Sampling regular graphs and a peer-to-peer network

Colin Cooper
Department of Computer Science
Kings College
London WC2R 2LS, U.K.
ccooper@dcs.kcl.ac.uk

Martin Dyer
School of Computing
University of Leeds
Leeds LS2 9JT, U.K.
dyer@comp.leeds.ac.uk

Catherine Greenhill[*]
School of Mathematics
The University of New South Wales
Sydney NSW 2052, Australia
csg@unsw.edu.au

Abstract

This paper has two parts. In the first part we consider a simple Markov chain for d-regular graphs on n vertices, where $d = d(n)$ may grow with n. We show that the mixing time of this Markov chain is bounded above by a polynomial in n and d. In the second part of the paper, a related Markov chain for d-regular graphs on a varying number of vertices is introduced, for even constant d. This is a model for a certain peer-to-peer network. We prove that the related chain has mixing time which is bounded above by a polynomial in N, the expected number of vertices, provided certain assumptions are met about the rate of arrival and departure of vertices.

1 Introduction

In the first part of this paper we show that a simple and natural Markov chain for generating random regular graphs is rapidly mixing. Further introduction is given in Section 1.1 and the analysis can be found in Section 2.

In the second part we use this result to model a peer-to-peer network implementation of Bourassa and Holt [4] (see also [11]), called the *SWAN network*, whose underlying topology is always a regular graph. Bourassa and Holt conjectured that their network acquires properties of a *random* regular graph as it evolves, and thus will have attractive properties such as high connectivity and logarithmic diameter, with high probability.

[*] Research supported by the UNSW Faculty Research Grants Scheme

Their claim was supported by computer simulation. We extend our Markov chain analysis to give theoretical justification of their conjecture. Further introduction is given in Section 1.2 and the analysis can be found in Section 3.

1.1 Sampling regular graphs

The problem of sampling graphs with a given degree sequence (on a fixed number of vertices) has been well studied. There is particular interest in the special case of sampling d-regular graphs on n vertices, where $d = d(n)$ may grow with n. The earliest method for generating d-regular graphs uniformly at random was implicit in the paper of Bollobás [3] (see also [1, 28]). This uses the configurations model and has polynomial expected runtime only when $d = O(\sqrt{\log n})$. Other expected polynomial-time algorithms for uniform generation were described by Frieze [8] and McKay and Wormald [21] (for degrees $o(n^{1/5})$ and $o(n^{1/11})$, respectively).

Simpler algorithms can be used if we are content to sample regular graphs *approximately* uniformly. Tinhofer [26] describes one but does not bound how far away the resulting probability distribution is from uniform. Jerrum and Sinclair [14] give a polynomial time approximately uniform generator for this problem: that is, the probability of a particular graph appearing is within ratio $1 + \varepsilon$ of the uniform probability, where $\varepsilon > 0$ is part of the input. The running time is polynomial in n and $\log(\varepsilon^{-1})$ for any $d = d(n) \leq n/2$ (and by complementation, higher degrees can be handled). Their algorithm uses a Markov chain which samples perfect and near-perfect matchings of a related graph, and extends to certain non-regular degree sequences (see [13]). Steger and Wormald [25] described an algorithm which is a modification of the configurations model and is practical for degrees up to $o(n^{1/3})$. They proved that the output is asymptotically uniform for degrees up to $o(n^{1/28})$. (That is, as n gets larger the distribution of the output is closer and closer to uniform, but there is no parameter ε controlling the error). Kim and Vu [18] improved the analysis of this algorithm, proving that the output of the Steger–Wormald algorithm is asymptotically uniform for degrees up to $n^{1/3-\varepsilon}$, for any positive constant ε.

Kannan, Tetali and Vempala [17] study the mixing of a simple Markov chain $\mathcal{M}$ on regular bipartite graphs. The moves of the Markov chain "switch" a pair of non-adjacent edges (i, k), (j, ℓ), replacing them with the edges (i, ℓ), (j, k). They give a proof of rapid mixing for the bipartite case, but this does not appear complete in every respect, and it is certainly unclear how it extends to generating nonbipartite graphs. We give a proof which holds for all regular graphs which provides, as a special case, the result given in [17] for sampling regular bipartite graphs.

The argument in [17] uses canonical paths. Goldberg and Jerrum [10] give a different way of defining canonical paths for this Markov chain. In Section 2 we will adapt their approach to give a proof of rapid mixing for all d-regular graphs (bipartite or not). Specifically, we prove the following theorem. (For the definition of mixing time, see Section 1.3.)

Theorem 1. *Let $\Omega_{n,d}$ be the set of all d-regular graphs on the vertex set $\{1, \ldots, n\}$,* *where $d = d(n) \geq 3$ may grow with n. Let $\tau(\varepsilon)$ be the mixing time of the Markov chain $\mathcal{M}$ with state space $\Omega_{n,d}$ and the transition procedure given in Figure 1. Then*

$$\tau(\varepsilon) \leq d^{15}\, n^8\, \left(dn \log(dn) + \log(\varepsilon^{-1})\right).$$

This bound on $\tau(\varepsilon)$ is polynomial time, but probably very far from the truth. One might guess that $O(n \log n)$ steps would suffice for constant d, but this seems beyond the reach of known proof techniques.

An obvious question is whether the analysis of the Markov chain $\mathcal{M}$ can be extended to non-regular degree sequences. Bezáková, Bhatnagar and Vigoda [2] gave an algorithm for sampling bipartite graphs with any degree sequence almost uniformly in polynomial time. Their approach uses simulated annealing (as in Jerrum, Sinclair and Vigoda's algorithm for the permanent [15]) with a greedy start, and also uses the canonical paths defined by Goldberg and Jerrum which we use in this paper. It is an open problem to show that the Markov chain $\mathcal{M}$ mixes rapidly for general degree sequences, even for bipartite graphs.

For future reference we note that there is a well known [1] asymptotic expression for the number of d-regular graphs on n vertices, namely

$$|\Omega_{n,d}| \sim \sqrt{2}\, e^{-(d^2-1)/4} \left(\frac{d^{d/2}}{e^{d/2}\, d!}\right)^n n^{dn/2}. \qquad (1)$$

1.2 Peer-to-peer networks

A *peer-to-peer* (P2P) network is a decentralised, dynamic network for sharing data and computing resources among the vertices (participants) of the network. The network is dynamic in the sense that the vertices join and leave the network. The participants must follow the protocol of the network, and if they do, then the network should maintain some desirable properties. The properties which the designers of protocols would like to ensure include connectivity, low degree and small diameter of the network. In addition, the protocol should ideally be simple, local (not requiring global knowledge of the network) and robust (that is, be able to sustain a wide range of the joining/leaving patterns and possibly be also able to sustain adversarial attacks on the network).

The low diameter of a changing network is usually maintained by imposing some randomness on the new connections. Pandurangan, Raghavan and Upfal [22] proposed a protocol which ensures that the network is connected and has logarithmic diameter with high probability, and has always bounded degree. The crucial feature of their protocol is a central cache, which maintains addresses of a random subset of the set of vertices and has to be accessed each time a node joins the network. The requirement of having a central cache leads to an infrastructure which is asymmetric and has a potential bottleneck.

Cooper, Klasing and Radzik [6] propose a decentralised protocol based on random walks: each joining vertex v donates a fixed number of tokens with its address. These tokens perform a random walk on the network and can be used by any vertex which requires them. A vertex joining the network (in an arbitrary manner) collects a fixed number of these tokens and establishes connections to the vertices which emitted them. Similarly, a vertex which has lost a connection can obtain a new one using the next token to visit it. The network is robust under adversarial deletion of vertices and edges and actively reconnects itself.

Bourassa and Holt [4, 11] proposed a fully decentralised protocol which is based on random walks: if a vertex v in the network needs an address of a random vertex, then v initiates a random walk and uses an address of the vertex reached at some specified step of the walk. We describe the exact mechanism by which a vertex joins or leaves the network in Section 3 below. Bourassa and Holt conjecture that their protocol produces random regular graphs, so the properties of such graphs (low diameter, good connectivity) should hold. In Section 3 we will show that this conjecture is true under reasonable assumptions, discussed below, about the dynamic behaviour of the network.

An important aspect of modeling such networks is to propose a mechanism for joining and leaving which allows the network to maintain a finite but varying size. The approach adopted in [22] is for the arrival rate of participants to be Poisson with parameter ν and for the connect time (that is, the time a participant stays connected to the network) to be exponential with parameter μ. This system corresponds to a queuing process with infinitely many service channels, and is discussed in Feller [7, XVII]. In particular [7] gives the generating function of the time dependent state distribution as a function of initial size. For an initially empty system, the distribution at time t is Poisson with parameter $\nu(1 - e^{-\mu t})/\mu$ as was deduced in [22] by an alternative method. The limiting distribution of the number of vertices is Poisson with parameter ν/μ, and thus the asymptotic size of the network has expected value ν/μ.

A more general approach, adopted in this paper is to model the network size as a Markov process with state space $\mathbb{N}$ and transition probabilities

$$p_{n,n+1} = \frac{\mu_n}{\mu_n + n\nu_n}, \qquad q_{n,n+1} = \frac{n\nu_n}{\mu_n + n\nu_n}.$$

The case where $\nu_n = \nu$, $\mu_n = \mu$ for all n corresponds to the process of [22].

In Section 3 we describe a Markov chain for d-regular graphs on a varying number of vertices, with d a fixed even constant. This Markov chain is a model for the Bourassa and Holt protocol [11]. Using a multicommodity flow argument we prove that this Markov chain mixes in time bounded above by a polynomial in the expected number of vertices of the graph, under some reasonable assumptions about the arrival and departure rate of vertices.

Other work by the theoretical computer science community to model and analyse peer-to-peer networks includes [9, 19].

1.3 Markov chain definitions

Before we give these details, here are some standard definitions. Let $\mathcal{M}$ be an ergodic, time-reversible Markov chain on the finite state space Ω with transition matrix P and stationary distribution π. The *total variation distance* between two probability distributions σ, π on Ω is given by

$$d_{\mathrm{TV}}(\sigma,\pi) = \tfrac{1}{2}\sum_{x\in\Omega} |\sigma(x) - \pi(x)|.$$

The *mixing time* $\tau(\varepsilon)$ is defined by

$$\tau(\varepsilon) = \max_{x\in X} \min\left\{T \geq 0 \mid d_{\mathrm{TV}}(P_x^t,\pi) \leq \varepsilon \text{ for all } t \geq T\right\},$$

where P_x^t is the distribution of the state X_t of the Markov chain after t steps from the initial state $X_0 = x$.

One method to bound the mixing time of a Markov chain is using *multicommodity flows* (see Sinclair [24]). Let $\mathcal{G}$ be the graph underlying the Markov chain $\mathcal{M}$, so that xy is an edge of $\mathcal{G}$ if and only if $P(x,y) > 0$. A *flow* in $\mathcal{G}$ is a function $f : \mathcal{P} \to [0,\infty)$ which satisfies

$$\sum_{p\in\mathcal{P}_{xy}} f(p) = \pi(x)\pi(y) \quad \text{for all } x,y \in \Omega,\ x \neq y,$$

where $\mathcal{P}_{xy}$ is the set of all simple directed paths from x to y in $\mathcal{G}$ and $\mathcal{P} = \cup_{x\neq y}\mathcal{P}_{xy}$. Extend f to a function on oriented edges by setting

$$f(e) = \sum_{p\ni e} f(p),$$

so that $f(e)$ is the total flow routed through e. Define $Q(e) = \pi(x)P(x,y)$ for the edge $e = xy$. (Here and throughout the paper we write xy for the edge $\{x,y\}$, for ease of notation.) Let $\ell(f)$ be the length of the longest path with $f(p) > 0$, and let

$$\rho(e) = f(e)/Q(e)$$

be the *load* of the edge e. The *maximum load* of the flow is

$$\rho(f) = \max_e \rho(e).$$

Using Sinclair [24, Proposition 1 and Corollary 6′], the mixing time of $\mathcal{M}$ can be bounded above by

$$\tau(\varepsilon) \leq \rho(f)\ell(f)\left(\log(1/\pi^*) + \log(\varepsilon^{-1})\right) \tag{2}$$

where $\pi^* = \min\{\pi(x) \mid x \in \Omega\}$.

2 A Markov chain on regular graphs of fixed size

Let $\Omega_{n,d}$ be the set of all d-regular graphs $G = (V,E)$ with $V = \{1,\ldots,n\}$. Here $d = d(n)$ may be a function of n, so long as $d(n) \geq 3$ for all n. In this section we analyse the Markov chain $\mathcal{M}$ described in Figure 1. (Here "u.a.r." is an abbreviation for *uniformly at random*.) This is almost the same as the Markov chain studied by [17] in the bipartite case. However, in order to ensure the validity of (2), we will add a "holding probability" of 1/2 at every move. A move of this Markov chain is called a *switch*. Note that this chain is symmetric: the number of unordered pairs of non-adjacent edges in a

```
From $G \in \Omega_{n,d}$ do
    with probability $\frac{1}{2}$ do nothing;
    otherwise
        choose two non-adjacent distinct edges $ij$, $k\ell$, u.a.r.,
        choose a perfect matching $M$ of $\{i, j, k, \ell\}$ u.a.r.,
        if $M \cap E(G) = \emptyset$ then
            delete the edges $ij$, $k\ell$ and add the edges of $M$,
        otherwise
            do nothing.
        end if;
    end;
end;
```

Figure 1: The Markov chain on $\Omega_{n,d}$

d-regular graph on n vertices is always

$$a_{n,d} = \binom{nd/2}{2} - n\binom{d}{2}, \tag{3}$$

and hence $P(X, Y) = P(Y, X) = 1/(6a_{n,d})$ if X and Y differ just by a switch. The proof that $\mathcal{M}$ is irreducible will follow from the canonical paths defined below. Since $P(X, X) > 0$ for each X, the chain is aperiodic. Therefore $\mathcal{M}$ is ergodic and time-reversible with uniform stationary distribution. We write $\mathcal{G}$ for the underlying graph of $\mathcal{M}$, so $\mathcal{G} = (\Omega_{n,d}, \Gamma)$ where each edge $e \in \Gamma$ corresponds to a transition of $\mathcal{M}$.

Now given $G, G' \in \Omega_{n,d}$, let $H = G \triangle G'$ be the symmetric difference of G and G'. We will refer to the edge set E_B of $G \setminus G'$ as *blue* and the edge set E_R of $G' \setminus G$ as *red*. Thus $H = (V, E_H)$, where $E_H = E_B \cup E_R$. The first step in defining the flow is to decompose H in a canonical way. For bipartite graphs, the symmetric difference can be written as the union of edge-disjoint alternating cycles. However, we may not be able to achieve this in the nonbipartite case. For example, if $d \geq 2$ then it is possible to find d-regular graphs G, G' with symmetric difference given by the "bowtie" graph of Figure 2. We must be able to deal with such symmetric differences.

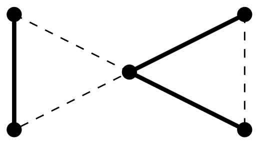

Figure 2: The bowtie graph.

2.1 Defining the flow

We define the flow using the approach of Goldberg and Jerrum [10]. The first step will be to decompose the symmetric difference H into a sequence $\mathcal{C}$ of *circuits*. A circuit $C = w_0 w_1 \cdots w_k$ is a string over the alphabet V such that $w_{i-1}w_i \in E_H$ $(i = 1, \ldots, k)$, $w_k w_0 \in E_H$, and all *edges* are distinct. Note that we do not require the *vertices* to be distinct. Thus we distinguish a circuit from a *cycle*. This usage is common, though not completely standard, in graph theory. See for example [5]. We will abuse notation to write C also for both the edge set and vertex set of the underlying graph of C. The intended meaning should be clear from the context.

Let $H = G \triangle G'$ be the above symmetric difference, with (equal) red and blue vertex degrees θ_v $(v \in V)$. We choose a *pairing* of the red and blue edges around each vertex. (That is, we give a bijection from the blue edges adjacent to v to the set of red edges adjacent to v, for each $v \in V$.) Let $\Psi(G, G')$ be the set of all such pairings, so $|\Psi(G, G')| = \prod_{v \in V} \theta_v!$. For each pairing in $\Psi(G, G')$, we construct a canonical path from G to G' in $\mathcal{G}$. Each of these paths will carry $1/|\Psi(G, G')|$ of the total flow from G to G'. The proof for the bipartite case in [17] goes via matchings and a construction of Tutte [27], which are used essentially to perform this pairing. This gives rise to additional complications with the matching edges lying entirely within the Tutte gadgets and their connections with the matching edges representing d-regular graphs. It seems to greatly simplify the analysis if we deal with this pairing directly.

Decomposition into circuits

Fix a pairing $\psi \in \Psi(G, G')$. Let the lexicographically least edge in E_H be $w_0 w_1$ $(w_0 < w_1)$. Choose the edge $w_1 w_2$ which is paired with $w_0 w_1$ at w_1. (Note that if $w_0 w_1$ is blue then $w_1 w_2$ is red, and vice-versa.) Now choose the edge $w_2 w_3$ which is paired with $w_1 w_2$ at w_2. (This edge will have the same colour as $w_0 w_1$.) In general, choose the edge $w_i w_{i+1}$ which is paired with the edge $w_{i-1} w_i$ at w_i. Note that the w_i are not necessarily distinct, but the edges are distinct. This terminates with a circuit $C_1 = w_0 w_1 \ldots w_{k-1} w_k$ when $w_k w_0$ is paired with $w_0 w_1$ at w_0. If $E_H = C_1$ then $\mathcal{C} = (C_1)$ and we are done. Otherwise, take the lexicographically least edge not in C_1 and generate a new circuit C_2 by the above procedure. Continue generating circuits until $E_H = C_1 \cup C_2 \cup \cdots \cup C_s$. Then $\mathcal{C} = (C_1, C_2, \ldots, C_s)$ and the circuits $C_1, C_2, \ldots, C_s$ are edge-disjoint. Note that each circuit C_i is an *alternating* circuit in H: as we traverse the edges they are alternately red and blue. This implies that each circuit has even length. Note also that, once the pairing has been chosen, $\mathcal{C}$ is formed without regard to the colouring of H. This property will be used later.

We form a path

$$G = Z_0, Z_1, \ldots, Z_M = G'$$

from G to G' in the underlying graph of the Markov chain (that is, to get from Z_a to Z_{a+1} we perform a switch). The path is defined by processing each circuit C_i in turn. At the end of processing C_i, its edges have been "switched" from agreeing with G to agreeing with G', and no other edges have been altered. Some may be altered temporarily while a circuit C_i is being processed, but at the end these alterations are all undone.

The canonical path is defined inductively. Suppose that

$$G = Z_0,\ Z_1,\ \ldots\ Z_r$$

is the path obtained by swapping the red and blue edges of C_1, the first circuit in $\mathcal{C}$, and

$$Z_r,\ Z_{r+1}, \ldots\ Z_N = G'$$

is the canonical path from Z_r to G' obtained by swapping the red and blue edges of $(C_2, \ldots, C_s)$ in order. Then the canonical path from G to G' corresponding to ψ is simply the concatenation of these two paths.

So it suffices to describe how to form a canonical path corresponding to a particular circuit $C = w_0 w_1 \ldots w_{k-1} w_k$. To this end we make the following definition. A *1-circuit* $S = v_0 v_1 v_2 \cdots v_t$ will be a substring of C such that $v_0 = w_0$, and w_0 appears only once in S. Usually a 1-circuit will be an alternating circuit in H, but occasionally this will not quite be true. We will show how to decompose a circuit into a sequence of 1-circuits. We *process* each 1-circuit in order, meaning we switch its red and blue edges (without permanently changing the status of any other edges). We explain later how this is done. This will give an inductive definition of the canonical path corresponding to C.

Decomposition of a circuit into 1-circuits

Let $v = w_0$, and suppose $C^{(0)} = C$ is initially the unprocessed section of C. Suppose that the current graph (on the canonical path from G to G') is Z_R. If $C^{(0)}$ is a 1-circuit then we process it, as described below, extending the canonical path as

$$G = Z_0, \ldots, Z_R, Z_{R+1}, \ldots, Z_{R+k}. \tag{4}$$

This completes the processing of C. Otherwise

$$C^{(0)} = vx_1 \cdots y_1 vx_2 \cdots y_2 \cdots vx_{\theta-1} \cdots y_{\theta-1} vx_\theta \cdots y_\theta,$$

where $\theta \leq \theta_v$. Firstly, if vx_1 and $y_1 v$ have different colours then $S = vx_1 \cdots y_1$ is a 1-circuit. We process it, extending the canonical path as in (4) and leaving

$$C^{(1)} = vx_2 \cdots y_2 \cdots vx_{\theta-1} y_{\theta-1} v \cdots vx_\theta \cdots y_\theta$$

as the unprocessed section of C. We process $C^{(1)}$ inductively. Next, if (vx_1 and $y_1 v$ have the same colour and) vx_θ and $y_\theta v$ are coloured differently, then $S = vx_\theta \cdots y_\theta$ is a 1-circuit. We process it, extending the canonical path as in (4) and leaving

$$C^{(1)} = vx_1 \cdots y_1 vx_2 \cdots y_2 v \cdots vx_{\theta-1} \cdots y_{\theta-1}$$

to be processed inductively. Finally suppose that vx_1, $y_1 v$ have the same colour, and vx_θ, $y_\theta v$ have the other colour. We process $S' = vx_\theta \cdots y_\theta vx_1 \cdots y_1$, extending the canonical path as in (4) and leaving

$$C^{(1)} = vx_2 \cdots y_2 v \cdots vx_{\theta-1} \cdots y_{\theta-1},$$

to be processed inductively. Let us call S' a *2-circuit.* However, it is not a 1-circuit, and we must show how to process it. We remark that the bipartite case is somewhat easier, since this case cannot occur. Note that the order in which we detect and process 1-circuits or 2-circuits implies that both the processed and unprocessed sections of C are continuous whenever the processing of a 1-circuit or 2-circuit is complete. (That is, these sections form continuous substrings of C, where a substring is allowed to wrap around in a cyclic fashion.)

Decomposition of a 2-circuit into 1-circuits

We now show how to process a 2-circuit S', given that we can process a 1-circuit. Suppose that we have reached the graph Z_R on the canonical path from G to G' in $\mathcal{G}$. Let

$$S' = vwx \cdots uvz \cdots y \tag{5}$$

where vw, uv belong to Z_R and vz, yv belong to G', by reversing S' if necessary (by which we mean following the pairings in the opposite direction). See Figure 3. Here solid edges belong to Z_R and dashed edges belong to G'. Recall that Z_R agrees with G on the 2-circuit S' before the processing of S' begins.

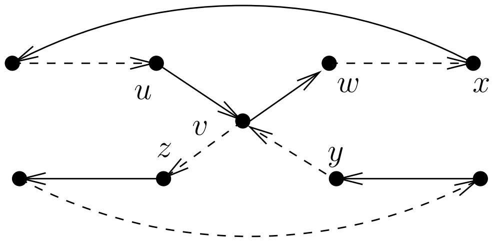

Figure 3: A 2-circuit

Since u, w, y, z are distinct neighbours of v, it follows that x is distinct from at least one of y, z. It is also distinct from v, w since they lie on successive edges. If x is only distinct from z, we will instead work with the 2-circuit

$$S' = vwx \cdots uvy \cdots z.$$

Hence we may assume that x is distinct from y in (5). Consider the edge xy. There are three cases.

(a) xy is an edge of S'.

(b) xy is not an edge of S' and xy is an edge of Z_R.

(c) xy is not an edge of S' and xy is not an edge of Z_R.

We consider these cases in order.

(a) There are eight subcases. In each we explain how to split the 2-circuit into two 1-circuits. These two 1-circuits are then processed in the given order, extending the canonical path from G to G' as

$$G = Z_0, \ldots, Z_R,\ Z_{R+1},\ \ldots, Z_{R+k}$$

after processing the first 1-circuit and

$$G = Z_0, \ldots, Z_R, \ldots, Z_{R+k},\ Z_{R+k+1},\ \ldots, Z_{R+k+\ell}$$

after processing the second 1-circuit. Later, when describing how to process the 1-circuit, we will assume that its first edge belongs to Z_R. So this is the convention followed below. The direction of the edges of the 1-circuits may not follow the direction given by the pairing, but that will not matter. The important thing is that the (edge sets of the) two 1-circuits form a partition of the (edge set of the) 2-circuit.

(a1) Suppose that $S' = vwx \cdots xy \cdots uvz \cdots y$ and $xy \notin Z_R$. Split S' into two 1-circuits $S_1 = vwx \cdots xy \cdots z$ and $S_2 = vu \cdots y$.

(a2) Suppose that $S' = vwx \cdots xy \cdots uvz \cdots y$ and $xy \in Z_R$. Split S' into two 1-circuits $S_1 = vwx \cdots xy$ and $S_2 = vu \cdots y \cdots z$.

(a3) Suppose that $S' = vwx \cdots yx \cdots uvz \cdots y$ and $yx \in Z_R$. This is similar to subcase (a1). Split S' into two 1-circuits $S_1 = vwx \cdots y \cdots z$ and $S_2 = vu \cdots xy$.

(a4) Suppose that $S' = vwx \cdots yx \cdots uvz \cdots y$ and $yx \notin Z_R$. This is similar to subcase (a2). Split S' into two 1-circuits $S_1 = vwx \cdots y$ and $S_2 = vu \cdots xy \cdots z$.

(a5) Suppose that $S' = vwx \cdots uvz \cdots xy \cdots y$ and $xy \in Z_R$. Split S' into two 1-circuits $S_1 = vwxy \cdots y$ and $S_2 = vu \cdots x \cdots z$.

(a6) Suppose that $S' = vwx \cdots uvz \cdots xy \cdots y$ and $xy \notin Z_R$ Split S'' into two 1-circuits $S_1 = vwx \cdots z$ and $S_2 = vu \cdots xy \cdots y$.

(a7) Suppose that $S' = vwx \cdots uvz \cdots yx \cdots y$ and $yx \in Z_R$. This is similar to subcase (a5). Split S' into two 1-circuits $S_1 = vwxy \cdots z$ and $S_2 = vu \cdots x \cdots y$.

(a8) Suppose that $S' = vwx \cdots uvz \cdots yx \cdots y$ and $yx \notin Z_R$. This is similar to subcase (a6). Split S'' into two 1-circuits $S_1 = vwx \cdots y$ and $S_2 = vu \cdots xy \cdots z$.

(b) Now assume that xy is not an edge of S' but xy is an edge of Z_R. We first switch the 4-cycle $yvwx$, extending the canonical path by one step to give

$$G = Z_0, \ldots, Z_R,\ Z_{R+1}.$$

Now xy is a non-edge of Z_{R+1}. We use it to give the 1-circuit $vz \cdots yx \cdots u$, which we then process. This extends the canonical path further to give

$$G = Z_0, \ldots, Z_R,\ Z_{R+1},\ Z_{R+2}, \ldots, Z_{R+1+k}.$$

Note that the edge xy is removed from Z_R when the 4-cycle is switched, but is reinstated when the 1-circuit is processed (and so is present in the graph Z_{R+1+k}). Call xy the *shortcut* edge. This is illustrated in Figure 4, where the edges drawn as arcs indicate any odd number of edges. (In the second picture the edge yx is shown as a dotted line to indicate that it has been removed from Z_R but that it may or may not belong to G'.)

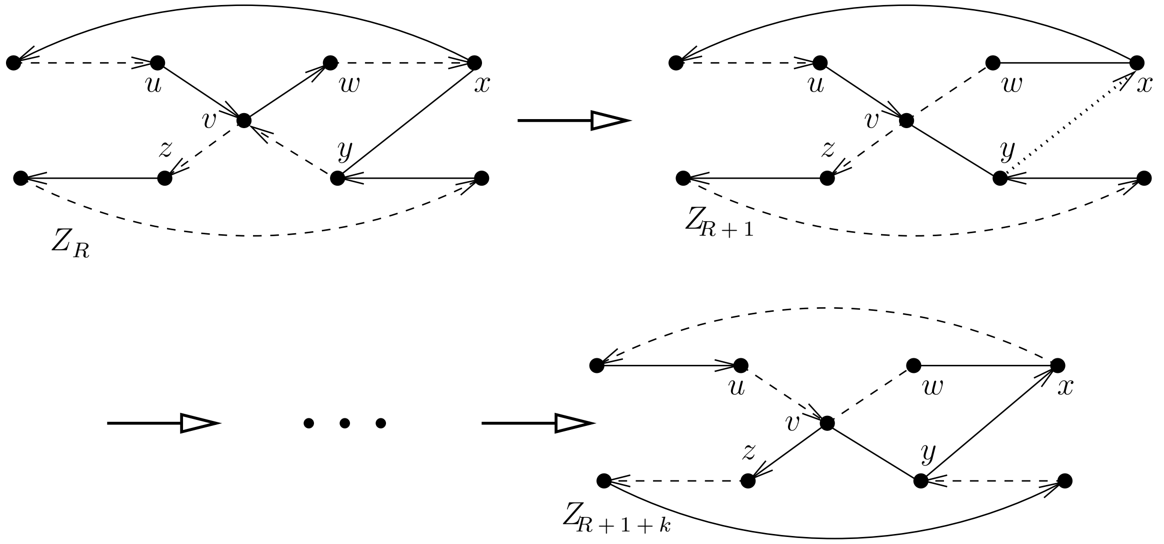

Figure 4: Case (b)

(c) Now assume that xy is not an edge of S' and xy is not an edge of Z_R. First we divert the path $yvwx$ through the non-edge yx. This gives the 1-circuit $vz \cdots yx \cdots u$, which we process. This extends the canonical path to give

$$G = Z_0, \ldots, Z_R,\ Z_{R+1},\ \ldots, Z_{R+k}.$$

Finally we switch the 4-cycle $yvwx$, extending the canonical path by one step to give

$$G = Z_0, \ldots, Z_R,\ Z_{R+1},\ \ldots, Z_{R+k},\ Z_{R+k+1}.$$

The edge xy is added to the current graph after processing the 1-circuit, so that xy is an edge of Z_{R+k}, but is removed again when the 4-cycle is switched. This is illustrated in Figure 5, where the edges drawn as arcs indicate any odd number of edges. (Again, in the first picture in the sequence, the non-edge xy of Z_R is shown as a dotted line rather than a dashed line, since it may or may not belong to G'.) Again we call xy the shortcut edge.

Note that this process works even when the 2-circuit is a bowtie as in Figure 2. Here $x = u$ and case (a) is impossible.

Also note that the 1-circuits produced in cases (b) and (c) are *not* necessarily alternating circuits in H, because of the shortcut edge. However, all other 1-circuits produced in the decomposition of a circuit are alternating circuits in H.

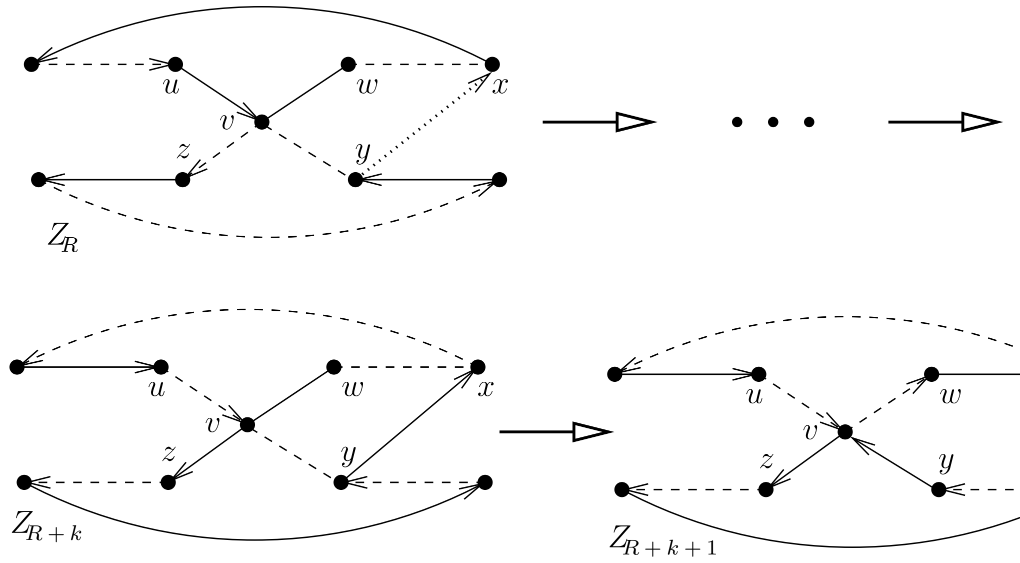

Figure 5: Case (c)

Processing a 1-circuit

To complete the definition of the canonical path from G (blue) to G' (red) corresponding to a given pairing $\psi \in \Psi(G, G')$, we need only describe how to process a 1-circuit.

Suppose that we have reached the current graph Z_R on the canonical path from G to G', and the next step is to process the 1-circuit $S = x_0x_1 \ldots x_{2k-1}$. If necessary, reverse S to ensure that the first edge of S belongs to Z_R. Then $x_{2t}x_{2t+1} \in Z_R$ for $0 \le t \le k-1$, while $x_{2t-1}x_{2t} \notin Z_R$ for $1 \le t \le k$ (identifying x_{2k} with x_0). Since Z_R agrees with G on S we can refer to edges of S as being blue (i.e. belong to $G \setminus G'$) or red (i.e. belonging to $G' \setminus G$).

First suppose that all vertices in the 1-circuit are distinct, for ease of explanation. There will be p phases in the processing of 1-circuit S, where

$$p = |\{t : 1 \le t \le k-1, \quad x_0x_{2t+1} \notin Z_R\}|.$$

As a running example, consider Figure 6. We are about to process the 1-circuit $S = x_0x_1 \cdots x_{11}$, where x_0 is the start vertex and the edge x_0x_1 is blue. The solid lines represent edges in Z_R. On the 1-circuit itself, these edges also belong to $G \setminus G'$, so they are blue edges (namely, the edges x_0x_1, x_2x_3, x_4x_5, x_6x_7, x_8x_9, $x_{10}x_{11}$). The dashed lines represent red edges which belong to $G' \setminus G$. Note that the solid chordal edges belong to Z_R, but they may or may not belong to either G or G'. We show all edges of Z_R which exist between x_0 and odd-indexed vertices in S. We call these *odd chords*. There are three odd-indexed vertices of S not joined to x_0 in Z_R, namely x_5, x_9 and x_{11}, so there will be three phases.

For the first phase, find the lowest t such that $x_0x_{2t+1} \notin Z_R$. (In the example this is $t = 2$.) Form Z_{R+1} from Z_R by deleting the edges x_0x_{2t-1}, $x_{2t}x_{2t+1}$ and replacing them by the edges x_0x_{2t+1}, $x_{2t-1}x_{2t}$. Note that either $t = k-1$ or x_{2t+1} now has

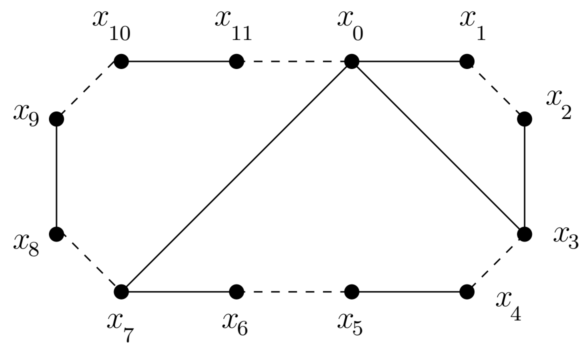

Figure 6: The 1-circuit $S = x_0x_1 \cdots x_{11}$

no Z_{R+1}-neighbours among x_{2t}, x_{2t+2}. Similarly, either $t = 1$ or x_{2t-1} has both x_{2t}, x_{2t-2} as Z_{R+1}-neighbours. If $t > 1$ then there are $t-1$ further steps in phase 1. For $j := t-1, t-2, \ldots, 1$ do: form $Z_{R+t-j+1}$ from Z_{R+t-j} by deleting the edges x_0x_{2j-1}, $x_{2j}x_{2j+1}$ and replacing them with the edges $x_0x_{2j+1}, x_{2j-1}x_{2j}$. Any chord $x_0x_{2\ell-1}$ which was in Z_R but was deleted when $j = \ell$, will be reinstated when $j = \ell - 1$. The only chord which is introduced during this phase which was not present in Z_R is the chord x_0x_{2t+1}. This is illustrated in the first row of Figure 7, for our example. In the first step, the chord x_0x_5 is introduced, which was not present in Z_R, and the chord x_0x_3 is deleted in the first step. However, the second (and final) step of phase 1 reinstates the chord x_0x_3.

This completes phase 1, and extends the path by $Z_{R+1}, \ldots, Z_{R+t}$. The edges of Z_{R+t} on C_{j_r} match those of G' on the segment $x_0x_1 \cdots x_{2t}x_{2t+1}$. Also, a chord x_0x_{2j+1} is present in Z_{R+t} if and only if it is present in Z_R, for $1 \leq j \leq t-1$. If $t = k-1$ then there is only one phase, so the processing of W is complete. Otherwise, there are further phases and x_{2t+1} still has no Z_{R+t}-neighbours in x_{2t}, x_{2t+2}. However, since the phase is complete, no vertex is joined to both its predecessor and successor around S in Z_{R+t}.

The subsequent phases are similar. For phase 2, look for the first index $q > t$ such that $x_0x_{2q+1} \notin Z_{R+t}$. Perform a step by switching x_0x_{2q-1}, $x_{2q}x_{2q+1}$ for edges x_0x_{2q+1}, $x_{2q-1}x_{2q}$. Then work backwards to fix up all segments of the 1-circuit between $j = q-1$ and $j = t+1$ (if any). There are $q - t$ steps in phase 2. Continue until all phases are complete. The two steps of phase 2 for our example produce the graphs Z_{R+3}, Z_{R+4} of Figure 7. The final phase of our example is phase 3, which only has one step, as shown in the last transition of Figure 7. In this example, there were only three phases, so the processing of S is complete. The edges of S in the final graph Z_T agree with G', and the only odd chords present are exactly those which were present at the start. Moreover, we have not added or deleted any other edges.

Now we explain how to adapt this process when some of the vertices in the 1-circuit are repeated. In fact, we need only concern ourselves with coincidences which occur between the vertices x_{2j+1}, for $1 \leq j \leq k-1$. These are the vertices which may be endvertices of odd chords at some point. Since the 1-circuit contains no repeated edges it follows that $|i-j| \geq 2$ whenever $x_{2i+1} = x_{2j+1}$.

To see why this care in is needed in this case consider the example shown in Figure 8.

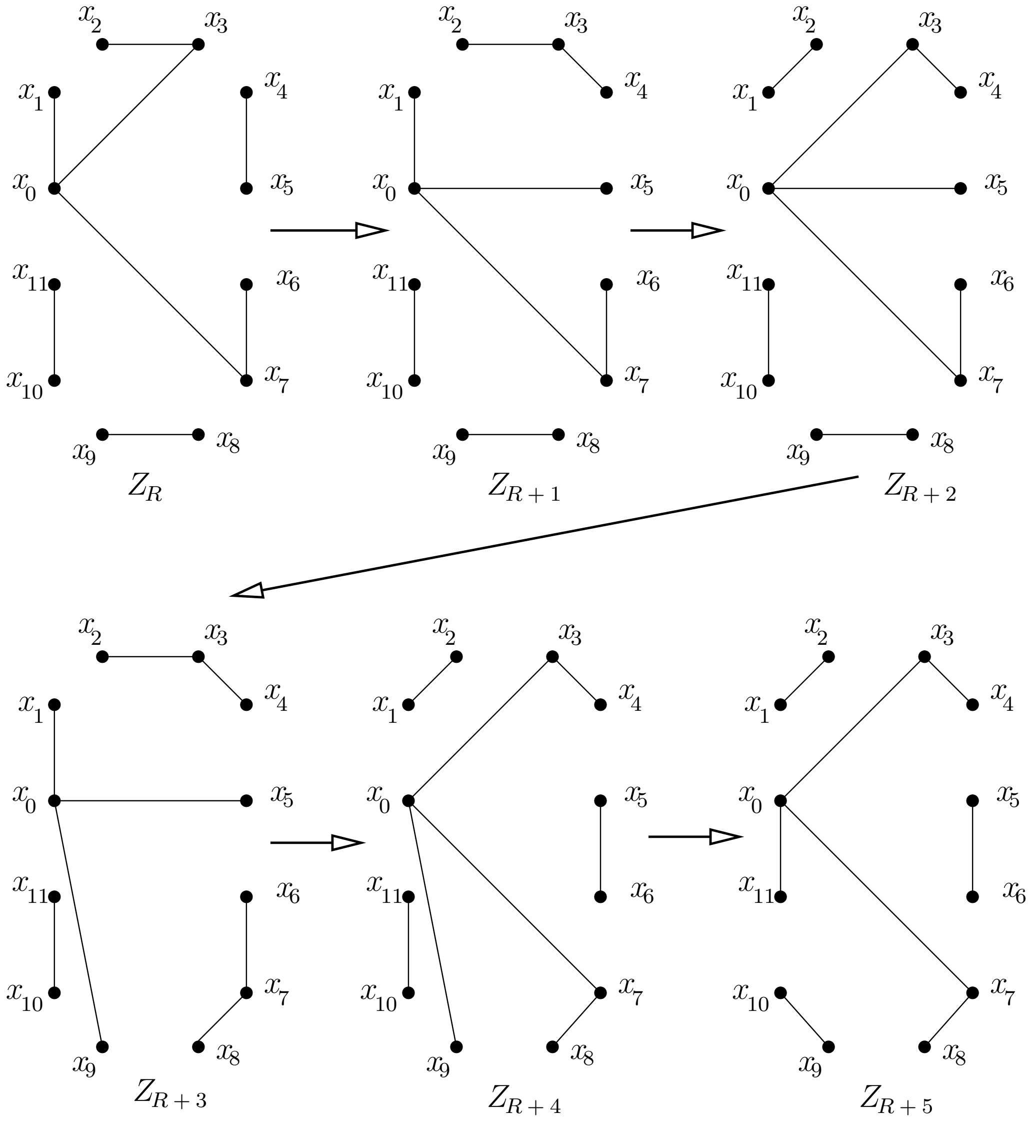

Figure 7: Processing the 1-circuit S

We wish to process the 1-circuit $S = x_0x_1 \cdots x_{11}$ where $x_3 = x_7 = x_{11}$. As in Figure 6, the solid lines are edges of Z_R, the dashed lines are edges of $G' \setminus G$ and all odd chords are shown. The edge x_0x_3 is an edge of G' and not to Z_R (as Z_R agrees with G on S before S is processed), and x_0x_5, $x_0x_9 \in Z_R$. We have still written the 1-circuit in an oval shape but the repeated vertex x_3 appears three times on the 1-circuit. It is easy to check that starting the phase by switching the 4-cycle $x_0x_3x_2x_1$ or $x_0x_7x_6x_5$ will result in a configuration where no further phases are possible. We must begin the first phase by switching the 4-cycle $x_0x_{11}x_{10}x_9$. Note that 11 is the highest odd label of the repeated vertex x_3.

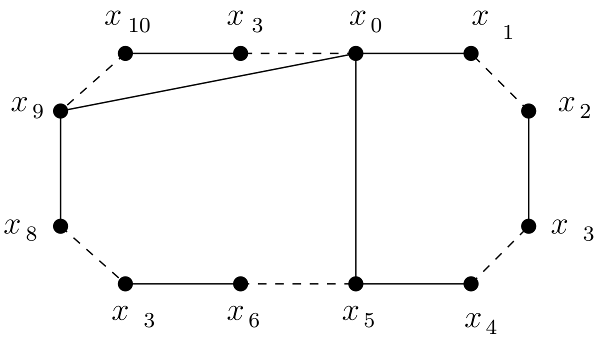

Figure 8: Vertices x_3, x_7 and x_{11} are all equal

Therefore we define

$$\mathcal{B} = \{t \mid 1 \leq t \leq k-1, \quad x_0x_{2t+1} \notin Z_R \\ \text{and if } x_{2\ell+1} = x_{2t+1} \text{ for some } \ell \text{ with } 1 \leq \ell \leq k-1 \text{ then } \ell \leq t.\}$$

(Instead of recording the value of ℓ for all vertices $x_{2\ell+1}$ such that $x_0x_{2\ell+1} \notin Z_R$, this ensures that *exactly one* value t is stored for each *distinct* vertex satisfying this condition.) The number of phases will be

$$p = |\mathcal{B}| = |\{x_{2t+1} \mid 1 \leq t \leq k-1, \quad x_0x_{2t+1} \notin Z_R\}|.$$

For the first phase, choose t minimum such that $t \in \mathcal{B}$, and proceed as before. In subsequent phases, if t was the starting point of the previous phase then choose $q > t$ minimum such that $q \in \mathcal{B}$. In the example of Figure 8 there is only one phase, since $\mathcal{B} = \{x_{11}\}$. The processing steps are shown in Figure 9.

With this adaptation, the process works exactly as described before. Removing or inserting one copy of a repeated edge will remove or insert them all, but we are always able to work through each phase and proceed to the next. (Note that by the end of Figure 9 it may appear at first glance that we have introduced odd chords that were not present before. But each of these is the edge x_0x_3 which belongs to $G' \cap S$ and therefore should be present after the 1-circuit S is switched.)

This completes the description of how to form the canonical path from G to G', by processing each circuit in the canonical decomposition (in order) as described above. Each switch performed is a proper move of the Markov chain since x_0 is distinct from all other vertices in the 1-circuit, and two consecutive edges in the circuit span three distinct vertices.

Note that reinstating the chords correctly is an important feature of the method, since 1-circuits can intersect arbitrarily. If we did not do this, we could not guarantee to process the decomposition "circuit by circuit" in the manner we have described.

2.2 Analysing the flow

In this section we obtain a bound on the mixing time under the assumption $d = d(n) \leq n/2$ for all n. This implies the general result for any $(d(n))$, by complementation where

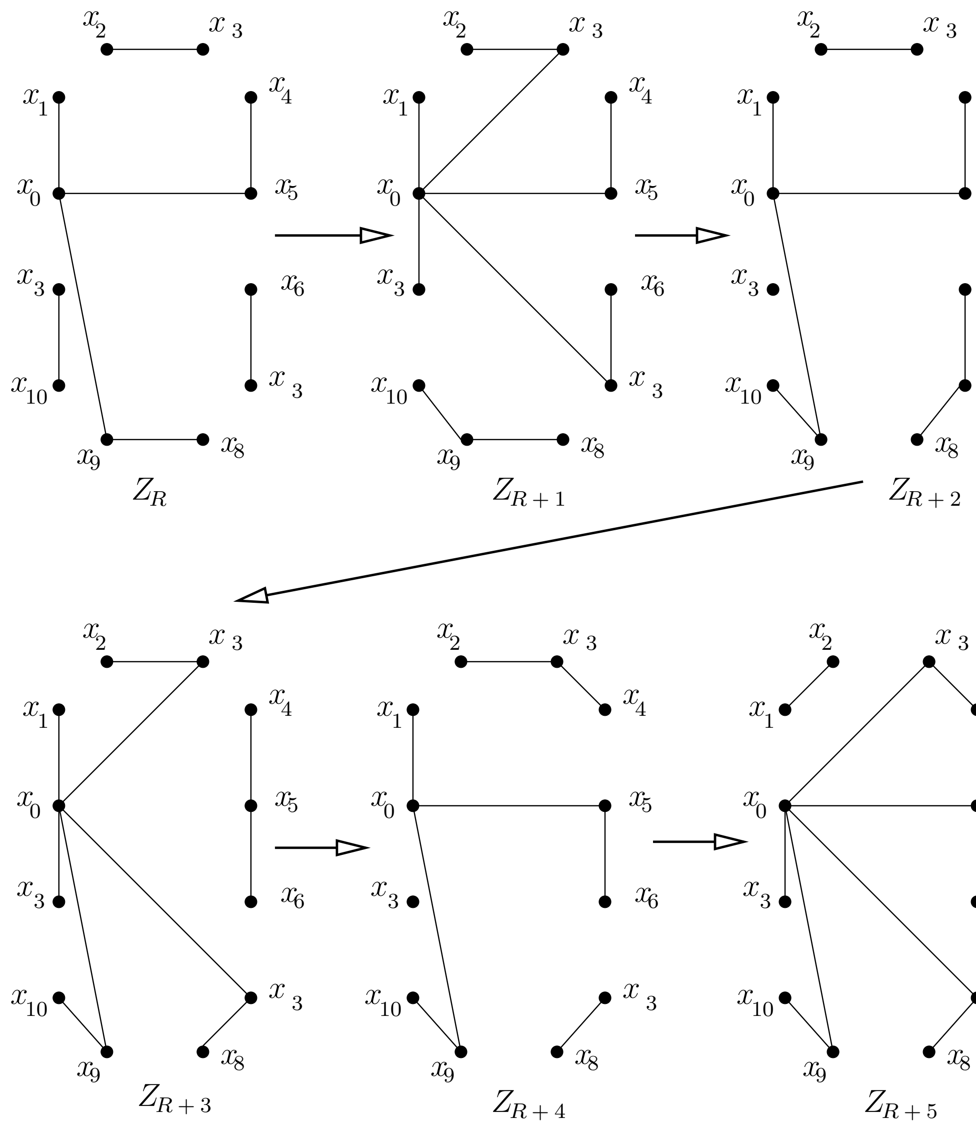

Figure 9: Processing a 1-circuit with repeated vertices

necessary.

Fix a pairing $\psi \in \Psi(G, G')$ and let Z be any graph on the corresponding canonical path from G to G'. Identify each graph with its symmetric $n \times n$ adjacency matrix. Define a symmetric $n \times n$ matrix L by $L + Z = G + G'$. Entries of L belong to $\{-1, 0, 1, 2\}$. Alternatively, we can think of L as K_n with edge labels given by the entries in L. An edge in L is called *bad* if its label is -1 or 2. Note that L is independent of ψ. We call L an *encoding* for Z (with respect to G, G'). Note that an edge receives label -1 if it is absent in both G and G' but it is present in Z, while an edge receives label 2 if it is present in both G and G' but it is not present in Z. Thus, edges in the symmetric difference $G \triangle G'$ never receive bad labels.

Let Z' be the next graph after Z in the canonical path from G to G'.

Lemma 1. *Given (Z, Z'), L and ψ, we can uniquely recover G and G'.*

Proof. From $Z + L$ we can find the matrix $G + G'$. Entries with value 2 in $Z + L$ are in both G, G', while those with value 0 are in neither. The symmetric difference graph H comes from those entries in $Z + L$ with entry 1. It remains to assign colours blue and red to the edges of H so that blue edges come from G and the red edges come from G'. From the uncoloured version of H we can find its circuit decomposition $\mathcal{C}$ using ψ. Next we show that, by observing the transition (Z, Z'), we can uniquely determine which circuit C_r is currently being processed. (Further, we will determine which 1-circuit or 2-circuit is currently being processed.)

Suppose that the transition removes the edges ab, cd and replaces them with edges ac, bd. Without loss of generality assume that a is the lowest-labelled vertex in $\{a, b, c, d\}$. Then a must be the start-vertex of the current circuit. Let $(C_k, \ldots, C_\ell)$ be the subsequence of $\mathcal{C}$ consisting of all circuits with a as start-vertex. Say that a 2-circuit is *standard* if it falls into case (a), and *nonstandard* otherwise. Extend this terminology to 1-circuits: a 1-circuit is *nonstandard* if it arises when processing a nonstandard 2-circuit, otherwise it is *standard*. (Hence a 1-circuit is nonstandard if and only if it contains a shortcut edge.) Now let $\mathcal{S}$ be the sequence of standard 1-circuits and nonstandard 2-circuits obtained by decomposing the circuits $(C_k, \ldots, C_\ell)$ in order, as described above. Since we have not decomposed the non-standard 1-circuits, the elements of $\mathcal{S}$ are pairwise edge-disjoint and the union is C_r.

Each transition in our canonical paths is of one of three types.

> **Type 1**: the transition is the 4-cycle switch involving the shortcut edge of a nonstandard 2-circuit (performed first in case (b) and last in case (c)). Here edges bd, ab, ac are consecutive edges of the 2-circuit and edge cd is the shortcut edge.
>
> **Type 2**: the transition is any step in the processing of a 1-circuit (standard or nonstandard) which does not involve the shortcut edge (in the case that the 1-circuit is nonstandard). Here edges bd, cd are consecutive edges of the 1-circuit.
>
> **Type 3**: the transition is a step in the processing of a nonstandard 1-circuit which does involve the shortcut edge. Here cd is an edge of the nonstandard 2-circuit from which the 1-circuit was formed, and edge bd is the shortcut edge of the 2-circuit.

First suppose that (Z, Z') is a Type 1 transition. So there exists a non-standard 2-circuit $S \in \mathcal{S}$ with consecutive edges $bd, ab, ac \in S$. Since distinct elements of $\mathcal{S}$ are edge-disjoint, this identifies S uniquely *unless* the edge cd belongs to another nonstandard 2-circuit $\widehat{S}$ for which the transition (Z, Z') is a Type 3 transition. (Here edge bd is the shortcut edge of $\widehat{S}$.) We must decide whether the transition (Z, Z') is a Type 1 transition for S or a Type 3 transition for $\widehat{S}$. We say that (Z, Z') *comes from* S if the former holds, and comes from $\widehat{S}$ if the latter holds. There are four cases.

- Suppose that $S < \widehat{S}$ in the ordering of $\mathcal{S}$, and S falls into case (b). The 2-circuit S has edges ab, au in Z. If (Z, Z') comes from S then edge au is present in Z, since the 4-cycle switch occurs before the processing of the 1-circuit. But if (Z, Z') comes from $\widehat{S}$ then au is absent in Z. This allows us to distinguish these two cases.

- Suppose that $S < \widehat{S}$ and S falls into case (c). Here $cd \notin G$, so it is absent from the current graph at the start of processing either S or $\widehat{S}$. If (Z, Z') comes from $\widehat{S}$ then some edges of $\widehat{S}$ have already been switched in Z (hence allowing us to determine whether the transition comes from S or $\widehat{S}$), *unless* the transition (Z, Z') is the *first* transition in the processing of $\widehat{S}$. If so then $\widehat{S}$ falls into case (c), which means that $bd \notin Z$. But then bd and cd are both absent in Z, which is a contradiction. So (Z, Z') cannot be the first transition of $\widehat{S}$ and we can always distinguish between S and $\widehat{S}$.

- Now suppose that $\widehat{S} < S$ and $\widehat{S}$ falls into case (b). If (Z, Z') comes from $\widehat{S}$ then not all of $\widehat{S}$ will have been switched in Z' (thereby enabling us to distinguish between S and $\widehat{S}$) *unless* the transition (Z, Z') is the *last* transition in the processing of $\widehat{S}$. But this can only occur if there is only one phase in the processing of $\widehat{S}$, and ab is an edge in the 1-circuit obtained from $\widehat{S}$. But then ab is an edge in $\widehat{S}$, contradicting the fact that elements of $\mathcal{S}$ are pairwise disjoint. Therefore we can always distinguish between S and $\widehat{S}$ in this case.

- Finally suppose that $\widehat{S} < S$ and that $\widehat{S}$ falls into case (c). Then the 4-cycle switch for $\widehat{S}$ switches the 4-cycle $aqbd$, where aq is the first edge of $\widehat{S}$. Since $\widehat{S}$ is in case (c) this 4-cycle switch is performed at the end of processing $\widehat{S}$, so $aq \in Z$ if the transition (Z, Z') comes from $\widehat{S}$. But $aq \notin Z$ if the transition (Z, Z') comes from S, so we can decide between the two.

Now suppose that (Z, Z') is not a Type 1 transition but it is a Type 2 transition. Then there exists $S \in \mathcal{S}$ with bd, $cd \in S$. Since all possible transition types involve at least one of bd, cd belonging to a given element of $\mathcal{S}$, this identifies S uniquely.

Finally suppose that (Z, Z') is not a Type 1 or a Type 2 transition. Then it is a Type 3 transition, so there exists a nonstandard 2-circuit $S \in \mathcal{S}$ with $cd \in S$. This identifies S uniquely, since elements of $\mathcal{S}$ are edge-disjoint.

In all cases we have uniquely identified the element $S \in \mathcal{S}$ which is being switched in the current transition. Suppose that S comes from the decomposition of the circuit C_r. We can now assign colours to the edges of all circuits C_i, $i \neq r$, to G or G'. For $i > r$, the edges in $C_i \cap Z$ belong to G and the remainder to G'. For $i < r$, the edges in $C_i \cap Z$ belong to G' and the remainder to G. Similarly, we can assign colours to the edges of every standard 1-circuit or nonstandard 2-circuit obtained in the decomposition of C_r, other than the current element $S \in \mathcal{S}$ being switched in the transition (Z, Z'). We now explain how to assign colours to the edges of S.

If (Z, Z') is a Type 1 transition then S is a nonstandard 2-circuit. Suppose that $S = abd \ldots uaz \ldots c$. Then the 1-circuit produced in the decomposition of S is $S' = au \ldots dc \ldots z$, since cd is the shortcut edge of S. The edge au is present in G. Therefore if au is present in Z then S falls into case (b) and the current transition is the first transition in the processing of S. Here the edges of $Z \cap S$ should all be coloured blue and the other edges of S should be coloured red. Otherwise au is absent in Z and S falls into case (c), so the current transition is the last transition in the processing of S. Here the edges of $Z' \cap S$ should be coloured red and the remaining edges of S should be coloured blue.

Now suppose that (Z, Z') is a Type 2 or Type 3 transition. First suppose that S is a standard 1-circuit. An edge of S is called *unswitched* if it agrees with G and *switched* if it agrees with G'. An unswitched section of S is separated from a switched section by a successive pair of edges of S such that both are in Z or neither is in Z. Thus we can decompose S into sections which are alternately switched and unswitched. Since the edge cd is switched in the transition (Z, Z'), we know that cd is unswitched in Z. This allows us to assign the sections of S as switched and unswitched. Finally, colour an edge of S blue if it belongs to Z and is unswitched, or if it does not belong to Z and is switched. All other edges of S should be coloured red.

If S is a nonstandard 2-circuit, let S' be the 1-circuit obtained by decomposing S. We adapt the above argument to the 1-circuit S'. Even without defining whether the shortcut edge is switched or unswitched, we can decompose S' into sections divided by successive pairs of edges which are either both in Z or both not in Z, as above. By considering all edges in each section other than the shortcut edge, we can label each section as switched or unswitched. (Each section must have at least two edges of S', by construction.) Hence we can assign colours to all edges of S' other than the shortcut edge, as above. There are three edges in S which are not in S', but only one way to assign colours red, blue to their edges so that S is alternating. This completes the proof. □

The following proof is similar to one in [17].

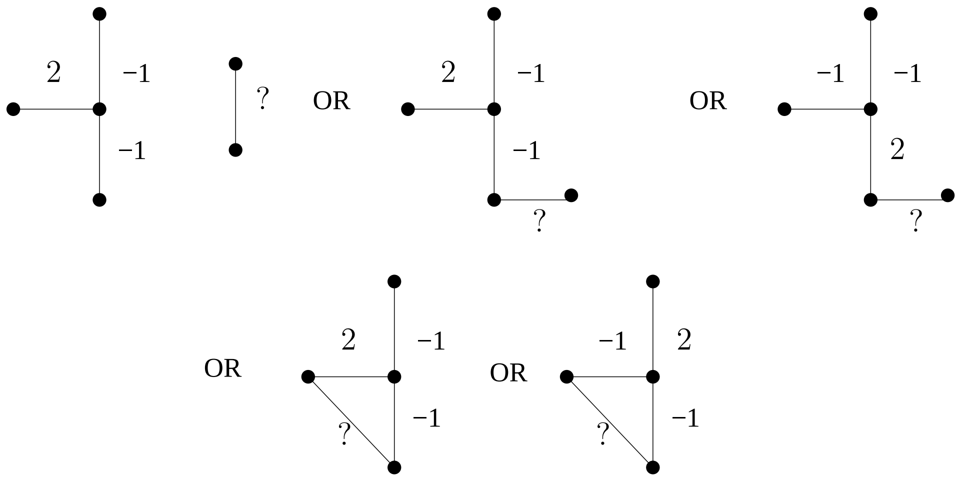

Figure 10: The five possible configurations of four bad edges

Lemma 2. *There are at most four bad edges in any encoding L. The labelled graph consisting of the bad edges in L must form a subgraph of one of the five possible labelled graphs shown in Figure 10, where "?" represents a label which may be either -1 or 2.*

Proof. At the start of processing any standard 1-circuit there are no bad edges. When processing a 2-circuit in case (b) or (c), the shortcut edge xy may become a bad edge during the processing of the 2-circuit. In case (b), if the shortcut edge belongs to G and to G' then it becomes a bad edge with $L(xy) = 2$ after the 4-cycle is switched. In case (c), if the shortcut edge does not belong to G or G' then it becomes a bad edge with $L(xy) = -1$ during the processing of the 1-circuit.

Now consider processing a 1-circuit. We use the notation from the previous section under the heading "Processing a 1-circuit". The first step in phase 1 of processing the 1-circuit can introduce at most two bad edges adjacent to x_0. In the first step of phase 1, the new edge x_0x_{2t+1} will receive the label -1 in L if it is absent in both G and G', while the deleted edge x_0x_{2t-1} will receive the label 2 in L if it is present in both G and G'. Edges of the standard 1-circuit currently being processed never receive bad labels, since they belong to the symmetric difference of G and G'. For a non-standard 1-circuit S, the only edge which may receive a bad label during the processing of S is the shortcut edge, as described above.

If x_0x_{2t+1} has label -1, this holds throughout the remainder of the first phase. However, if the label of x_0x_{2t-1} is 2, then the worst that can happen is that this bad label "shifts" down one at each step of the first phase (i.e. from x_0x_{2t-1} to x_0x_{2t-3}, to x_0x_{2t-5}, and so on). Hence during the first phase there are at most two bad labels adjacent to x_0. If there are two bad labels adjacent to x_0 at the penultimate step of the first phase, then the last step of the first phase completely removes one of these bad labels (since edge x_0x_1 belongs to the symmetric difference), while the label x_0x_{2t+1} may still be bad. If it is still bad, there is at least one more phase to come.

The first step in any subsequent phase may introduce at most two more bad labels adjacent to x_0, similarly to the first phase. During the remainder of the phase, one of these bad labels (a 2) is shifted towards another (a -1), while the third (a -1) remains where it is (it will enter the next phase). The final step of the phase ensures that there is at most one bad edge. Hence at most three bad edges adjacent to x_0 can arise from the processing of a 1-circuit. Therefore at most four bad edges arise from processing a 2-circuit. The set of all bad edges is a subset of these possible ones, hence the labelled graph of all bad edges must be a subgraph of those shown in Figure 10. (Note that the bad edge labelled "?" in Figure 10 arises as a bad shortcut edge, when processing a 2-circuit in cases (b) or (c). In order to obtain some of these configurations, one or both of the endvertices of the shortcut edge must appear more than once in the 2-circuit.) □

We can extend the notion of a switch to encodings. Let $L(ab)$ denote the label of edge ab in the encoding L. By definition, the sum of edge-labels around any vertex v in an encoding L adds up to d. If x, y, z, w are vertices with $L(xy) > -1$, $L(zw) > -1$, $L(xz) < 2$ and $L(yw) < 2$ then we may perform a switch by decreasing $L(xy)$ and $L(zw)$ by one and increasing $L(xz)$ and $L(yw)$ by one, to give a new encoding L'. (This is the analogue of deleting edges xy, zw and inserting edges xz, yw.)

A *valid* encoding is one which satisfies the statement of Lemma 2. This proof is adapted from Goldberg and Jerrum [10].

Lemma 3. *Given a valid encoding, one can obtain a graph (with no bad edges) using at most three switches.*

Proof. Let L be a valid encoding. By Lemma 2, we know that L has at most four bad edges, and that if there are four bad edges then three of them, with labels -1, -1, 2, are adjacent.

Suppose first that a vertex α is adjacent to two bad edges $\alpha\beta, \alpha\delta$ such that $L(\alpha\beta) = 2$, $L(\alpha\gamma) = -1$. Since the sum of the L-edge-labels around each vertex is d, it follows that there exists some vertex $\delta \notin \{\alpha, \beta, \gamma\}$ with $L(\delta\beta) = 0$ and $L(\delta\gamma) = 1$. Perform a switch on the cycle $\alpha\beta\delta\gamma$ to give $L'(\alpha\beta) = L'(\delta\beta) = 1$ and $L'(\alpha\gamma) = L'(\delta\gamma) = 0$. We have reduced the number of bad edges by two. Call this operation a $(-1, 2)$-*switch*.

Next suppose that there is a bad edge $\alpha\beta$ with $L(\alpha\beta) = 2$, but there is no vertex γ with $L(\alpha\gamma) = -1$. Choose γ such that $L(\alpha\gamma) = 0$. Such a vertex γ exists since the sum of the labels around α is d and $n \geq d + 1$. We can choose $\delta \notin \{\alpha, \beta, \gamma\}$ as before so that $L(\delta\beta) = 0$ and $L(\delta\gamma) = 1$. Perform the same switch as before to give $L'(\alpha\beta) = L'(\delta\beta) = L'(\alpha\gamma) = 1$ and $L'(\delta\gamma) = 0$. We have reduced the number of bad edges by one. Call this operation a 2-*switch*.

Finally suppose that there is a bad edge $\alpha\beta$ with $L(\alpha\beta) = -1$, but there is no γ with $L(\alpha\gamma) = 2$. Choose γ so that $L(\alpha\gamma) = 1$. Such a vertex γ exists since the sum of the labels around α is d. Choose $\delta \notin \{\alpha, \beta, \gamma\}$ such that $L(\delta\beta) = 1$ and $L(\delta\gamma) = 0$. Perform a switch on the cycle $\alpha\beta\delta\gamma$ to give $L'(\alpha\beta) = L'(\delta\beta) = L'(\alpha\gamma) = 0$ and $L'(\delta\gamma) = 1$. We have reduced the number of bad edges by one. Call this operation a -1-*switch*.

Hence we can reduce the number of bad edges from three to none using at most three switches. Moreover, if L has four bad edges then we can reduce the number of bad edges to two using a $(-1, 2)$-switch, and then to none using two more switches. This completes the proof. □

We say that an encoding L is *consistent with* a given graph Z if the matrix $L + Z$ only takes entries in the set $\{0, 1, 2\}$. Let $\mathcal{L}(Z)$ be the set of all valid encodings which are consistent with Z.

Lemma 4. *For any* $Z \in \Omega_{n,d}$ *we have*

$$|\mathcal{L}(Z)| \leq 2d^6 n^5 |\Omega_{n,d}|.$$

Proof. Fix $Z \in \Omega_{n,d}$ and let $L \in \mathcal{L}(Z)$ be an encoding which is consistent with Z. In at most three switches we can transition from L to some graph $A \in \Omega_{n,d}$ (with no bad edges), by Lemma 3. We could turn this into a function $\varphi : \mathcal{L}(Z) \to \Omega_{n,d}$ by performing these switches in a canonical way. (For example we could perform all $(-1, 2)$-switches, then all -1-switches, then all 2-switches, breaking ties by ascending order of the 4-tuple $(\alpha, \beta, \gamma, \delta) \in [n]^4$.) If $|\varphi^{-1}(A)| \leq 2d^6 n^5$ for all $A \in \Omega_{n,d}$ then the result follows.

Now fix $A \in \Omega_{n,d}$. Define a *reverse* $(-1, 2)$-*switch* to be the reverse of a $(-1, 2)$-switch, and similarly define *reverse* -1-*switches* and *reverse* 2-*switches*. For an upper bound we count all valid encodings in $\mathcal{L}(Z)$ which can be obtained from A using at most 3 reverse switches, regardless of whether A is the "canonical" image of that encoding under φ.

Let $N_{(-1,2)}$ be the number of 4-tuples of distinct vertices $(\alpha, \beta, \gamma, \delta)$ where a reverse $(-1, 2)$-switch may be performed in A to produce an encoding $L' \in \mathcal{L}(Z)$. Since L' is consistent with Z and $L'(\alpha\gamma) = -1$ we must have $Z(\alpha\gamma) = 1$. (Also $L'(\alpha\beta) = 2$ implies that $Z(\alpha\beta) = 0$, but we ignore this for an upper bound). Therefore there are n ways to choose α and there are d ways to choose γ such that $Z(\alpha\gamma) = 1$. From A we see that there are d ways to choose β such that $A(\alpha\beta) = 1$, and $d - 1$ ways to choose $\delta \neq \beta$ such that $A(\beta\gamma) = 1$. This 4-tuple of vertices is counted by $N_{(1,2)}$ if the vertices are all distinct. Therefore

$$N_{(-1,2)} \leq d^2(d-1)n.$$

Similarly we can prove that

$$N_{-1} \leq d^2 n(n-1) \text{ and } N_2 \leq d(d-1)^2 n,$$

where N_{-1} (respectively, N_2) is the number of 4-tuples of distinct vertices $(\alpha, \beta, \gamma, \delta)$ where a reverse -1-switch (respectively, a reverse 2-switch) may be performed in A to produce an encoding $L' \in \mathcal{L}(Z)$.

Next we can count the number of ways to perform two or three such reverse switchings, to produce valid encodings which are consistent with Z. There are several different cases corresponding to all possible labelled subgraphs of the graphs in Figure 10. The three cases with the most significant contribution are

(a) *three bad edges with labels -1, with two adjacent.* An upper bound for the number of ways to produce such an encoding which is consistent with Z is

$$\big(d^2 n(n-1)\big)^2 \times d^2(n-1) = d^6 n^2 (n-1)^3.$$

(b) *three bad edges, two which have labels -1 and are adjacent, and a non-adjacent bad edge with label 2.* An upper bound for the number of ways to produce such an encoding which is consistent with Z is

$$d^2 n(n-1) \times d^2(n-1) \times d(d-1)^2 n = d^5(d-1)^2 n^2 (n-1)^2.$$

(c) *four bad edges, three of which are adjacent with labels $-1, -1, 2$ and a fourth non-adjacent bad edge with label 2.* An upper bound for the number of ways to produce such an encoding which is consistent with Z is

$$d^2(d-1)n \times d^2(n-1) \times d(d-1)^2 n = d^5(d-1)^3 n^2 (n-1).$$

It is straightforward but tedious to check that the contributions from the other 29 terms are covered by increasing by overestimating these three terms as follows:

$$d^6 n^5 + d^7 n^4 + d^8 n^3.$$

Finally since $d \leq n/2$ we conclude that

$$|\varphi^{-1}(A)| \leq \tfrac{7}{4}\, d^6 n^5 \leq 2d^6 n^5,$$

completing the proof. □

For each pair (G, G') of distinct graphs in $\Omega_{n,d}$, let $\mathcal{P}_{G,G'}$ be the set of $|\Psi(G, G')|$ canonical paths which we have defined from G to G', one for each pairing $\psi \in \Psi(G, G')$. Let $\mathcal{P} = \cup_{G \neq G'} \mathcal{P}_{G,G'}$. Define

$$f(\gamma) = |\Omega_{n,d}|^{-2}\, |\Psi(G, G')|^{-1}$$

for each path $\gamma \in \mathcal{P}_{G,G'}$. Then

$$\sum_{\gamma \in \mathcal{P}_{G,G'}} f(\gamma) = |\Omega_{n,d}|^{-2} = \pi(G)\pi(G')$$

where π is the stationary probability of the Markov chain, which is uniform on $\Omega_{n,d}$. Thus $f : \mathcal{P} \to [0, \infty)$ is a flow. We want to apply (2). First we bound $f(e)$ for all transitions e of the Markov chain.

Lemma 5. *For any transition $e = (Z, Z')$ of the Markov chain,*

$$f(e) \leq 2\, d^{12}\, n^5 / |\Omega_{n,d}|.$$

Proof. Fix a transition $e = (Z, Z')$ of the Markov chain. Let $(G, G') \in \Omega_{n,d} \times \Omega_{n,d}$ and suppose that e lies on the canonical path $\gamma_\psi(G, G')$ from G to G' corresponding to the pairing $\psi \in \Psi(G, G')$. From e and (G, G', ψ) we can produce the encoding L and a yellow-green colouring of $H = G \triangle G' = Z \triangle L$, where green edges belong to Z and yellow edges belong to L. (Here the symmetric difference $Z \triangle L$ consists only of edges corresponding to edges with entry 1 in the matrix $Z + L$.)

The pairing ψ produces a circuit decomposition $\mathcal{C}$ of H, with colours alternating (yellow, green, yellow, green,...) around the circuit *except* possibly for the current circuit C_r which is being switched in the transition e. The current circuit may have *bad vertices*, where a vertex is *bad* with respect to ψ if there is a yellow edge paired with a yellow edge at v, or a green edge paired with a green edge at v. If a vertex is not bad it is *good*. A yellow-yellow or green-green pair at a bad vertex v is called a *bad pair*. Bad vertices are only found adjacent to odd chords which have just been switched during the processing of a 1-circuit, or adjacent to a shortcut edge if the 1-circuit is non-standard. In both cases, bad vertices are the endpoints of potentially bad edges, so we may use the results of Lemma 2. There are at most four bad edges, of which all but one must share a common endpoint, so there are at most six bad vertices. Indeed, there are at most six bad pairs in total.

Say that a pairing $\psi \in \Psi(H)$ is *consistent with* L if there are at most six bad pairs in the yellow-green colouring of H with respect to ψ. Let $\Psi'(L)$ be the set of all pairings ψ of H which are consistent with L. Given any (G, G') with $G \triangle G' = H$, any pairing $\psi \in \Psi(G, G')$ is consistent with the yellow-green colouring of H. Therefore each triple

(G, G', ψ) with $\psi \in \Psi(G, G')$ and $e \in \gamma_\psi(G, G')$ gives rise to at least one pair (L, ψ) where $L \in \mathcal{L}$ and $\psi \in \Psi'(L)$.

On the other hand, we can start with an encoding $L \in \mathcal{L}$ and find an upper bound for the number of pairings $\psi \in \Psi'(L)$. Once ψ is given, if $e \in \gamma_\psi(G, G')$ for some (G, G') then we can uniquely construct (G, G'), by Lemma 1. For an upper bound we count all pairings, even though for some ψ there may not be any (G, G') with $e \in \gamma_\psi(G, G')$.

Recall that

$$|\psi(H)| = \prod_{v \in V} \theta_v!$$

where $2\theta_v$ is the degree of v in H. (In $G \triangle G'$, the quantity θ_v is the blue-degree of v, which equals the red-degree of v.) Now we bound $|\psi'(L)|$. Each good vertex v contributes a factor $\theta_v!$ to $|\psi'(L)|$, but a bad vertex may contribute more. For example, consider a bad vertex v with one bad pair. Then there are

$$\frac{\theta_v(\theta_v + 1)}{2}\,(\theta_v - 1)! = \frac{\theta_v + 1}{2}\,\theta_v! \le \theta_v \cdot \theta_v!$$

ways to pair up the green and yellow edges around v giving one bad pair. If v has one green-green pair and one yellow-yellow pair, then there are

$$\theta_v! + \frac{{\theta_v}^2(\theta_v - 1)^2}{4}\,(\theta_v - 2)! = \left(1 + \frac{\theta_v(\theta_v - 1)}{4}\right)\theta_v! \le {\theta_v}^2 \cdot \theta_v!$$

ways to pair up the green and yellow edges around v giving either no bad pairs (the $\theta_v!$ term) or one bad yellow-yellow pair and one bad green-green pair (the second term). Careful consideration shows that these are the only possibilities. That is, a bad vertex has at most two bad pairs, and if it has two bad pairs then they have different colours. To see this, suppose that in the current graph Z there are up to four bad edges, labelled va, vb, vc, xy, with $L(va) = L(vc) = -1$, $L(vb) = 2$ and xy the shortcut edge. Since v is the start-vertex of the 1-circuit, v cannot equal any of the other vertices. Also y cannot be the endpoint of an odd chord. Clearly $a \ne b$ and $b \ne c$, else the edge vb is simultaneously present and absent in G. It is possible that $a = c$, that $x = a$ or $x = c$, or that $x = b$. If $a = c \ne x$ then a can be a bad vertex with two bad pairs (of opposite colour). If $x = a = c$ then a only ever has at most one bad pair. If $x = b$ in case (b) then xy and vb cannot both have bad label 2 simultaneously, as the switch which makes vb bad with label 2 will change xy from a bad edge with label 2 into an edge of the current graph. Next suppose that $x = a$ and $a \ne c$ in case (c). By the time xy becomes bad, the edge va is no longer bad. The case that $x = c$ and $a \ne c$ is identical. If $x = a$ (or $x = c$) in case (b) then x can be the endpoint of two bad edges, with opposite labels, but x will have *no bad pairs*. Similarly if $x = b$ in case (c).

Since there are at most six bad pairs and $\theta_v \le d$, we have

$$|\Psi'(L)| \le d^6\,|\Psi(H)|. \tag{6}$$

Write $\mathbf{1}(e \in \gamma_\psi(G, G'))$ for the indicator variable which is 1 if $e \in \gamma_\psi(G, G')$ and 0 otherwise, for graphs $G, G' \in \Omega_{n,d}$ and $\psi \in \Psi(G, G')$. Combining all the above, we can bound the total flow through e by

$$\begin{aligned}|\Omega_{n,d}|^2 f(e) &= \sum_{(G,G')} \sum_{\psi\in\Psi(G,G')} \mathbf{1}(e \in \gamma_\psi(G,G'))\, |\Psi(H)|^{-1}\\ &\leq \sum_{L\in\mathcal{L}(Z)} \sum_{\psi\in\Psi'(L)} |\Psi(H)|^{-1}\\ &\leq \sum_{L\in\mathcal{L}(Z)} d^6\\ &\leq 2\, d^{12}\, n^5\, |\Omega_{n,d}|.\end{aligned}$$

The second line follows from counting all pairings in $\Psi(G, G')$, whether e lies on γ_ψ or not, while the third line follows from (6) and the last line follows from Lemma 4. This gives the desired result. □

We can now present the proof of Theorem 1.

Proof of Theorem 1. For any transition $e = (Z, Z')$ we have

$$1/Q(e) = |\Omega_{n,d}|/P(Z, Z') = 6a_{n,d}\, |\Omega_{n,d}| \leq d^2\, n^2 |\Omega_{n,d}|.$$

Therefore

$$\rho(f) \leq 2d^{14}\, n^7. \tag{7}$$

Also

$$\ell(f) \leq dn/2, \tag{8}$$

since each transition along a canonical path replaces an edge of G by an edge of G'. Since π is uniform we have, using (1),

$$\log 1/\pi^* = \log |\Omega_{n,d}| \leq dn \log(dn).$$

Plugging all this into (2) and applying Lemma 5 gives the stated bound. □

3 A chain on regular graphs with random size

We consider a model of the SWAN process [4, 11] for peer-to-peer networking, which we view as a Markov chain on $\Omega = \cup_{n\geq d+1}\, \Omega_{n,d}$, for some even constant d. For simplicity, we will examine the case $d = 4$ in detail and discuss later how this generalises. (The production version of the SWAN network uses $d = 4$.)

The process described in [4, 11] is a way of maintaining a self-organised network which is a randomly constructed 4-regular graph. At any time, a new client may arrive and request to join the network. This happens by taking two non-adjacent edges of the network and "clothespinning" them, replacing them by four new edges which join the new client to each of the four endpoints of the two chosen edges. Similarly, clients may choose to leave at any time, at which point this clothespinning procedure must be (randomly) reversed. This is described in more detail below.

The clothespinning operation was used by Robinson and Wormald [23, Section 3] in their proof that almost all regular graphs are Hamiltonian. They showed that the probability space of random d-regular graphs on $2n+1$ vertices is asymptotically equivalent to the probability space obtained by choosing a random d-regular graph on $2n$ vertices and performing a random clothespinning operation. This enabled them to carry the result over to graphs with an odd number of vertices. The clothespinning operation was also used by Hu, Macdonald and McKay [12, Appendix B] for generating certain 2-in, 2-out digraphs.

We will assume that the inter-arrival times are independently exponentially distributed. While there are n clients in the system, the expected inter-arrival time is ν_n. Similarly, we assume that the residual service times for clients currently in the system are identically and independently exponentially distributed with mean μ_n. We will further assume that $n\nu_n/\mu_n$ (the ratio of the probability of a departure to the probability of an arrival) is a strictly increasing function of n. This means that a larger system does not *too strongly* encourage arrivals or discourage departures compared with a smaller system.

If the system currently contains n clients then the probability that the next event is an arrival (or *insertion*) is

$$p_n = \frac{1/\nu_n}{1/\nu_n + n/\mu_n} = \frac{\mu_n}{\mu_n + n\nu_n},$$

and $q_n = 1 - p_n$ is the probability that it is a departure (or *deletion*). Note that, by our assumption, p_n is a strictly decreasing function of n. Also, assuming $\nu_0 > 0$, we have $p_0 = 1$. Suppose $p(n) = p_n$ can be extended to a twice-differentiable function of $n > 0$. We assume that there is an N (not necessarily an integer) such that $p(N) = \frac{1}{2}$. Then $n\nu_n/\mu_n < 1$ for $n > N$. Note that if no such N exists, the network will grow indefinitely, and there is no equilibrium behaviour. If we write $p(n) = f(n/N)$, then our assumptions about $f(x)$ are that

$$f(0) = 1, \quad f(1) = 1/2, \quad f'(1) = -\alpha < 0, \quad |f''(x)| < c \text{ for all } x \geq 0, \tag{9}$$

for some positive constants α, c.

An example which fits this model is Poisson arrivals with constant rate $1/\nu$, and constant departure rate $1/\mu$. If there are n clients in the system, then $p_n = \mu/(\mu + n\nu)$. Then $N = \mu/\nu$, and

$$p_n = \frac{1}{1 + n/N} = f(n/N)$$

with $f(x) = 1/(1+x)$, which satisfies our conditions. This model for arrivals and departures was used by Pandurangan, Raghavan and Upfal [22] in an analysis of a different architecture for peer-to-peer networking. For this example (and using $N\mu = \nu$) the system size settles down to Poisson(N) as t tends to infinity. More precisely, the generating function $G(z,t)$ for the probability $P_n(t)$ that the system is size n at step t is given by

$$G(z,t) = e^{-(1-z)N(1-e^{-t/\mu})} \left(1 - (1-z)e^{-t/\mu}\right)^{k_0},$$

where k_0 is the initial size of the system (see eg. Feller [7]).

A second example has Poisson arrivals with rate $1/\nu$ and residual service times $(\mu - \nu n)$ for $0 \leq n \leq \mu/\nu$. Then $N = \mu/2\nu$ and $f(x) = 1 - x/2$. If the p_n were the transition probabilities of a simple random walk, this would give a symmetric binomial distribution for n. However the situation is rather more complicated for the algorithm we describe, as a service transition will not be allowed if this causes the resulting graph to be non-simple.

We now describe how this process can be viewed as a Markov chain on the state space $\Omega = \cup_{n\geq 5} \Omega_{n,4}$. Since $\Omega_{n,4}$ is the set of all 4-regular graphs with vertex set $\{1, \ldots, n\}$, our "conceptual" chain must do some relabelling to make sure that the vertex set of the graph always has this form. Of course, the process does not do this relabelling in practice. The Markov chain algorithm is shown in Figure 11. Note that we have added a holding probability 1/2 to ensure the applicability of (2).

```
From $G \in \Omega_{n,4}$ do
    with probability $\frac{1}{2}$ do nothing;
    otherwise
        with probability $p_n$ do
            choose two distinct non-adjacent edges u.a.r.,
            add vertex $n+1$ on these edges,
            choose $i \in \{1, \ldots, n+1\}$ u.a.r.,
            swap the labels of vertex $i$ and vertex $n+1$,
        else (with probability $q_n$) do
            choose $i \in \{1, \ldots, n\}$ u.a.r.,
            let $a, b, c, d$ be the neighbours of vertex $i$,
            choose a perfect matching $M$ of $\{a, b, c, d\}$, u.a.r,
            if $M \cap E(G) = \emptyset$ then
                swap the labels of vertex $i$ and vertex $n$,
                delete vertex $n$ and add $M$ to the edge set,
            else
                do nothing,
            end if;
        end;
    end;
end;
```

Figure 11: The Markov chain on Ω

Think of the sets $\Omega_{n,4}$ as "levels", with $\Omega_{n+1,4}$ being the level above $\Omega_{n,4}$. We want the stationary distribution to be uniform on labelled 4-regular graphs when conditioned on being at a certain level. So that the process has a chance of mixing, we must show that the process does not grow too far away from having the mean number of vertices in time polynomial in N.

For ease of notation, let us write Ω_n instead of $\Omega_{n,4}$. Let π be the stationary distribution of the Markov chain and define $\sigma_n = \pi(\Omega_n)$ for all $n \geq 5$. If the stationary distribution is to be uniform on the levels we must have

$$\pi(X) = \frac{\sigma_n}{|\Omega_n|} \text{ for all } X \in \Omega_n.$$

Recall from (3) that $a_n = a_{n,4}$ is the number of unordered pairs of non-adjacent edges in a 4-regular graph on n vertices, where $a_{n,4} = \binom{2n}{2} - n\binom{4}{2} = 2n^2 - 7n$. Suppose that $Y \in \Omega_{n+1}$ and that the Markov chain can move from X to Y in one step. Then

$$P(X, Y) = \frac{p_n}{a_n\,(n+1)}, \quad P(Y, X) = \frac{q_{n+1}}{3\,(n+1)},$$

except in the particular case where the set of four endvertices of the two chosen edges (to be clothespinned) all share a common neighbour. Suppose first that the endvertices of the two chosen edges have exactly one common neighbour labelled k (as in Figure 2 but with all edges solid, where vertex k is the central vertex). We make the transition from X to Y whether we choose $i = k$ or $i = n + 1$ for the relabelling, so $P(X, Y)$ is twice the above value. Similarly $P(Y, X)$ has twice the above value as we may choose to delete vertex k or $n + 1$. In general, if the four endvertices of the two chosen edges have exactly j neighbours in common, with $1 \leq j \leq 3$, then $P(X, Y)$ and $P(Y, X)$ are both increased by a factor of $j + 1$.

If the Markov chain has a stationary distribution π and is time-reversible then it satisfies the detailed balance conditions $\pi(X)P(X, Y) = \pi(Y)P(Y, X)$. That is,

$$\frac{\sigma_n\, p_n}{|\Omega_n|\, a_n} = \frac{\sigma_{n+1}\, q_{n+1}}{3\,|\Omega_{n+1}|} \tag{10}$$

(where the factor $j + 1$ has been cancelled from both sides in the special cases described above). From (1) we find that

$$\frac{|\Omega_{n+1}|}{|\Omega_n|} = \frac{a_n}{3}\Big(1 + O\Big(\frac{1}{n}\Big)\Big),$$

for $n \geq 5$. Therefore the ratio σ_{n+1}/σ_n satisfies

$$\frac{\sigma_{n+1}}{\sigma_n} = \frac{p_n}{q_{n+1}}\Big(1 + O\Big(\frac{1}{n}\Big)\Big). \tag{11}$$

We now prove two very useful results. It will be convenient to consider only levels n where $|N - n| \leq N^{3/4}$. Define

$$\Omega^* = \bigcup_{|N-n|\leq N^{3/4}} \Omega_n$$

and call this set the *centre* of the state space. Elements of Ω^* are called *central* states. We will show that states are exponentially unlikely to be outside the centre of the state space after polynomial time, at least after a "burn-in" period. We will also show that the equilibrium distribution of the size of the system n is asymptotically normal with mean N and variance $N/4\alpha$.

From now on, we write N rather than $\lfloor N \rfloor$, since we assume that N is large and hence the error in this approximation is small. Similarly we will write $f(N)$ rather than $\lfloor f(N) \rfloor$ for any function f which tends to infinity with N.

Lemma 6. *The following statements fail with probability exponentially small in N. If started from any level below the centre, the system reaches the centre of the state space in time $N^{5/4}\log^2 N$. Thereafter the system size does not leave the centre in any time polynomial in N.*

Proof. Let $N^- = N - N^{3/4}/\log N$. If $n \le N^-$, then Taylor's theorem implies that

$$\begin{aligned} p_n &\ge f(1 - N^{-1/4}/\log N) \\ &= \tfrac{1}{2} + \alpha/(N^{1/4}\log N) + O(N^{-1/2}\log^{-2} N) \\ &> \tfrac{1}{2}(1 + \alpha/(N^{1/4}\log N)) \end{aligned}$$

for large N. Denote the system size at time t by n_t, and assume for the worst case that $n_0 = 0$. Since level N^- lies in the centre of the state space it suffices to prove that with very high probability (i.e. with exponentially small failure probability) we have $n_t \ge N^-$ for some $t \le N^{5/4}\log^2 N$. (Proving that we reach level N^-, and not just any central level, will help us to prove the second statement of the lemma.)

Let $r = \frac{1}{2}(1 + \alpha/(N^{1/4}\log N))$ and consider the random walk on the set $\{k \in \mathbb{Z} \mid k \le N^-\}$, equipped with a loop at N^- but at no other point, and with probability r of moving right at any step (or of looping at N^-). Let m_t be the position of this random walk at time t and assume that $m_0 = 0$. Then (m_t) is stochastically dominated by (n_t) so long as $n_t < N^-$. Hence it suffices to show that with very high probability (in the same sense as above) there is some $t \le N^{5/4}\log^2 N$ such that $m_t \ge N^-$.

Let X_t be the number of steps right made by the walk up to time t. If $X_t \ge (t+N^-)/2$ then $m_t \ge N^-$. Set $t = N^{5/4}\log^2 N$. Then

$$\mathbf{E}X_t = \tfrac{1}{2}N^{5/4}\log^2 N + \alpha N \log N$$

and

$$\frac{t + N^-}{2} = (1-\varepsilon)\mathbf{E}X_t$$

where

$$\varepsilon = \frac{2\alpha N\log N + N^{3/4}/\log N - N}{N^{5/4}\log^2 N + 2\alpha N \log N} = \frac{2\alpha(1+o(1))}{N^{1/4}\log N}.$$

Therefore using Chernoff–Hoeffding,

$$\mathrm{Prob}(X_t < t/2 + N^-) = \mathrm{Prob}(X_t < (1-\varepsilon)\mathbf{E}X_t) \le (1+o(1))\exp\left(-8\alpha^2 N^{3/4}\right),$$

which is exponentially small. This proves the first statement of the lemma.

Now assume that the system has size at least N^- at some time. We wish to show that at any later time, if the system size drops into the range $\{N - N^{3/4}, \ldots, N^- - 1\}$ from above, it is exponentially unlikely to exit that range from below (i.e. to leave the centre) and is extremely likely to return to size at least N^-. By restarting the clock, assume that $n_0 = N^- - 1$. Let $a = N^{3/4} - N^{3/4}/\log N$ and consider the random walk ("gambler's ruin") on the set $\{0, 1, \ldots, a\}$, where 0 and a are absorbing states and from all states other than 0, the probability of going right is r (defined above). Let Y_t be the state of this random walk after t steps, where $Y_0 = a - 1$. Arguing as above shows that the random process $(n_t - N + N^{3/4})$ stochastically dominates (Y_t). We wish to show that the probability ξ of "ruin" (i.e. that the process ends with absorbtion at zero, rather than absorbtion at a) is exponentially small. It is a standard result (see for example Feller [7, p. 344]) that

$$\xi = \frac{r/(1-r) - 1}{(r/(1-r))^a - 1}.$$

Now $\log((1+x)/(1-x) \geq 2x$ for all x satisfying $0 < x < 1$, and $r/(1-r) = (1+\varepsilon)/(1-\varepsilon)$ where $\varepsilon = \alpha/(N^{1/4}\log N)$. Hence

$$(r/(1-r))^a = \exp\left(a \log \frac{1+\varepsilon}{1-\varepsilon}\right) \geq \exp(2a\varepsilon).$$

Since $\varepsilon = o(1)$ and $a\varepsilon = (1+o(1))\,\alpha N^{1/2}/\log N \to \infty$, we have

$$\xi \leq (1+o(1)) \exp(-2a\varepsilon) = (1+o(1)) \exp\left(-2\alpha N^{1/2}/\log N\right).$$

Now let $N^+ = N + N^{3/4}/\log N$. Using similar arguments, one can show that whenever the system size enters the range $\{N^+ + 1, \ldots, N + N^{3/4}\}$ from below, it is exponentially unlikely to exit this range at the top (i.e. to leave the centre): rather, it is highly likely to return to size at most N^+. The "ruin probability" in this case also has the form $(1+o(1)) \exp\left(-2\alpha N^{1/2}/\log N\right)$. Finally, in T steps starting from N^-, say, there are only $O(T)$ times that the system size drops from N^- to $N^- - 1$, or rises from N^+ to $N^+ + 1$. Therefore the probability that the system exits the centre in T steps is at most

$$O(T) \exp(-2\alpha N^{1/2}/\log N)).$$

This probability is exponentially small whenever T is polynomially bounded, as required. □

Lemma 7. *The equilibrium distribution of the size n of the system is asymptotic to a normal distribution with mean N and variance $N/4\alpha$, for all n with $|N - n| \leq N^{3/4}$.*

Proof. Suppose that k is an integer with $|k| \leq N^{3/4}$. From (11) we have

$$\frac{\sigma_{N+k}}{\sigma_N} = \frac{q_N}{q_{N+k}} \cdot \prod_{j=0}^{k-1} \frac{p_{N+j}}{q_{N+j}} \cdot \prod_{j=0}^{k-1} \left(1 + \frac{O(1)}{N+j}\right).$$

Write $x = 1 + h$ where $h = j/N$ and apply Taylors theorem to obtain

$$f(1+h) = \frac{1}{2} - \alpha h + \frac{h^2}{2} f''(1+\theta h),$$

where $f''(1+\theta h)$ is uniformly bounded for $\theta \in [0,1]$. Let $c(j) = f''(1+\theta j/N)$. Then

$$\frac{p_{N+j}}{q_{N+j}} = \frac{1 - 2\alpha j/N + j^2 c(j)/N^2}{1 + 2\alpha j/N - j^2 c(j)/N^2}$$

and since $j = o(N)$,

$$\frac{q_N}{q_{N+k}} \prod_{j=0}^{k-1} \left(1 + \frac{O(1)}{N+j}\right) = 1 + O\left(\frac{k}{N}\right).$$

Moreover, for $|z| < 1$,

$$\log \frac{1-z}{1+z} = -2\left(z + z^3/3 + \cdots + z^{2j+1}/(2j+1) + \cdots\right).$$

Thus we have

$$\begin{aligned} \prod_{j=0}^{k-1} \frac{p_{N+j}}{q_{N+j}} &= \exp - \sum_{j=0}^{k-1} \left(\frac{4\alpha j}{N} - \frac{j^2 c(j)}{N^2}(1 + O(j/N)) \right) \\ &= \exp\left(-\frac{2\alpha k^2}{N}(1 + O(k/N)) \right). \end{aligned}$$

So for any integer k with $|k| \le N^{3/4}$,

$$\sigma_{N+k} = \sigma_N \left(1 + O\left(\frac{k}{N}\right)\right) \exp\left(-\frac{2\alpha k^2}{N}\right).$$

Hence as $N \to \infty$, the distribution on the central levels approaches a normal distribution with mean N and variance $N/4\alpha$, as claimed. □

Now we will analyse the mixing rate of our Markov chain using a multicommodity flow argument which builds upon the flow we defined on each level $\Omega_{n,d}$ in Section 2. Before beginning the analysis, we remark that other approaches to analysing the mixing time are possible, including a Markov chain decomposition approach as in [16, 20]. In these results, the spectral gap (or log-Sobolev constant) of a Markov chain is related to the spectral gap (or log-Sobolev constant) of simpler "projection" and "restriction" Markov chains. In our case the restriction Markov chain would be related to the swap chain of Section 2. But this would take some work to push through, and we have found it convenient instead to work directly with a multicommodity flow argument.

We will analyse the mixing time of our Markov chain *conditioned on* its trajectory remaining within the central states. By Lemma 6, the difference between the conditional and unconditional probabilities is an exponentially small relative error, which we ignore. Hence we will write σ_n, π for the conditional probabilities as well, by a slight abuse of notation. The analysis of the Markov chain proceeds by defining a flow between all pairs of states in Ω^*.

3.1 Defining the flow

Note that we can simulate a switch move within level n, using an insertion immediately followed by a deletion. The transition which inserts vertex $n+1$ on edges ab, cd is used to simulate a switch move. When we delete we may replace the original edges ab, cd or instead, one of the pairs ac, bd or ad, bc. Refer to an insertion followed by a deletion (simulating a switch move) as a *horizontal move*. By simulating switch moves in this way we can simulate every canonical path used in the "switch flow" defined in Section 2.1. The maximum load of the simulated flow is at most twice the maximum load of the switch flow.

For each $X \in \Omega_n$ where $n \neq N$ is a central level, we will first define a set F_X of paths from X to a set of graphs $S_X \subseteq \Omega_N$. Each such path p will correspond to a sequence of "branching numbers" $(b_p^{(1)}, \ldots, b_p^{(|N-n|)})$ which satisfy

$$\sum_{p \in F_X} \left(\prod_{i=1}^{|N-n|} b_p^{(i)} \right)^{-1} = 1. \tag{12}$$

Now let $X \in \Omega_n$, $Y \in \Omega_m$ be elements of Ω^*. We define a family $\mathcal{P}_{X,Y}$ of simple paths, which we call *flow-bearing*, from X to Y implicitly, as follows:

- for each path p from X to $t_p \in S_X$,
- for each path q from Y to $t_q \in S_Y$,
- for each $\psi \in \Psi(t_p, t_q)$ as defined in Section 2.1,
- use horizontal moves (defined above) to simulate the canonical path from t_p to t_q corresponding to the pairing ψ.

This gives a path $\gamma = \gamma(p, q, \psi) \in \mathcal{P}_{X,Y}$ obtained by following p, then the canonical path from t_p to t_q corresponding to ψ, and then by reversing the path q.

The flow $f(\gamma)$ for a path $\gamma = \gamma(p, q, \psi) \in \mathcal{P}_{X,Y}$ is defined by

$$f(\gamma) = \frac{\pi_X \, \pi_Y}{|\Psi(t_p, t_q)| \left(\prod_{i=1}^{|N-n|} b_p^{(i)} \right) \left(\prod_{j=1}^{|N-m|} b_q^{(j)} \right)}.$$

Assuming (12), we see that

$$\begin{aligned}
&\sum_{\gamma \in \mathcal{P}_{X,Y}} f(\gamma) \\
&= \sum_{\gamma \in \mathcal{P}_{X,Y}} \frac{\pi_X \, \pi_Y}{|\Psi(t_p, t_q)| \left(\prod_{i=1}^{|N-n|} b_p^{(i)} \right) \left(\prod_{j=1}^{|N-m|} b_q^{(j)} \right)} \\
&= \pi_X \pi_Y \sum_{p \in F_X} \left(\prod_{i=1}^{|N-n|} b_p^{(i)} \right)^{-1} \sum_{q \in F_Y} \left(\prod_{j=1}^{|N-m|} b_q^{(j)} \right)^{-1} \sum_{\psi \in \Psi(t_p, t_q)} |\Psi(t_p, t_q)|^{-1} \\
&= \pi_X \pi_Y.
\end{aligned}$$

This shows that f is a flow, as required.

It remains to show how to define the set F_X of flow-bearing paths from X to some set S_X of graphs in level N, for each $X \in \Omega^*$, and to define the sequences of branching numbers for each path in F_X such that (12) holds.

Suppose first that $X \in \Omega_n$ where $n < N$. There are a_n neighbours of X in Ω_{n+1} which are obtained by a transition of the Markov chain involving a null relabelling: that is, the vertex label i chosen is $i = n+1$, so the "relabelling" has no effect. The set of flow-bearing paths from X up to level N is the set of all paths

$$X = Z_0, Z_1, \ldots, Z_{N-n}$$

where Z_1 is one of these neighbours of X in Ω_{n+1}, and $Z_1, \ldots, Z_{N-n}$ is one of the flow-bearing paths from Z_1 up to level N. Let p be one such path. Then the sequence $(b_p^{(1)}, \ldots, b_p^{(N-n)})$ is defined inductively starting with $b_p^{(1)} = a_n$ which equals the number of possible choices for Z_1 given Z_0. Similarly, $b_p^{(2)}$ will equal the number of possible choices for Z_2, given (Z_0 and) Z_1. We may think of this as distributing 1 unit of charge from Z_0 equally among the $b_p^{(1)}$ possibilities for Z_1, so that each partial path inherits a $1/b_p^{(1)}$ fraction of the charge at Z_1. Performing this equal distribution at each step leaves the path p from X to S_X with the fraction

$$\left(\prod_{i=1}^{N-n} b_p^{(i)} \right)^{-1}$$

of the unit charge out of X to Ω_N. Summing these fractions over all paths $p \in F_X$ clearly gives 1, which is the total charge out of X. This proves that (12) holds in this case.

Next suppose that $X \in \Omega_n$ where $n > N$. We wish to make a similar inductive definition of a set F_X of flow-bearing paths from X down to level N. Let d_X be the number of neighbours of X in Ω_{n-1} which are obtained by a transition of the Markov chain involving a null relabelling (this means that vertex n is deleted). If d_X is positive then $1 \leq d_X \leq 3$ and the set of canonical paths from X down to level N is the set of all paths

$$X = Z_0, Z_1, \ldots, Z_{n-N}$$

where Z_1 is one of these d_X neighbours of X in Ω_{n-1} and $Z_1, \ldots, Z_T$ is one of the canonical paths from Z_1 down to level N. If p is one of these paths then set $b_p^{(1)} = d_X$ and define the rest of the sequence $(b_p^{(1)}, \ldots, b_p^{(n-N)})$ inductively.

Unfortunately, some graphs $X \in \Omega_n$ can have $d_X = 0$. This means that there is no matching of the neighbours of vertex n which avoids all edges of X. In fact, we can construct arbitrarily large 4-regular graphs where no vertex deletion is possible. (For example, take a square grid on a torus and replace each vertex by a copy of the complete graph K_4.) Call $X \in \Omega_n$ *bad* if $d_X = 0$, and good otherwise. If X is bad, we start by using horizontal moves to define a path from X to each good state $W \in \Omega_n$. Then the set of canonical paths from X down to level N is the set of all paths

$$X = Z_0, Z_1, \ldots, Z_R, \ldots, Z_T$$

where $Z_R = W \in \Omega_n$ is a good state, the path $X = Z_0, Z_1, \ldots, Z_R = W$ is a canonical path from X to W (defined in Section 2) simulated using horizontal moves on levels n and $n+1$, and $Z_R, \ldots, Z_T$ is one of the flow-bearing paths from Z_R down to level N. Note that $R \leq 4n$ using (8), recalling that it takes two horizontal moves to simulate one switch move.

Let $p = Z_0, \ldots, Z_R, Z_{R+1}, \ldots, Z_T$ be any such path, where $W = Z_R$ is a good state in Ω_n as above. Define

$$b_p^{(1)} = d_W \, g_n$$

where g_n is the number of good states in Ω_n. Then define the rest of the sequence $(b_p^{(1)}, \ldots, b_p^{(n-N)})$ inductively.

The same argument as above shows that (12) holds in these cases also.

3.2 Analysing the flow

We wish to apply (2). To do this we bound the parameter $\rho(f)$. Let $e = (Z, Z')$ be a transition from Ω_n to Ω_{n+1}. Note that

$$Q(e) \geq \frac{\sigma_n \, p_n}{|\Omega_n| \, a_n \, (n+1)}.$$

First suppose that $X \in \Omega_n$ where $N - N^{3/4} < n < N$. The amount of flow which originates at X is $\pi(X) = \sigma_n/|\Omega_n|$, the stationary probability of X. We distribute the flow evenly between the a_n neighbours of X in Ω_{n+1}. Hence a neighbour $Y \in \Omega_{n+1}$ of X receives $\sigma_n/(a_n|\Omega_n|)$ flow from X, and the total amount of flow it receives from its neighbours in Ω_n is at most

$$\frac{3\,\sigma_n}{a_n|\Omega_n|} = \frac{\sigma_{n+1} q_{n+1}}{p_n|\Omega_{n+1}|} < \frac{\sigma_{n+1}}{|\Omega_{n+1}|}. \tag{13}$$

This follows from detailed balance (see (10)) and

$$p_n + p_{n+1} = f(n/N) + f((n+1)/N) > 1$$

for $n < N$, since f is decreasing and $f(1) = 1/2$. So the total flow through $X \in \Omega_n$ originating in level $n-1$ is at most $\sigma_n/|\Omega_n|$. Similarly, the total flow through X originating in level $n-2$ is at most

$$\frac{9\,\sigma_{n-2}}{a_{n-1}a_{n-2}|\Omega_{n-2}|} \leq \frac{3\,\sigma_{n-1}}{a_{n-1}|\Omega_{n-1}|} \leq \frac{\sigma_n}{|\Omega_n|}$$

using (13) twice. Iterating this argument shows that the total flow through X originating in level $n-j$ is at most $\sigma_n/|\Omega_n|$ for all $j \geq 1$ such that $n-j$ is a central level. Therefore, the total flow through X originating in all central states below level n is at most $N^{3/4}\sigma_n/|\Omega_n|$.

This flow is divided evenly between the a_n neighbours of X in Ω_{n+1} resulting from a null relabelling. Therefore the loading on any flow-bearing edge e from Ω_n to Ω_{n+1} is at most

$$\frac{f(e)}{Q(e)} \leq \frac{1}{a_n} \cdot \frac{N^{3/4}\sigma_n}{|\Omega_n|} \cdot \frac{|\Omega_n|\, a_n\, (n+1)}{\sigma_n\, p_n} < 2N^{7/4}, \tag{14}$$

since $n \sim N$ and $p_n \sim \frac{1}{2}$ for central states. Moreover, the total flow arriving at a state X in level N originating at central states below level n is at most $N^{3/4}\sigma_N/|\Omega_N|$.

Next, consider states $X \in \Omega_m$ where $N < m < N+N^{3/4}$. We route $\pi(X) = \sigma_m/|\Omega_m|$ out of X and send it down to level N. Recall that the flow from each bad state is distributed evenly among each good state, and then sent down to level N. Equivalently, we can throw away the bad states and increase the flow out of each good state $X \in \Omega_m$ by a factor of

$$1 + \frac{h_m}{g_m} = \frac{|\Omega_m|}{g_m},$$

where h_m is the number of bad states in Ω_m and as above, g_m is the number of good states in Ω_m. (Of course we also have to increase the load on transitions between levels m and $m+1$ to account for the horizontal moves used on the paths from the bad to good states in Ω_m. We return to this point below.)

We claim that

$$h_m/|\Omega_m| = O(m^{-3}).$$

For suppose that $X \in \Omega_m$ is bad. Then for each of the three perfect matchings of the neighbours of m, at least one of the two edges must be present. There are 2^3 ways to choose one edge from each perfect matching, and the probability that the three given edges are present in a uniformly chosen element of Ω_m is asymptotically $O(m^{-3})$.

It follows that

$$\frac{|\Omega_m|}{g_m} = 1 + O(m^{-3}).$$

So now we can forget about the bad states, and just route $\pi(X)\,(1 + O(m^{-3}))$ units of flow from each good $X \in \Omega_m$ down to level N. First we distribute the flow equally among the neighbours of X in level $m-1$. Since X is good, the number of such neighbours is d_X where $1 \leq d_X \leq 3$. Moreover, most (good) states in Ω_m have exactly 3 neighbours in Ω_{m-1}, as we now show. For $W \in \Omega_{m-1}$ let $\mathcal{U}(W)$ be the set of neighbours of W in level m. Then $|\mathcal{U}(W)| = a_{m-1}$ for each $W \in \Omega_{m-1}$. Now suppose that $X \in \mathcal{U}_W$ and $d_X \in \{1, 2\}$. Then X is obtained from W by performing a clothespinning operation using two non-adjacent edges of W which are at distance 1 apart in W. There are most $18m$ such pairs of non-adjacent edges in W. Since $a_{m-1} = 2(m-1)^2 - 7(m-1) \sim 2m^2$ we obtain

$$\sum_{X \in \mathcal{U}(W)} \frac{1}{d_X} = \frac{a_{m-1}}{3}(1 + O(1/m)) = \frac{a_{m-1}}{3}(1 + O(1/N)).$$

Hence the total amount of flow received by a vertex $W \in \Omega_{m-1}$ from level m is

$$\begin{aligned}(1+O(m^{-3}))\,\frac{\sigma_m}{|\Omega_m|}\sum_{X\in\mathcal{U}(W)}\frac{1}{d_X} &= (1+O(1/N))\,\frac{a_{m-1}\sigma_m}{3|\Omega_m|}\\ &= (1+O(1/N))\,\frac{\sigma_{m-1}p_{m-1}}{q_m|\Omega_{m-1}|}\\ &< (1+O(1/N))\,\frac{\sigma_{m-1}}{|\Omega_{m-1}|}.\end{aligned}$$

This uses the fact that

$$p_{m-1}+p_m = f((m-1)/N) + f(m/N) < 1$$

since f is decreasing and $f(1) = 1/2$. Again, iterating this argument shows that the total amount of flow received by a vertex $W \in \Omega_m$ from a central level $m+j$ is at most

$$(1+O(1/N))^j\,\frac{\sigma_m}{|\Omega_m|}.$$

It follows that, for sufficiently large N, the total amount of flow into a central state at level m originating from central states above level m is at most

$$N^{3/4}\frac{\sigma_m}{|\Omega_m|}\,(1+O(1/N))^{N^{3/4}} \le 2N^{3/4}\frac{\sigma_m}{|\Omega_m|}.$$

The load on any edge e between central levels m and $m-1$ has two parts. There is the load from the flow-bearing paths being routed down from central states above level m. We can now show, similarly to (14), that this contribution is at most $2N^{7/4}$. But we also have a load from the horizontal moves used to route the flow from all the bad states on level $m-1$ to the good states. Certainly this can be achieved by scaling the flow for the switch chain by a factor of σ_{m-1}. There is a further factor of 2 for simulating the switch moves by horizontal moves. Let ρ_{m-1} be the maximum load of transitions in the switch flow on Ω_{m-1}. From (7) we have $\rho_{m-1} \le 2^{29}(m-1)^7$, so the second contribution to the load on this edge is

$$2^{30}\sigma_{m-1}(m-1)^7.$$

That is,

$$\rho(e) \le 2N^{7/4} + 2^{30}N^7(1+O(N^{-1/4})).$$

It remains to calculate the load on the transitions between levels N and $N+1$ which are used to perform the horizontal moves linking the sets S_X and S_Y, for all central states X, Y. By the above argument, the total amount of flow arriving at each state $X \in \Omega_N$ from central levels below level N is at most $N^{3/4}\sigma_N/|\Omega_N|$, and the total amount of flow arriving at each state $Y \in \Omega_N$ from central levels above level N is at most $2N^{3/4}\sigma_N/|\Omega_N|$. Hence it suffices to route a total flow of at most $3N^{3/4}\sigma_N/|\Omega_N|$ from each state $X \in \Omega_N$ equally to all states $Y \in \Omega_N$, using the switch flow. Simulating the switch flow using horizontal moves increases the maximum load by a factor of at most 2. Let ρ_N denote the maximum load for the flow defined in Section 2. From (7) we have $\rho_N \le 2^{29}N^7$. In the new flow, the maximum loading on any edge between levels

N and $N+1$ is simply scaled up by a factor of at most

$$6N^{3/4}\sigma_N < N^{3/4}.$$

Therefore we have

$$\rho(f) \le 2^{29} N^{31/4}$$

for this chain, since the transitions which carry the most load for the new flow are those between levels N and $N+1$, by construction.

For large enough N, flow-bearing paths have length at most

$$\ell(f) \le 2N^{3/4} + 4N + \sum_{m=N+1}^{N+N^{3/4}} 4m \le 5N^{7/4},$$

using (8), and $\log(1/\pi^*) \le 4N \log N$. Combining all this gives

$$\tau(\varepsilon) \le 5\ 2^{29}\ N^{19/2} \big(4N \log N + \log(\varepsilon^{-1})\big)$$

using (2). We must, of course add the $O(N^{5/4}/\log^2 N)$ "burn-in" time resulting from Lemma 6 to the overall mixing time, but this term can be absorbed into our upper bound on $\ell(f)$. As before, the bound we obtain on the mixing time is almost certainly several orders of magnitude too large.

3.3 Higher degrees

A a similar analysis can be performed for the corresponding process with larger even degree. For $2r$-regular graph, on arrival of a new vertex r non-adjacent edges are randomly chosen, and clothespinned. The number of sets of r non-adjacent edges in a $2r$-regular graph on n vertices satisfies

$$\binom{nr}{r} - n\binom{2r}{2}\binom{nr-2}{r-2} \le a_{n,2r} \le \binom{nr}{r}, \quad \text{so } a_{n,2r} \sim \frac{(nr)^r}{r!}.$$

When releasing a vertex there are $(2r)!/(2^r\, r!)$ ways to wire up the neighbours. But (1) shows that

$$\frac{|\Omega_{n+1,2r}|}{|\Omega_{n,2r}|} \sim \frac{(2nr)^r}{(2r)!} \sim \frac{a_{n,r}\, r!\, 2^r}{(2r)!}.$$

Hence detailed balance holds with σ_n satisfying (11), and the analysis can proceed as before. A move of the "switch" Markov chain from Section 2 can still be simulated with a factor at most 2 increase in the flow. Hence we can define the flow as before, and bound the mixing time similarly. The details are omitted.

4 Acknowledgements

We are very grateful to Fred Holt for bringing [4] to our attention. We especially thank Leslie Goldberg and Mark Jerrum for making their unpublished notes available to us.

These notes provided some of the central ideas used in the proof of the main result of Section 2.

References

[1] E. Bender and E. Canfield, The asymptotic number of labeled graphs with given degree sequences, *Journal of Combinatorial Theory (Series A)* **24** (1978), 296–307.

[2] I. Bezáková, N. Bhatnagar and E. Vigoda, Sampling binary contingency tables with a greedy start (Preprint, 2005).

[3] B. Bollobás, A probabilistic proof of an asymptotic formula for the number of labelled regular graphs, *European Journal of Combinatorics* **1** (1980), 311–316.

[4] V. Bourassa and F. Holt, SWAN: Small-world wide area networks, in *Proceedings of International Conference on Advances in Infrastructure (SSGRR 2003w)*, L'Aquila, Italy, 2003, paper # 64.
Available at `http://www.panthesis.com/content/swan_white_paper.pdf`

[5] G. Chartrand and L. Lesniak, *Graphs and Digraphs* (3rd ed.), Chapman and Hall, London, 1996.

[6] C. Cooper, R. Klasing, T. Radzik, A randomized algorithm for the joining protocol in dynamic distributed networks, *Theoretical Computer Science* (to appear). Also available as INRIA research report RR-5376 (CNRS report I3S/RR-2004-39-FR).

[7] W. Feller. *An Introduction to Probability Theory and Its Applications*, Volume I, 3rd Edition (1968), Wiley (New York).

[8] A. Frieze, On random regular graphs with non-constant degree, Research report 88-2, Department of Mathematics, Carnegie-Mellon University, 1988.

[9] C. Gkantsidis, M. Mihail and A. Saberi, *On the Random Walk method in Peer-to-Peer Networks*, INFOCOM 04, (2004).

[10] L. Goldberg and M. Jerrum, Private communication, 2004.

[11] F.B. Holt, V. Bourassa, A.M. Bosnjakovic, J. Popovic, SWAN – Highly reliable and efficient networks of true peers, in *Handbook on Theoretical and Algorithmic Aspects of Sensor, Ad Hoc Wireless, and Peer-to-Peer Networks* (J. Wu, ed.), CRC Press, Boca Raton, Florida, 2005, pp. 799–824.

[12] J. Hu, A.H. Macdonald and B.D. McKay, Correlations in two-dimensional vortex liquids, *Physical Review B* **49** (1994), 15263–15270.

[13] M. Jerrum, A. Sinclair and B. McKay, When is a graphical sequence stable?, in *Random Graphs, Vol. 2 (Poznań, 1989)*, Wiley, New York (1992), pp. 101–115.

[14] M. Jerrum and A. Sinclair, Fast uniform generation of regular graphs, *Theoretical Computer Science* **73** (1990), 91–100.

[15] M. Jerrum, A. Sinclair and E. Vigoda, A polynomial-time approximation algorithm for the permanent of a matrix with non-negative entries, *Journal of the Association for Computing Machinery* **51** (2004), 671–697.

[16] M. Jerrum, J.-B. Son, P. Tetali and E. Vigoda, Elementary bounds on Poincaré and log-Sobolev constants for decomposable Markov chains, *Annals of Applied Mathematics* **14** (2004), 1741–1765.

[17] R. Kannan, P. Tetali and S. Vempala, Simple Markov-chain algorithms for generating bipartite graphs and tournaments, *Random Structures and Algorithms* **14** (1999), 293–308.

[18] J.H. Kim and V.H. Vu, Generating random regular graphs, in *Proceedings of 35th Symposium on Theory of Computing* (2003), ACM Press, New York, 213–222.

[19] C. Law and K.-Y. Siu, *Distributed construction of random expander networks*, in *Proceedings of INFOCOM 2003*, IEEE (2003).

[20] R.A. Martin and D. Randall, Sampling adsorbing staircase walks using a new Markov chain decomposition method, in *Proceedings of 41st Symposium on Foundations of Computer Science* (2000), IEEE, Los Alimitos, pp. 492-502.

[21] B. McKay and N. Wormald, Uniform generation of random regular graphs of moderate degree, *Journal of Algorithms* **11** (1990), 52–67.

[22] G. Pandurangan, P. Raghavan and E. Upfal, Building low-diameter peer-to-peer networks, *IEEE Journal on Selected Areas in Communications* **26** (2003), 995–1002.

[23] R.W. Robinson and N.C. Wormald, Almost all regular graphs are Hamiltonian, *Random Structures and Algorithms* **5** (1994), 363–374.

[24] A. Sinclair, Improved bounds for mixing rates of Markov chains and multicommodity flow, *Combinatorics, Probability and Computing* **1** (1992), 351–370.

[25] A. Steger and N. Wormald, Generating random regular graphs quickly, *Combinatorics, Probability and Computing* **8** (1999), 377–396.

[26] G. Tinhofer, On the generation of random graphs with given properties and known distribution, *Applied Computer Science Berichte zur Praktischen Informatik* **13** (1979), 265–297.

[27] W. Tutte, A short proof of the factor theorem for finite graphs, *Canadian Journal of Mathematics* **6** (1954), 347–352.

[28] N. Wormald, *Some problems in the enumeration of labelled graphs*, Ph.D. thesis, University of Newcastle, 1978.