{10} = \{14, 15, 16, 17\} & S_{11} = \{16', 17', 18'\} & S_{12} = \{18, 19\} \\ S_{13} = \{19', 20'\} & S_{14} = \{20, 21\} & S_{15} = \{21'\} \end{array}$$

and edges between consecutive layers such that $E = \{(u, v) \mid i \in [14], u \in S_i, v \in S_{i+1}\}$. For clarity we sometimes suffix vertices by the number of their layer.

The graph G has the biadjacency matrix A is shown in Fig. 26. We consider the Hamilton cycle H given by the entries 1_{r} and 1_{b} in the matrix and solid and dashed edges in Fig. 27 As above, we denote both the Hamilton cycle and its edge set by H.

$A =$

	1′	2′	3′	4′	5′	6′	7′	8′	9′	10′	11′	12′	13′	14′	15′	16′	17′	18′	19′	20′	21′
1	1_{r}	1_{b}	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1_{b}	1	1_{r}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	1_{r}	1	1_{b}	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	1	1_{b}	1	1_{r}	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	1	1	1	1	1_{b}	1_{r}	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	1_{r}	1	1	1	1_{b}	1	1	1	0	0	0	0	0	0	0	0	0	0
7	0	0	0	1	1_{b}	1	1	1	1_{r}	1	1	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	1	1_{r}	1	1	1	1_{b}	1	0	0	0	0	0	0	0	0	0	0
9	0	0	0	1	1	1	1_{b}	1	1	1	1_{r}	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	1_{r}	1	1	1	1_{b}	1	1	1	0	0	0	0	0	0
11	0	0	0	0	0	0	0	1	1_{b}	1	1	1	1_{r}	1	1	0	0	0	0	0	0
12	0	0	0	0	0	0	0	1	1	1_{r}	1	1	1	1_{b}	1	0	0	0	0	0	0
13	0	0	0	0	0	0	0	1	1	1	1_{b}	1	1	1	1_{r}	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	1_{r}	1	1	1	1_{b}	1	1	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	1	1_{b}	1	1	1	1_{r}	1	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	1	1	1_{r}	1	1	1_{b}	1	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1_{b}	1	1	1_{r}	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1_{r}	1	1	1_{b}	1	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1_{b}	1	1_{r}	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1_{r}	1	1_{b}
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1_{b}	1_{r}

Figure 26. — The biadjacency matrix of G

We know that each (minimal) quadrangulation Q of H is an outerplanar graph with the edges in H on the outer face. All the inner faces of Q are quadrangles, and the dual graph of Q is a tree T. A *chord* of H is an edge $e \in E \setminus H$.

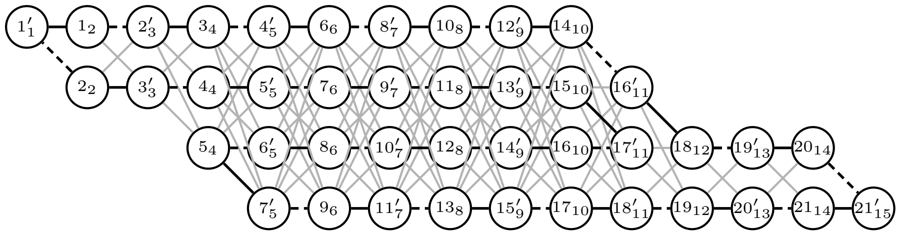


Figure 27. — The graph G, showing the edges of H

Here we are interested in the minimum pathwidth of T taken over all quadrangulations Q. The pathwidth of a tree T is at most one if and only if T is a caterpillar. That is, if removing all leaves from T results in a path, called the *body* of the caterpillar. In this case the edges that attach the leaves of T (the *feet*) to the body are called *legs*.

The Hamilton cycle (V, H) of G has several quadrangulations of pathwidth 2. Two of them are given in Fig. 28. The chords added to H in the left hand graph obviously give a good quadrangulation. This is less obvious for the chords in the right hand graph. Since we have to argue over all quadrangulations Q of G we use a circular layout of the vertices in Figs. 29(a) and 29(b).

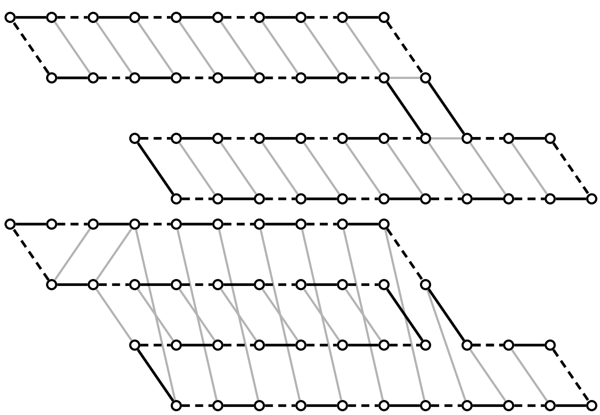


Figure 28. — Two quadrangulations of H

We want to show that all quadrangulations of H have pathwidth at least 2. We say the hamiltonian cycle *turns* in vertex $v \in V$ if the two neighbours of v in H belong to the same layer. Our graph G has four turning vertices, namely $1'_1$, $21'_{15}$, 5_4 and $17'_{11}$.

First we observe that every leaf of the dual tree of a quadrangulation Q contains a turning vertex of H. Therefore, every such tree has at most 4 leaves. The vertices $1'_1$ and $21'_{15}$ are leaves in every dual tree T. No chord of Q is incident to $1'_1$ or $21'_{15}$. Hence both vertices belong to only one quadrangle of any Q. This cannot be a skew quadrangle because the only vertices v with $\mathcal{N}(1'_1) \subseteq \mathcal{N}(v)$ are $v = 2'_3$ and $v = 3'_3$. Similarly, only vertices v with $\mathcal{N}(21'_{15}) \subseteq \mathcal{N}(v)$ are $v = 20'_{13}$ and $v = 19'_{13}$.

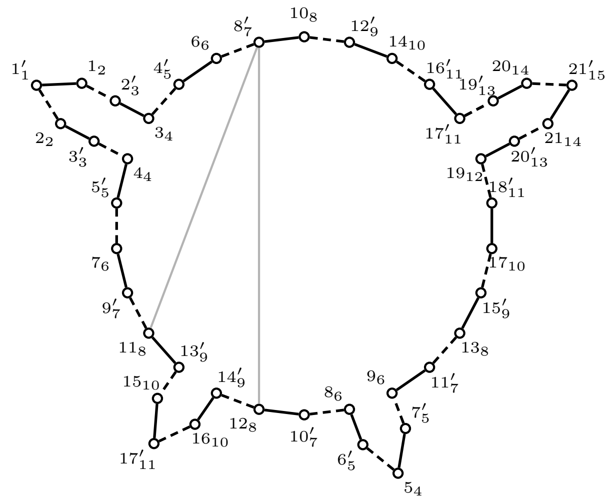
Now we consider a quadrangulation Q of H such that the dual tree T of Q is a caterpillar. The vertices $1'_1$ and $21'_{15}$ belong to leaf quadrangles of Q, and their neighbours in T are good quadrangles. Therefore the path between these neighbours is the body of the caterpillar. All other possible feet (there are at most two of them containing the vertices 5_4 and $17'_{11}$, respectively) must be adjacent to junction quadrangles on the body. Hence the endpoints of every chord of Q, except of those who cut off the two possible remaining feet of the caterpillar, are separated by $1'_1$ and $21'_{15}$ on the hamiltonian cycle.

We consider the quadrangle of Q containing the edge $(12_8, 10'_7) \in H$. This quadrangle contains either the vertex $8'_7$ or the vertex 10_8. We handle these cases separately.

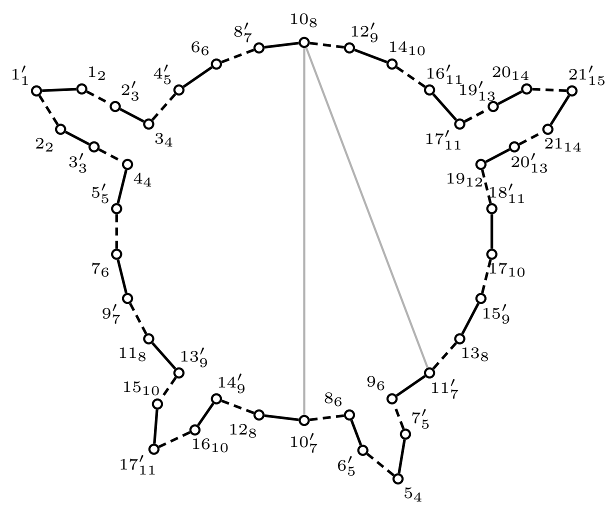
In the former case $(12_8, 8'_7)$ is a chord of Q, see Fig. 29(a). It divides H in two shorter cycles. The one containing $21'_{15}$ can be quadrangulated caterpillar-like. But if we start at $1'_1$ with a quadrangulation of the shorter cycle containing this vertex we get stuck with the chord $(11_8, 8'_7)$. This creates a cycle $(8'_7, 11_8, 13'_9, 15_{10}, 17'_{11}, 16_{10}, 14'_9, 12_8)$ of length eight. Its potential chords are $(16_{10}, 13'_9)$ and $(15_{10}, 14'_9)$. Only one of these can be present in Q, leaving a chordless cycle of length 6. This contradicts the fact that Q is a quadrangulation. Hence no quadrangulation of H contains the chord $(12_8, 8'_7)$, unless its pathwidth exceeds 1.

In the latter case $(10_8, 10'_7)$ is a chord of Q, see Fig. 29(b). It divides H into two shorter cycles. The one containing $1'_1$ can be quadrangulated caterpillar-like. But if we start at $21'_{15}$ with a quadrangulation of the shorter cycle containing this vertex we get stuck with the chord $(10_8, 11'_7)$. This creates a cycle $(10_8, 11'_7, 9_6, 7'_5, 5_4, 6'_5, 8_6, 10'_7)$ of length 8. Its potential chords are $(9_6, 6'_5)$ and $(8_6, 7'_5)$. Only one of them can be present in Q which leaves us with a chordless cycle of length 6. This contradicts the fact that Q is a quadrangulation. Hence no quadrangulation of H contains the chord $(10_8, 10'_7)$, unless its pathwidth exceeds one.

Consequently, every quadrangulation Q of H must have $\mathrm{pw}(Q) \geq 2$.



(a) $(12_8, 8'_7) \in E \setminus H$



(b) $(10_8, 10'_7) \in F \setminus H$

Figure 29

4. Conjectures and conclusions

From Section 3.3, the mixing time of the switch chain will be polynomial for monotone graphs if the following conjecture is true.

CONJECTURE 1. — *For any Hamilton cycle H in a monotone graph $G = ([n] \cup [n]', E)$, there exists a good quadrangulation Q with* $\mathrm{pw}(Q) = O(1)$.

A weaker conjecture is

CONJECTURE 2. — *For any Hamilton cycle H in a monotone graph $G = ([n] \cup [n]', E)$, there exists a quadrangulation Q with* $\mathrm{pw}(Q) = O(1)$.

We can show the following :

(a) Conjecture 1 is true for the subclass CHAIN.
(b) Conjecture 1 is a consequence of Conjecture 2 and (a).

Thus Conjecture 2 is an interesting graph-theoretic question, and we have no evidence that it is untrue. We have shown in Section 3.6 that we may have $\min_Q \mathrm{pw}(Q) \geq 2$ for every quadrangulation, but we are unable to give any example where $\min_Q \mathrm{pw}(Q) > 2$.

We omit the proofs of (a) and (b) above, since they are lengthy, and we have recently developed a different, though related, approach to bounding the mixing time of the switch chain on monotone graphs. Using this alternative approach, we can show polynomial mixing time for the switch Markov chain. This analysis will appear elsewhere. The result clearly reduces the significance of Conjecture 2, but does not imply it, so we believe that it remains an interesting graph-theoretic question. And a proof of polynomial time mixing for monotone graphs increases the likelihood that it is true.

Acknowledgements. — We thank Mary Cryan and Mark Jerrum for useful discussions.

Bibliography

[1] ALDOUS (D.) and FILL (J. A.). — Reversible Markov chains and random walks on graphs. 2002. `http://www.stat.berkeley.edu/~aldous/RWG/book.html`. Unfinished monograph, recompiled 2014.
[2] BEZÁKOVÁ (I.), STEFANKOVIC (D.), VAZIRANI (V. V.), and VIGODA (E.). — Accelerating simulated annealing for the permanent and combinatorial counting problems. SIAM Journal on Computing, 37(5), p. 1429-1454 (2008).

[3] Blumberg (O.). — Permutations with interval restrictions. PhD thesis, Stanford University (2012).
[4] Booth (K. S.) and Lueker (G. S.). — Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms. Journal of Computer and System Sciences, 13(3), p. 335-379 (1976).
[5] Brandstädt (A.), Le (V. B.), and Spinrad (J. P.). — Graph Classes : A Survey, volume 3 of Monographs on Discrete Mathematics and Applications. SIAM, Philadelphia (1999).
[6] Coudert (D.), Huc (F.), and Sereni (J.-S.). — Pathwidth of outerplanar graphs. Journal of Graph Theory, 55(1), p. 27-41 (2007).
[7] Courcelle (B.), Makowsky (J. A.), and Rotics (U.). — On the fixed parameter complexity of graph enumeration problems definable in monadic second-order logic. Discrete Applied Mathematics, 108(1-2), p. 23-52 (2001).
[8] Diaconis (P.), Graham (R.), and Holmes (S. P.). — Statistical problems involving permutations with restricted positions. In State of the art in probability and statistics, volume 36 of Lecture Notes-Monograph Series, p. 195-222. Institute of Mathematical Statistics (2001).
[9] Dyer (M.). — Approximate counting by dynamic programming. In Proc. 35th Annual ACM Symposium on Theory of Computing, p. 693-699. ACM Press (2003).
[10] Ellis (J.A.), Sudborough (I.H.), and Turner (J.S.). — The vertex separation and search number of a graph. Information and Computation, 113(1), p. 50-79 (1994).
[11] Fomin (F.) and Bodlaender (H.). — Approximation of pathwidth of outerplanar graphs. In Andreas Brandstädt and VanBang Le, editors, Graph-Theoretic Concepts in Computer Science, volume 2204 of Lecture Notes in Computer Science, p. 166-176. Springer (2001).
[12] Fürer (M.). — A natural generalization of bounded tree-width and bounded clique-width. CoRR, abs/1402.1810 (2014).
[13] Glover (F.). — Maximum matching in a convex bipartite graph. Naval Research Logistics Quarterly, 14(3), p. 313-316 (1967).
[14] Golumbic (M. C.). — Algorithmic graph theory and perfect graphs, volume 57 of Annals of Discrete Mathematics. North-Holland, Amsterdam (2004).
[15] Gross (J.L.) and Yellen (J.). — Handbook of Graph Theory. CRC Press (2004).
[16] Hanlon (P.). — A random walk on the rook placements on a Ferrer's board. The Electronic Journal of Combinatorics, 3 :R26 (1996).
[17] Harvey (D. J.) and Wood (D. R.). — The treewidth of line graphs. Sep. 2014.
[18] Jerrum (M.). — Counting, sampling and integrating : algorithms and complexity. ETH Zürich Lectures in Mathematics. Birkhäuser, Basel (2003).
[19] Jerrum (M.) and Sinclair (A.). — Approximating the permanent. SIAM Journal on Computing, 18(6), p. 1149-1178, December 1989.
[20] Jerrum (M.), Sinclair (A.), and Vigoda (E.). — A polynomial-time approximation algorithm for the permanent of a matrix with non-negative entries. Journal of the ACM, 51, p. 671-697 (2004).
[21] Jerrum (M.), Valiant (L.), and Vazirani (V.). — Random generation of combinatorial structures from a uniform distribution. Theoretical Computer Science, 43, p. 169-188 (1986).
[22] Kinnersley (N. G.). — The vertex separation number of a graph equals its pathwidth. Information Processing Letters, 42(6), p. 345-350 (1992).
[23] Köhler (E.). — Graphs without asteroidal triples. PhD thesis, Technische Universität Berlin (1999).
[24] Levin (D. A.), Peres (Y.), and Wilmer (E. L.). — Markov chains and mixing times. American Mathematical Society (2006).

[25] LUBIW (A.). — Doubly lexical orderings of matrices. SIAM Journal on Computing, 16 :854-879 (1987).
[26] MAKEDON (F.) and SUDBOROUGH (I. H.). — On minimizing width in linear layouts. Discrete Applied Mathematics, 23(3), p. 243-265 (1989).
[27] MATTHEWS (J.). — Markov chains for sampling matchings. PhD thesis, Edinburgh University (2008).
[28] OKAMOTO (Y.), UEHARA (R.), and UNO (T.). — Counting the number of matchings in chordal and chordal bipartite graph classes. In Graph-Theoretic Concepts in Computer Science, volume 5911 of Lecture Notes in Computer Science, p. 296-307. Springer (2010).
[29] ROBERTSON (N.) and SEYMOUR (P. D.). — Graph minors I : Excluding a forest. Journal of Combinatorial Theory, Series B, 35(1), p. 39-61 (1983).
[30] SKODINIS (K.). — Construction of linear tree-layouts which are optimal with respect to vertex separation in linear time. Journal of Algorithms, 47(1), p. 40-59, April 2003.
[31] SPINRAD (J.). — Nonredundant 1's in Γ-free matrices. SIAM J. Discrete Math., 8(2), p. 251-257 (1995).
[32] SPINRAD (J.), BRANDSTÄDT (A.), and STEWART (L.). — Bipartite permutation graphs. Discrete Applied Mathematics, 18(3), p. 279-292 (1987).
[33] SPINRAD (J.P.). — Efficient Graph Representations. : The Fields Institute for Research in Mathematical Sciences. Fields Institute monographs. American Mathematical Soc. (2003).
[34] TUCKER (A.). — A structure theorem for the consecutive 1's property. Journal of Combinatorial Theory, Series B, 12(2), p. 153-162 (1972).
[35] UEHARA (R.). — Linear time algorithms on chordal bipartite and strongly chordal graphs. In Automata, Languages and Programming, 29th International Colloquium, p. 993-1004 (2002).
[36] VALIANT (L. G.). — The complexity of computing the permanent. Theoretical Computer Science, 8, p. 189-201 (1979).
[37] WEST (D. B.). — Introduction to Graph Theory. Prentice Hall, 2nd edition (2001).
[38] YANNAKAKIS (M.). — Computing the minimum fill-in is NP-complete. SIAM J. Alg. Disc. Meth., 2, p. 77-79 (1981).