INFORMATION AND COMPUTATION 82, 93–133 (1989)

Approximate Counting, Uniform Generation and Rapidly Mixing Markov Chains*

ALISTAIR SINCLAIR AND MARK JERRUM

Department of Computer Science, University of Edinburgh, Edinburgh EH9 3JZ, Scotland

The paper studies effective approximate solutions to combinatorial counting and uniform generation problems. Using a technique based on the simulation of ergodic Markov chains, it is shown that, for self-reducible structures, almost uniform generation is possible in polynomial time provided only that randomised approximate counting to within some arbitrary polynomial factor is possible in polynomial time. It follows that, for self-reducible structures, polynomial time randomised algorithms for counting to within factors of the form $(1+n^{-\beta})$ are available either for all $\beta \in \mathbf{R}$ or for no $\beta \in \mathbf{R}$. A substantial part of the paper is devoted to investigating the rate of convergence of finite ergodic Markov chains, and a simple but powerful characterisation of rapid convergence for a broad class of chains based on a structural property of the underlying graph is established. Finally, the general techniques of the paper are used to derive an almost uniform generation procedure for labelled graphs with a given degree sequence which is valid over a much wider range of degrees than previous methods: this in turn leads to randomised approximate counting algorithms for these graphs with very good asymptotic behaviour.

1. INTRODUCTION

This paper is concerned with two classes of problems involving a finite set of combinatorial structures: counting them and generating them uniformly at random.

Combinatorial counting problems have a long and distinguished history. Apart from their intrinsic interest, they arise naturally from investigations in numerous other branches of mathematics and the natural sciences and have given rise to a rich and beautiful theory. Uniform generation problems are less well studied but have a number of computational applications. For example, they can be seen as a way of exploring a large set of structures and constructing typical representatives of it. These may

* An extended abstract of this paper appeared in the "Proceedings of the 13th International Workshop on Graph-Theoretic Concepts in Computer Science, Staffelstein, June/July 1987"; published by Springer-Verlag as Lecture Notes in Computer Science Vol. 314.

0890-5401/89 $3.00

be used to formulate conjectures about the set, or perhaps as test data for the empirical analysis of some heuristic algorithm which takes inputs from the set.

The study of counting problems as a class from a computational perspective was initiated by Valiant (1979a). A parallel approach to generation problems was proposed more recently by Jerrum, Valiant, and Vazirani (1986). In this paper, we improve and extend some results of the latter authors concerning the relationship between counting and generation problems, and in particular the existence of efficient approximation algorithms for their solution.

Typically, the sets of structures we encounter are defined implicitly by some other combinatorial entity drawn from a family of *problem instances*, together with a relation R which associates with each instance x a finite set $R(x)$ of *solutions*, as in the following examples:

1. Problem instances: Boolean formulae B in disjunctive normal form (DNF). Solution set $R(B)$: all satisfying assignments of B.

2. Problem instances: undirected graphs G. Solution set $R(G)$: all 1-factors (perfect matchings) of G.

3. Problem instances: positive integers n. Solution set $R(n)$: all partitions of n.

Thus we can talk about the counting and (uniform) generation problems associated with a relation R: given as input a problem instance x, count or generate the elements of the solution set $R(x)$.

Many naturally occurring relations of this kind exhibit a *self-reducibility* property, first studied by Schnorr (1976). Informally, this means that the solutions in $R(x)$ have a simple inductive construction from the solution sets of a few smaller instances of the same problem. (For precise definitions of terms in this Introduction, the reader is referred to the next section.) All of the above examples are self-reducible: in 1, for instance, there is an obvious (1–1)-correspondence between the satisfying assignments of B and those of the reduced formulae B_T and B_F, which are obtained from B by substituting for one of its variables the values true and false respectively.

For self-reducible relations, an efficient (i.e., polynomial time) algorithm for the counting problem immediately yields a polynomial time uniform generation procedure. This approach has been used extensively in the literature to generate particular combinatorial structures, such as 3 above, for which counting information is readily available, typically in the form of a recurrence relation (see, e.g., Guénoche, 1983; Nijenhuis and Wilf, 1978; Wilf, 1981).

Efficient exact solutions to counting problems are, however, relatively uncommon: indeed there are many natural relations, among them 1 and 2

above, whose counting problem is $\#P$-complete and hence apparently intractable (Valiant, 1979a, 1979b), but whose *construction* problem can be solved in polynomial time. (Given a problem instance x, the construction problem asks for a solution $y \in R(x)$ if one exists, and the answer "no" otherwise.) In some cases, such as 1 above, the structures can also be generated efficiently. Note that generation seems to be harder in general than construction since it requires that all solutions be "equally accessible."

In circumstances where exact methods are elusive, it is natural to seek efficient procedures for counting structures *approximately* in some appropriate sense. Following Stockmeyer (1983) and Karp and Luby (1983), we allow our counting algorithms to flip coins, and demand that they produce an answer which approximates $|R(x)|$ within some specified factor with high probability.

It turns out that, for self-reducible relations, the problems of approximate counting and *almost* uniform generation, in which a small bias in the output distribution over $R(x)$ is allowed, are closely related. (An almost uniform generator will generally be as useful in practice as a uniform one and may be effectively indistinguishable from it.) Specifically, Jerrum, Valiant, and Vazirani (1986) show how the standard reduction from generation to exact counting can be modified so as to yield an almost uniform generator given only approximate counting estimates, provided these are within a factor of $1 + O(|x|^{-k_R})$ of the correct values, where $|x|$ is the input size and $k_R > 0$ is a constant depending on R. Conversely, approximate counting to within *any* factor of the form $1 + |x|^{-\beta}$, with $\beta \in \mathbf{R}$, is polynomial time reducible to almost uniform generation. They also locate these two problems for general NP relations within the second level of the (probabilistic) polynomial time hierarchy (Stockmeyer, 1977).

In this paper, we present an improved polynomial time reduction from almost uniform generation to approximate counting for self-reducible relations, in which the counting estimates need only be within a factor of $1 + O(|x|^{\alpha})$ of the exact values, for an *arbitrary* real constant α. Thus a very crude counting procedure (to within a constant factor, say) can be used to generate solutions almost uniformly, and so can in turn be bootstrapped in polynomial time to a counting procedure which approximates within a factor of $1 + \varepsilon$ for any specified $\varepsilon > 0$. Moreover, the runtime of the improved procedure depends only polynomially on ε^{-1}. (Such an algorithm is often called a *fully polynomial randomised approximation scheme.*) A remarkable consequence of this result is that the concept of approximate counting to within factors of the form $1 + O(|x|^{-\beta})$, for $\beta \in \mathbf{R}$, is *robust* with respect to polynomial time computation for the large class of self-reducible relations.

The novel reduction is accomplished by stochastic simulation of an ergodic Markov chain whose structure mirrors the self-reducibility of the relation in question. The states of the chain include the solutions of

interest, and as the chain evolves it converges to a stationary distribution which is uniform over these states. Therefore, provided the convergence is rapid enough, a modest number of simulation steps will ensure that the final state is almost uniformly distributed over the solution set. A similar approach, based on a rather different type of Markov chain, can be used to generate in more direct fashion various structures such as matchings in graphs. This is the subject of another paper (Jerrum and Sinclair, 1988; see also Sinclair, 1988).

As a stepping stone to the above result, we derive a simple characterisation of rapid convergence, in a suitably defined sense, for a broad class of finite ergodic Markov chains in terms of a structural property of the underlying weighted graph. This characterisation, which is related to recent work by Alon (1986) on eigenvalues and expander graphs, appears to be quite generally applicable and we believe it to be of independent interest. Further examples of its use appear in (Jerrum and Sinclair, 1988; Sinclair, 1988). Very recently, a similar characterisation for Markov chains was discovered independently by Lawler and Sokal (1988).

Finally, as a concrete example of these ideas in action we consider the problem of generating labelled graphs with specified vertex degrees and a specified excluded subgraph. Using a result of McKay (1985) which provides analytic counting estimates for these graphs, we show that it is possible to generate them in polynomial time from a distribution which is very close to uniform provided only that the maximum degree grows no faster than $O(m^{1/4})$, where m is the number of edges. Although the problem is apparently not self-reducible under this restriction, our techniques can still be applied with a little extra work. This result represents a considerable improvement on hitherto known methods (Bollobás, 1980; Wormald, 1984). It also implies the existence of polynomial time randomised algorithms for counting such graphs to within a factor of $1 + m^{-\beta}$, for any desired $\beta \in \mathbf{R}$. Since the approach here is quite general, it seems likely that other natural combinatorial counting and generation problems can be treated in similar fashion.

The remainder of the paper is organised as follows. In Section 2 we introduce some definitions and notation. Section 3 covers the characterisation of rapid convergence for Markov chains and is largely self-contained. In Section 4 we use this characterisation to establish the improved reduction from almost uniform generation to approximate counting and discuss the implications of this result. Lastly, in Section 5 we apply the ideas of Section 4 to the degree constrained graph problem mentioned above.

2. Definitions and Notation

This section is devoted to establishing a formal framework for the notions mentioned in the Introduction.

Let Σ be a fixed finite alphabet in which the combinatorial structures of interest are to be encoded, and let $R \subseteq \Sigma^* \times \Sigma^*$ be a binary relation over Σ. For each string (*problem instance*) $x \in \Sigma^*$ the corresponding *solution set* with respect to R is

$$R(x) = \{ y \in \Sigma^* : \langle x, y \rangle \in R \}.$$

We shall always assume that these sets are finite. Note that we make no distinction between strings which do not encode a "valid" problem instance and those which encode a problem instance with empty solution set. Thus the formal counterpart of Example 1 of the Introduction is

$$R = \{ \langle x, y \rangle : x \in \Sigma^* \text{ encodes a Boolean formula } B \text{ in DNF}$$
$$y \in \Sigma^* \text{ encodes a satisfying assignment of } B \}.$$

Throughout we shall move freely between the formal and informal problem descriptions, assuming always that the encoding scheme used is "reasonable" in the sense of Garey and Johnson (1979).

The *counting problem* for a relation R over Σ involves computing the function $\#R: \Sigma^* \to \mathbf{N}$ defined by $\#R(x) = |R(x)|$. As indicated in the Introduction, we shall be concerned with effective approximate solutions to this problem which estimate the value of the function within a specified factor. This notion of approximation, which is familiar from combinatorial optimisation and asymptotic analysis, has also been applied to counting problems in computer science by Stockmeyer (1983) and Karp and Luby (1983). (A less conventional and much more severe definition of approximate counting is studied by Cai and Hemachandra, 1986.)

If a, $\hat{a}$, and r are non-negative real numbers with $r \geqslant 1$, we say that $\hat{a}$ *approximates* a *within ratio* r if $\hat{a}r^{-1} \leqslant a \leqslant \hat{a}r$. Let R be a relation over Σ, and ρ a real-valued function of the natural number n such that $\rho(n) \geqslant 1$ for all $n \in \mathbf{N}$. A *randomised approximate counter for* R *within ratio* ρ is a probabilistic algorithm $\mathscr{C}$ whose output on input $x \in \Sigma^*$ is a non-negative real-valued random variable $\mathscr{C}(x)$ satisfying

$$\Pr(\mathscr{C}(x) \text{ approximates } \#R(x) \text{ within ratio } \rho(|x|)) \geqslant \tfrac{3}{4}.$$

If $\mathscr{C}$ is in fact deterministic then it is an *approximate counter* for R within ratio ρ. In either case, $\mathscr{C}$ is *polynomially time-bounded* if it runs within time $p(|x|)$ for some polynomial p and all inputs $x \in \Sigma^*$.

The significance of the lower bound of $\frac{3}{4}$ in the above definition lies in the

fact that it allows the counter to be "powered" so that the probability of producing a bad estimate becomes very small in polynomial time. (This would still hold if $\frac{3}{4}$ were replaced by any fixed constant greater than $\frac{1}{2}$.) More precisely, we have

LEMMA 2.1. *If there exists a polynomially time-bounded randomised approximate counter $\mathscr{C}$ for R within ratio ρ, then there exists a probabilistic algorithm $\mathscr{C}'$ which on inputs $\langle x, \delta \rangle \in \Sigma^* \times \mathbf{R}^+$ runs in time polynomial in $|x|$ and $\lg \delta^{-1}$, and whose output is a random variable $\mathscr{C}'(x, \delta)$ satisfying*

$$\Pr(\mathscr{C}'(x, \delta) \text{ approximates } \#R(x) \text{ within ratio } \rho(|x|)) \geqslant 1 - \delta.$$

Proof. The required procedure $\mathscr{C}'$ makes $p(\lg \delta^{-1})$ calls to $\mathscr{C}$, with input x, for a suitable polynomial p and returns the median of the values obtained. For the details, see Lemma 6.1 of (Jerrum *et al.*, 1986). ∎

In the (*uniform*) *generation problem* for a relation $R \subseteq \Sigma^* \times \Sigma^*$, we are given an input $x \in \Sigma^*$ and asked to select an element of $R(x)$ at random in such a way that each solution has equal a priori probability of being chosen. In practice, the strict uniformity requirement can generally be weakened slightly, and we say that a probabilistic algorithm $\mathscr{G}$ is an *almost uniform generator* for R if its output on inputs $\langle x, \varepsilon \rangle \in \Sigma^* \times \mathbf{R}^+$ is a random variable $\mathscr{G}(x, \varepsilon)$ satisfying

(i) $\mathscr{G}(x, \varepsilon)$ takes values in the set $R(x) \cup \{?\}$ with $? \notin \Sigma$, and

$$R(x) \neq \varnothing \Rightarrow \Pr(\mathscr{G}(x, \varepsilon) = ?) \leqslant \tfrac{1}{2}.$$

(ii) There exists a function $\phi: \Sigma^* \times \mathbf{R}^+ \to (0, 1]$ such that, for all $y \in \Sigma^*$,

$$y \notin R(x) \Rightarrow \Pr(\mathscr{G}(x, \varepsilon) = y) = 0$$

$$y \in R(x) \Rightarrow (1 + \varepsilon)^{-1} \phi(x, \varepsilon) \leqslant \Pr(\mathscr{G}(x, \varepsilon) = y) \leqslant (1 + \varepsilon) \phi(x, \varepsilon).$$

Thus ε represents the pointwise *bias* tolerated in the output distribution. An almost uniform generator is *fully polynomial* (f.p.) if it runs in time bounded by a polynomial in $|x|$ and $\lg \varepsilon^{-1}$: in this case, the generator may be regarded as effectively indistinguishable from a uniform one by polynomial time statistical tests. Note that as a matter of convenience we allow the generator to *fail* (i.e., output the special symbol ?) with probability $\leqslant \frac{1}{2}$. This can always be made to decay exponentially fast using repeated trials; moreover, if the construction problem is solvable in polynomial time, as is often the case for the relations we consider, then we can force the generator *never* to fail when $R(x)$ is non-empty.

Remark. In the above definitions we have not been precise about our model of randomised computation. For definiteness, we take this to be the probabilistic Turing machine of Gill (1977), in which the only source of randomness is a fair coin. However, we shall feel free to express algorithms in terms of much more general branching probabilities involving the ratio of two previously computed integers. This behaviour can always be simulated by a fair coin to a high degree of accuracy at negligible extra cost. In particular, the notions of polynomial time approximation algorithm presented here are robust with respect to such changes in the model of computation.

Next we formalise the concept of self-reducibility described in the Introduction. A relation $R \subseteq \Sigma^* \times \Sigma^*$ is (*polynomial time*) *self-reducible*, in the sense of Schnorr (1976), if

(i) There exists a polynomial time computable length function $l_R: \Sigma^* \to \mathbf{N}$ such that $l_R(x) = O(|x|^{k_R})$ for some constant $k_R > 0$, and

$$y \in R(x) \Rightarrow |y| = l_R(x) \qquad \forall x, y \in \Sigma^*.$$

(ii) For all $x \in \Sigma^*$ with $l_R(x) = 0$, the predicate $\Lambda \in R(x)$ can be tested in polynomial time. (Λ denotes the empty string over Σ.)

(iii) There exist polynomial time computable functions $\psi: \Sigma^* \times \Sigma^* \to \Sigma^*$ and $\sigma: \Sigma^* \to \mathbf{N}$ satisfying

$$\begin{aligned} \sigma(x) &= O(\lg|x|) \\ l_R(x) > 0 &\Leftrightarrow \sigma(x) > 0 && \forall x \in \Sigma^* \\ |\psi(x, w)| &\leqslant |x| && \forall x, w \in \Sigma^* \\ l_R(\psi(x, w)) &= \max\{l_R(x) - |w|, 0\} && \forall x, w \in \Sigma^* \end{aligned}$$

and such that each solution set can be expressed in the form

$$R(x) = \bigcup_{w \in \Sigma^{\sigma(x)}} \{wy: y \in R(\psi(x, w))\}.$$

Condition (iii) provides an inductive construction of the solution sets as follows: if the solution length $l_R(x)$ is greater than zero, $R(x)$ can be partitioned into classes according to the small initial segment w of length $\sigma(x)$, and each class can then be expressed as the solution set of another instance $\psi(x, w)$, concatenated with w. The partitioning of satisfying assignments of a DNF formula indicated in the Introduction is easily seen to be of the required form, under some natural encoding. An *atom* is an instance $x \in \Sigma^*$ with solution length $l_R(x) = 0$: in the above example, these would include (encodings of) the constants true and false, viewed as DNF formulae. Con-

dition (ii) says that, for atoms x, we can test in polynomial time whether $R(x) = \varnothing$ or $R(x) = \{\Lambda\}$. Note that this, together with condition (iii), implies that we can test whether a candidate solution $y \in \Sigma^*$ belongs to any solution set $R(x)$ in time polynomial in $|x| + |y|$. In view of condition (i), the *existence problem* associated with R is therefore in NP. (R is called a *p-relation* in Jerrum *et al.*, 1986.) It appears that the vast majority of naturally occurring relations can be formulated so as to be self-reducible.

It is conceptually helpful to capture the inductive construction of solutions of a self-reducible relation explicitly in a tree structure. For each $x \in \Sigma^*$ with $R(x) \neq \varnothing$, the *tree of derivations* $T_R(x)$ is a rooted tree in which each vertex v bears both a *problem instance label* $\text{inst}(v) \in \Sigma^*$ and a *partial solution label* $\text{sol}(v) \in \Sigma^*$, defined inductively as follows:

(i) The root u has labels $\text{inst}(u) = x$ and $\text{sol}(u) = \Lambda$.

(ii) For any vertex v in $T_R(x)$, if the problem instance $z = \text{inst}(v)$ is an atom then v is a leaf. Otherwise, define

$$W(v) = \{w \in \Sigma^{\sigma(z)} : R(\psi(z, w)) \neq \varnothing\}.$$

(Note that $W(v)$ is non-empty.) Then v has a child v_w for each $w \in W(v)$, with labels $\text{inst}(v_w) = \psi(z, w)$ and $\text{sol}(v_w) = \text{sol}(v) \cdot w$.

Note that the labels $\text{sol}(v)$ are distinct, while the $\text{inst}(v)$ are in general not. It should be clear that the labels $\text{sol}(v)$ for leaves v are precisely the elements of $R(x)$, so there is a (1–1)-correspondence between leaves and solutions. More generally, for any vertex v of $T_R(x)$ there is a (1–1)-correspondence between the solution set $R(\text{inst}(v))$ and the leaves of the maximal subtree rooted at v. The bounds on σ and ψ in the definition of self-reducibility ensure that the depth of $T_R(x)$ is bounded by $l_R(x) = O(|x|^{k_R})$, and that the number of children of any vertex is also polynomially bounded.

In order to infer the structure of the tree of derivations, it is clearly necessary to solve the existence problem for the relation in question. Since we will not always be able to do this with certainty, it is useful to define the *self-reducibility tree* $\tilde{T}_R(x)$ as above except that the restriction $R(\psi(z, w)) \neq \varnothing$ in the definition of $W(v)$ is removed. Obviously $\tilde{T}_R(x)$ contains $T_R(x)$ as a subgraph and their labels agree. All solutions in $R(x)$ still occur precisely once as labels of leaves of $\tilde{T}_R(x)$, but there may be other leaves whose partial solution labels are not in $R(x)$. The depth and vertex degree of $\tilde{T}_R(x)$ remain polynomially bounded as before.

Most known uniform generation algorithms for combinatorial structures (see, e.g., Nijenhuis and Wilf, 1978) may be viewed as instances of the following generic reduction to the corresponding counting problem. Given

that the structures are described by a self-reducible relation R, select a random path from the root of the tree of derivations to a leaf (solution), at each stage choosing the next edge with probability proportional to the number of solutions in the maximal subtree rooted at its lower end. This information may be obtained from a counter which evaluates the function $\#R$ for appropriate problem instance labels in the tree. By appending a correction process based on the a posteriori probability of the path, this procedure can be made to work even if the counter is slightly inaccurate, specifically if it is within ratio $1+O(n^{-k_R})$, where $k_R>0$ is a constant satisfying $l_R(x)=O(|x|^{k_R})$ (see Jerrum *et al.*, 1986, and also Bach, 1983). Furthermore, if the counter is randomised then a f.p. *almost* uniform generator is still obtained. (In the latter case we have to work with the self-reducibility tree rather than the tree of derivations.) Since a f.p. almost uniform generator can itself be used to construct a polynomially time-bounded randomised approximate counter within ratio $1+n^{-\beta}$ for any desired $\beta \in \mathbf{R}$, counters within the threshold ratio $1+O(n^{-k_R})$ can be bootstrapped to achieve arbitrarily good asymptotic behaviour (Jerrum *et al.*, 1986).

When rather cruder counting information is available (to within a constant factor, say) the above "one-pass" technique breaks down owing to the accumulation of errors which are too large to be corrected. We will therefore adopt a more flexible, self-correcting approach in which a random process moves dynamically around the tree, with backtracking allowed. The generator we will construct in Section 4 views the vertices of the tree of derivations as the states of a Markov chain $\mathscr{MC}(x)$ in which there is a non-zero transition probability between two states iff they are adjacent in the tree. The transition probabilities themselves are computed with the aid of the crude approximate counter. Clearly all states communicate, so that, leaving aside questions of periodicity, if the chain is allowed to evolve for t steps from any initial state the distribution of its final state approaches a unique stationary distribution as $t \to \infty$. Now suppose that this distribution is *uniform* over the leaves of the tree. Then we get an almost uniform generator by simulating the chain for sufficiently many steps starting at (say) the root and, if the final state is a leaf, outputting the corresponding solution.

The efficiency of this procedure will of course depend crucially on the *rate of convergence* of the chain. In particular, since the size of the tree is in general exponential in $|x|$, we require the chain to be very close to stationarity after visiting only a small fraction of its states. There appear to be no quantitative results in the literature which would readily provide useful analytic bounds on the rate of approach to stationarity in this case. Accordingly, in the next section we develop a characterisation of rapid convergence, in a suitably defined sense, for a broad class of Markov

chains. This will enable us to show in Section 4 that the almost uniform generation procedure sketched above is in fact fully polynomial.

Remark. The Markov chain approach to almost uniform generation also points to a fundamentally different strategy which appeals neither to self-reducibility nor to counting. Here transitions are made more or less directly between solutions by means of local perturbations, in a manner suggested by Broder (1986) and familiar from Monte Carlo studies in statistical physics (Binder, 1976). We can also consider generating solutions from more general distributions by making appropriate adjustments to the stationary distribution of the chain. The machinery developed in the next section can be used to analyse Markov chains of this kind as well: for applications, the reader is referred to (Jerrum and Sinclair, 1988; Sinclair, 1988).

3. Markov Chains and Rapid Mixing

We assume that the reader is familiar with the elementary theory of finite Markov chains in discrete time: an introduction can be found in (Feller, 1968, Chap. XV). First, we establish some terminology and quote some basic facts.

Let the sequence of random variables $(X_t)_{t=0}^{\infty}$ be a time-homogeneous Markov chain on a finite *state space* $[N] = \{0, 1, ..., N-1\}$, $N \geqslant 1$, with *transition matrix* $P = (p_{ij})_{i,j=0}^{N-1}$. (All Markov chains in this paper will be assumed to be of this form.) Thus for any ordered pair i, j of states the quantity $p_{ij} = \Pr(X_{t+1} = j \mid X_t = i)$ is the *transition probability* from state i to state j and is independent of t. The matrix P is non-negative and *stochastic*, i.e., its row sums are all unity. For $s \in \mathbf{N}$, the *s-step transition matrix* is simply the power $P^s = (p_{ij}^{(s)})$; thus $p_{ij}^{(s)} = \Pr(X_{t+s} = j \mid X_t = i)$, independent of t. We denote the distribution of X_t by the row vector $\boldsymbol{\pi}^{(t)'} = (\pi_i^{(t)})_{i=0}^{N-1}$, so that $\pi_i^{(t)} = \Pr(X_t = i)$. Here $\boldsymbol{\pi}^{(0)'}$ denotes the *initial distribution*, and $\boldsymbol{\pi}^{(t)'} = \boldsymbol{\pi}^{(0)'} P^t$ for all $t \in \mathbf{N}$. Usually we will have $\pi_i^{(0)} = 1$ for some $i \in [N]$ (and 0 elsewhere); i is then called the *initial state*.

The chain is *ergodic* if there exists a distribution $\boldsymbol{\pi}' = (\pi_i) > \mathbf{0}$ over $[N]$ such that

$$\lim_{s \to \infty} p_{ij}^{(s)} = \pi_j \qquad \forall i, j \in [N].$$

In this case, we have that $\boldsymbol{\pi}^{(t)'} = \boldsymbol{\pi}^{(0)'} P^t \to \boldsymbol{\pi}'$ pointwise as $t \to \infty$, and the limit is independent of $\boldsymbol{\pi}^{(0)'}$. The *stationary distribution* $\boldsymbol{\pi}'$ is the unique vector satisfying $\boldsymbol{\pi}' P = \boldsymbol{\pi}'$, $\sum_i \pi_i = 1$, i.e., the unique normalised left eigenvector of P with eigenvalue 1. Necessary and sufficient conditions for ergodicity are that the chain should be (a) *irreducible*, i.e., for each pair of

states $i, j \in [N]$, there is an $s \in \mathbf{N}$ such that $p_{ij}^{(s)} > 0$ (j can be reached from i in a finite number of steps); and (b) *aperiodic*, i.e., $\gcd\{s: p_{ij}^{(s)} > 0\} = 1$ for all $i, j \in [N]$.

Suppose now that we wish to sample elements of the state space, assumed very large, according to the stationary distribution π'. This problem arises frequently in the mathematical modelling of physical systems, where states correspond to configurations of the system and appropriate functions of the stationary process to physical constants or parameters (Binder, 1976), and is also fundamental to stochastic optimisation techniques such as simulated annealing (Kirkpatrick, Gellatt, and Vecchi, 1983). In the applications we have in mind here, some of the states can be identified with certain combinatorial structures of interest and π' is uniform over these states. However, our approach will address the general problem.

The desired distribution can be realised by picking an arbitrary initial state and simulating the transitions of the Markov chain according to the probabilities p_{ij}, which we assume can be computed locally as required. As the number t of simulation steps increases, the distribution of the random variable X_t will approach π'. Clearly, for this process to be effective it is necessary to know a priori how many steps are required to achieve a distribution sufficiently close to π' for our purposes, or in other words to have some bound on the rate of convergence of the chain. As a time-dependent measure of deviation from the limit, we define the *relative pointwise distance* (r.p.d.) over a non-empty subset $U \subseteq [N]$ after t steps by

$$\Delta_U(t) = \max_{i,j \in U} \frac{|p_{ij}^{(t)} - \pi_j|}{\pi_j}.$$

Thus $\Delta_U(t)$ gives the largest relative difference between $\pi^{(t)'}$ and π' at any state $j \in U$, maximised over all possible initial states $i \in U$.[1] The inclusion of the parameter U merely allows us to specify that certain portions of the state space are not relevant in the sampling process, as will prove helpful later. The aim of this section is to obtain a useful upper bound on Δ_U as a function of t. In particular, we want to investigate conditions under which convergence is *rapid* in the sense that $\Delta_{[N]}(t)$ becomes very close to 0 while $t \ll N$: this is sometimes referred to as the "rapid mixing" property (Aldous, 1981).

A number of techniques for investigating the rate of convergence of

[1] This measure, which is a symmetrical version of the *separation distance* defined in (Aldous and Diaconis, 1986), has been chosen by analogy with our definition of almost uniform generation in Section 2. We could alternatively have used a measure based on the *variation distance*, namely $\Delta'_U(t) = \max_{i \in U} \sum_j |p_{ij}^{(t)} - \pi_j|$. For most interesting chains, this choice makes no essential difference to the rapid convergence criterion.

Markov chains have recently been proposed by other authors. Methods based on coupling (Aldous, 1981) and stopping times (Aldous and Diaconis, 1986) are attractive and yield tight bounds for simple chains, such as random walks on hypercubes and various card-shuffling processes. However, the analysis involved appears to become extremely complex in more interesting cases where the chain lacks a highly symmetrical structure. The approach used here based on the eigenvalues of the transition matrix is more classical, but seems hitherto to have been of little practical value. Our contribution is to develop from it a simple yet powerful tool for obtaining good analytic bounds for a broad class of chains. Crucially, we will be able to apply this tool to chains which have not proved amenable to analysis by other means.

An ergodic Markov chain is said to be *time-reversible* if either (and hence both) of the following equivalent conditions holds:

(i) For all $i, j \in [N]$, $p_{ij}\pi_i = p_{ji}\pi_j$.

(ii) The matrix $D^{1/2}PD^{-1/2}$ is symmetric, where $D^{1/2}$ is the diagonal matrix $\mathrm{diag}(\pi_0^{1/2}, \ldots, \pi_{N-1}^{1/2})$ and $D^{-1/2}$ is its inverse.

Condition (i) says that in the stationary distribution the expected numbers of transitions per unit time from state i to state j and from state j to state i are equal, and is usually called the "detailed balance" property. As we shall see, time-reversible chains are particularly susceptible to detailed analysis, and for this reason play a major role in applications where a rigorous quantitative treatment is necessary (see, e.g., Keilson, 1979).

It is illuminating to identify an ergodic time-reversible chain with a weighted undirected graph containing self-loops as follows. The vertex set of the graph is the state space $[N]$ of the chain, and for each pair of states i, j (which need not be distinct) the edge (i, j) has weight $w_{ij} = \pi_i p_{ij} = \pi_j p_{ji}$. By detailed balance this definition is consistent. Thus there is an edge of non-zero weight between i and j iff $p_{ij} > 0$. We call this graph the *underlying graph* of the chain. It should be clear that such a chain is uniquely specified by its underlying graph.

As already stated, the stationary distribution π' of an ergodic chain is a left eigenvector of P with eigenvalue $\lambda_0 = 1$. Let $\{\lambda_i : 1 \leqslant i \leqslant N-1\}$, with $\lambda_i \in \mathbf{C}$, be the remaining eigenvalues (not necessarily distinct) of P. By standard Perron–Frobenius theory for non-negative matrices (Seneta, 1981), these satisfy $|\lambda_i| < 1$ for $1 \leqslant i \leqslant N-1$. Furthermore, the transient behaviour of the chain, and hence its rate of convergence, is governed by the magnitude of the eigenvalues λ_i. In the time-reversible case, condition (ii) of the definition implies that the eigenvalues of P are just those of the similar *symmetric* matrix $D^{1/2}PD^{-1/2}$, and so are all real. This fact leads to a clean formulation of the above dependence.

PROPOSITION 3.1. *Let P be the transition matrix of an ergodic time-reversible Markov chain, π' its stationary distribution and $\{\lambda_i: 0 \leqslant i \leqslant N-1\}$ its (necessarily real) eigenvalues, with $\lambda_0 = 1$. Then for any non-empty subset $U \subseteq [N]$ and all $t \in \mathbf{N}$, the relative pointwise distance $\Delta_U(t)$ satisfies*

$$\Delta_U(t) \leqslant \frac{\lambda_{\max}^t}{\min_{j \in U} \pi_j},$$

where $\lambda_{max} = \max\{|\lambda_i|: 1 \leqslant i \leqslant N-1\}$.

Proof. Let $D^{1/2}$ and $D^{-1/2}$ be as in the definition of time-reversibility, so that the matrix $A = D^{1/2} P D^{-1/2}$ is symmetric with the same eigenvalues as P, and these are real. Hence we can select an orthonormal basis $\{\mathbf{e}^{(i)'}: 0 \leqslant i \leqslant N-1\}$ for $\mathbf{R}^N$ consisting of left eigenvectors of A, where $\mathbf{e}^{(i)'} = (e_j^{(i)})$ has associated eigenvalue λ_i and $e_j^{(0)} = \pi_j^{1/2}$ for $j \in [N]$.

Following (Keilson, 1979), A has the spectral representation

$$A = \sum_{i=0}^{N-1} \lambda_i \mathbf{e}^{(i)} \mathbf{e}^{(i)'} = \sum_{i=0}^{N-1} \lambda_i E^{(i)},$$

where $E^{(i)} = \mathbf{e}^{(i)} \mathbf{e}^{(i)'}$ is a *dyad* (i.e., has rank 1) with $E^{(i)} E^{(j)} = 0$ for $i \neq j$, and $E^{(i)2} = E^{(i)}$. It follows that, for any $t \in \mathbf{N}$, $A^t = \sum_i \lambda_i^t E^{(i)}$, and hence

$$P^t = D^{-1/2} A^t D^{1/2} = \sum_{i=0}^{N-1} \lambda_i^t (D^{-1/2} \mathbf{e}^{(i)})(\mathbf{e}^{(i)'} D^{1/2})$$

$$= \mathbf{1}_N \pi' + \sum_{i=1}^{N-1} \lambda_i^t (D^{-1/2} \mathbf{e}^{(i)})(\mathbf{e}^{(i)'} D^{1/2}),$$

where $\mathbf{1}_N$ is the N-vector all of whose entries are 1; in component form,

$$p_{jk}^{(t)} = \pi_k + \sqrt{\frac{\pi_k}{\pi_j}} \sum_{i=1}^{N-1} \lambda_i^t e_j^{(i)} e_k^{(i)}.$$

By definition, the r.p.d. $\Delta_U(t)$ is therefore given by

$$\Delta_U(t) = \max_{j,k \in U} \frac{|\sum_{i=1}^{N-1} \lambda_i^t e_j^{(i)} e_k^{(i)}|}{\sqrt{\pi_j \pi_k}}$$

$$\leqslant \lambda_{\max}^t \frac{\max_{j,k \in U} \sum_{i=1}^{N-1} |e_j^{(i)}|\, |e_k^{(i)}|}{\min_{j \in U} \pi_j}$$

$$\leqslant \frac{\lambda_{\max}^t}{\min_{j \in U} \pi_j},$$

where the second inequality follows from the Cauchy–Schwarz inequality

(Kolmogorov and Fomin, 1970, p. 38) and the orthonormality of the $\mathbf{e}^{(i)}$. ∎

Proposition 3.1 says that a time-reversible Markov chain will be rapidly mixing in the sense indicated earlier provided that π' is not extremely small in any state of interest and that $\lambda_{\max}$ is bounded away from 1. The first of these conditions can be checked immediately from our knowledge of π' and is rarely violated in practice. We therefore focus our attention on the second condition, which is not so easily handled. (Note that P is a large matrix so that direct numerical evaluation of the eigenvalues is not feasible.)

Suppose the eigenvalues of P are ordered so that $1 = \lambda_0 > \lambda_1 \geqslant \cdots \geqslant \lambda_{N-1} > -1$. Then the value of $\lambda_{\max}$ is governed by λ_1 and λ_{N-1}, the latter being significant only if some of the eigenvalues are negative. Negative eigenvalues correspond to oscillatory, or "near-periodic" behaviour and cannot occur if each state is equipped with a sufficiently large self-loop probability. Specifically, it is enough to have $\min_j p_{jj} \geqslant \frac{1}{2}$. To see this, let I_N denote the $N \times N$ identity matrix and consider the non-negative matrix $2P - I_N$, whose eigenvalues are $\mu_i = 2\lambda_i - 1$. By Perron–Frobenius, $\mu_i \geqslant -1$ for all $i \in [N]$, which implies that $\lambda_{N-1} \geqslant 0$.

In fact, negative eigenvalues never present an essential obstacle to rapid mixing because any chain can be modified in a simple way so that the above condition holds without risk of slowing down the convergence too much:

PROPOSITION 3.2. *Let P be the transition matrix of an ergodic time-reversible Markov chain, and $1 = \lambda_0 > \lambda_1 \geqslant \cdots \geqslant \lambda_{N-1} > -1$ its eigenvalues. Then the modified chain with transition matrix $P' = \frac{1}{2}(I_N + P)$ is also ergodic and time-reversible with the same stationary distribution, and its eigenvalues $\{\lambda_i'\}$, similarly ordered, satisfy $\lambda_{N-1}' > 0$ and $\lambda_{\max}' = \lambda_1' = \frac{1}{2}(1 + \lambda_1)$.*

From the above discussion, it is sufficient for rapid mixing to bound the *second eigenvalue* λ_1 away from 1. We shall do this by relating λ_1 to a more accessible structural property of the underlying graph.

Intuitively, we would expect an ergodic chain to converge rapidly if it is unlikely to "get stuck" in any subset S of the state space whose total stationary probability is fairly small. We can formalise this idea by considering the cut edges which separate S from the rest of the space in the underlying graph, and stipulating that these must be capable of supporting a sufficiently large "flow" in the graph, viewed as a network. With this in mind, for any non-empty subset S of states with non-empty complement $\bar{S}$ in $[N]$ we define the quantity $\Phi_S = F_S / C_S$, where

$$C_S = \sum_{i \in S} \pi_i \qquad \text{the } \textit{capacity} \text{ of } S;$$

$$F_S = \sum_{\substack{i \in S \\ j \in \bar{S}}} p_{ij}\pi_i \qquad \text{the } \textit{ergodic flow} \text{ out of } S.$$

Note that $0 < F_S \leqslant C_S < 1$. Φ_S may be visualised as the conditional probability that the stationary process crosses the cut from S to $\bar{S}$ in a single step, given that it starts in S. Finally, we define the global *conductance* of the chain by

$$\Phi = \min_{\substack{0 < |S| < N \\ C_S \leqslant 1/2}} \Phi_S.$$

It is easy to see that $F_S = F_{\bar{S}}$ for all such sets S. This implies that $\Phi_{\bar{S}} = \Phi_S C_S (1 - C_S)^{-1}$, so we may equivalently write

$$\Phi = \min_{0 < |S| < N} \max\{\Phi_S, \Phi_{\bar{S}}\}.$$

Now suppose that the chain is time-reversible, and let G be its underlying graph. Then for all S as above we have

$$F_S = F_{\bar{S}} = \sum_{\substack{i \in S \\ j \in \bar{S}}} w_{ij},$$

a function of the edge weights of G. The conductance $\Phi \equiv \Phi(G)$ may then be viewed as a structural property of the weighted graph G. In view of the above remarks, we might hope that $\Phi(G)$, which in some sense measures the minimum relative connection strength between "small" subsets S and the rest of the space, has some bearing on the rate of convergence of the chain. This relationship is manifested via a bound on the second eigenvalue λ_1.

LEMMA 3.3. *For an ergodic time-reversible Markov chain with underlying graph G, the second eigenvalue λ_1 of the transition matrix satisfies*

$$\lambda_1 \leqslant 1 - \frac{\Phi(G)^2}{2}.$$

Proof. Let $\mathbf{e}' = (e_i)_{i=0}^{N-1}$ be an eigenvector of P with associated eigenvalue $\lambda < 1$, and define the matrix $Q = I_N - P$ (the "Laplace operator" associated with P). Then clearly

$$\mathbf{e}'Q = (1 - \lambda)\mathbf{e}'. \tag{1}$$

Define the subset of states $S = \{i \in [N]: e_i > 0\}$. It is easy to check that, since P is stochastic and $\lambda < 1$, $\sum_i e_i = 0$. Hence $0 < |S| < N$, and we may assume without loss of generality that $C_S = \sum_{i \in S} \pi_i \leqslant \frac{1}{2}$. Now let $\hat{\mathbf{e}}' = (\hat{e}_i)$ be the vector defined by

$$\hat{e}_i = \begin{cases} e_i/\pi_i, & \text{for } i \in S; \\ 0, & \text{otherwise.} \end{cases}$$

Renumbering states as necessary, we shall assume that $\hat{e}_0 \geqslant \hat{e}_1 \geqslant \cdots \geqslant \hat{e}_{N-1}$, which implies also that $S = \{0, 1, ..., r\}$ for some r with $0 \leqslant r < N-1$.

Taking the scalar product of (1) with $\hat{\mathbf{e}}'$ gives

$$(\mathbf{e}'Q, \hat{\mathbf{e}}') = (1-\lambda)(\mathbf{e}', \hat{\mathbf{e}}'). \tag{2}$$

The right-hand side of (2) is just

$$(1-\lambda) \sum_{i \in S} \pi_i \hat{e}_i^2. \tag{3}$$

Note that if $Q = (q_{ij})$ then $q_{ij} = -p_{ij}$ for $i \neq j$, and $q_{ii} = 1 - p_{ii} = \sum_{j \neq i} p_{ij}$, so we can expand the left-hand side of (2) as

$$\begin{aligned} \sum_{i \in S} \sum_{j \in [N]} \hat{e}_i q_{ji} e_j &\geqslant \sum_{i \in S} \sum_{j \in S} \hat{e}_i q_{ji} e_j \\ &= -\sum_{i \in S} \sum_{\substack{j \in S \\ j \neq i}} w_{ij} \hat{e}_i \hat{e}_j + \sum_{i \in S} \sum_{j \neq i} w_{ij} \hat{e}_i^2 \\ &= -2 \sum_{i<j} w_{ij} \hat{e}_i \hat{e}_j + \sum_{i<j} w_{ij} (\hat{e}_i^2 + \hat{e}_j^2) \\ &= \sum_{i<j} w_{ij} (\hat{e}_i - \hat{e}_j)^2, \end{aligned} \tag{4}$$

where the inequality follows from the fact that all contributions with $j \notin S$ are positive. Using (3) and (4), Eq. (2) therefore yields

$$1 - \lambda \geqslant \frac{\sum_{i<j} w_{ij} (\hat{e}_i - \hat{e}_j)^2}{\sum_{i \in S} \pi_i \hat{e}_i^2}. \tag{5}$$

Now consider the sum

$$\sum_{i<j} w_{ij} (\hat{e}_i + \hat{e}_j)^2 \leqslant 2 \sum_{i<j} w_{ij} (\hat{e}_i^2 + \hat{e}_j^2) \leqslant 2 \sum_{i \in S} \pi_i \hat{e}_i^2.$$

Combining this with (5) gives

$$1-\lambda \geqslant \frac{\sum_{i<j} w_{ij}(\hat{e}_i-\hat{e}_j)^2}{\sum_{i\in S}\pi_i\hat{e}_i^2}\cdot\frac{\sum_{i<j} w_{ij}(\hat{e}_i+\hat{e}_j)^2}{2\sum_{i\in S}\pi_i\hat{e}_i^2}$$
$$\geqslant \frac{1}{2}\left(\frac{\sum_{i<j} w_{ij}(\hat{e}_i^2-\hat{e}_j^2)}{\sum_{i\in S}\pi_i\hat{e}_i^2}\right)^2, \tag{6}$$

where we have used the Cauchy–Schwarz inequality. To complete the proof, we need to relate the quotient in (6) to the quantity $\Phi(G)$. To do this, consider the increasing sequence $(S_k)_{k=0}^r$ of subsets of S with $S_k=\{0,...,k\}$. The numerator of the quotient in (6) may be expressed in terms of ergodic flows across the boundaries between successive sets S_k as follows:

$$\sum_{i<j} w_{ij}(\hat{e}_i^2-\hat{e}_j^2)=\sum_{i<j} w_{ij}\sum_{i\leqslant k<j}(\hat{e}_k^2-\hat{e}_{k+1}^2)$$
$$=\sum_{k=0}^{r}(\hat{e}_k^2-\hat{e}_{k+1}^2)\sum_{\substack{i\in S_k\\ j\notin S_k}} w_{ij}$$
$$=\sum_{k=0}^{r}(\hat{e}_k^2-\hat{e}_{k+1}^2)F_{S_k}. \tag{7}$$

Now the capacities of the S_k satisfy $C_{S_k}\leqslant C_S\leqslant\frac{1}{2}$ for $0\leqslant k\leqslant r$, and hence by definition of Φ, $F_{S_k}\geqslant\Phi(G)C_{S_k}$. We therefore get from (7)

$$\sum_{i<j} w_{ij}(\hat{e}_i^2-\hat{e}_j^2)\geqslant\Phi(G)\sum_{k=0}^{r}(\hat{e}_k^2-\hat{e}_{k+1}^2)C_{S_k}$$
$$=\Phi(G)\sum_{k=0}^{r}(\hat{e}_k^2-\hat{e}_{k+1}^2)\sum_{i=0}^{k}\pi_i$$
$$=\Phi(G)\sum_{i=0}^{r}\pi_i\sum_{k=i}^{r}(\hat{e}_k^2-\hat{e}_{k+1}^2)$$
$$=\Phi(G)\sum_{i\in S}\pi_i\hat{e}_i^2.$$

This inequality ensures that the quotient in (6) is bounded below by $\Phi(G)$, so that finally

$$1-\lambda\geqslant\frac{\Phi(G)^2}{2},$$

as required. ∎

Combining Propositions 3.1 and 3.2 and Lemma 3.3 we arrive at the main result of this section, which says that the number of steps required for an ergodic time-reversible Markov chain to lose its memory (approach stationarity) is $O(\Phi(G)^{-2} \lg(1/\pi_{\min}))$, where G is its underlying graph and $\pi_{\min}$ the minimum stationary probability of any state. Thus, under mild assumptions about the stationary distribution, convergence is rapid if $\Phi(G)$ is not too small.

THEOREM 3.4. *Let G be the underlying graph of an ergodic time-reversible Markov chain, modified if necessary as in Proposition 3.2 to ensure that $\min_j p_{jj} \geq \frac{1}{2}$, and $\boldsymbol{\pi}'$ its stationary distribution. Then for any non-empty subset $U \subseteq [N]$ and all $t \in \mathbf{N}$, the relative pointwise distance $\Delta_U(t)$ satisfies*

$$\Delta_U(t) \leq \frac{(1 - \Phi(G)^2/2)^t}{\min_{i \in U} \pi_i}.$$

Remarks. (a) Theorem 3.4 has a converse which states that, under the same assumptions, $\Delta_{[N]}(t) \geq (1 - 2\Phi(G))^t$ (Sinclair, 1988). Hence we effectively have a *characterisation* of rapid mixing for time-reversible chains in terms of the graph-theoretic quantity Φ. (Note however that this does not cover cases in which π_i is extremely small for some i. Such chains may exhibit a range of convergence behaviour regardless of the value of $\Phi(G)$.)

(b) In the interest of simplicity, we have appealed to the rather crude device of Proposition 3.2 for eliminating negative eigenvalues: the effect of this operation on the conductance is to reduce it by a factor of $\frac{1}{2}$. In practice it may often be possible to reason about negative eigenvalues on an ad hoc basis for the chain at hand. Proposition 3.1 and Lemma 3.3 may then be used directly to get a bound on $\Delta_U(t)$.

(c) Lemma 3.3 and its proof parallel an earlier continuous result of Cheeger (1970) for Riemannian manifolds. In the discrete setting, the lemma and its converse are closely related to recent work of Alon (1986) and Alon and Milman (1985) (see also Dodziuk, 1984) in which a relationship between a similar structural property of simple, unweighted graphs and the second eigenvalue of the adjacency matrix is established. This property, called the *magnification*, measures the minimum number of *vertices* adjacent to a small subset S as a fraction of $|S|$, and is a generalisation of the widely studied concept of *expansion* for bipartite graphs. Our conductance Φ is a weighted edge analogue of magnification, and is the natural quantity to study in the present application. The significance of Alon's result as a sufficient condition for rapid mixing for certain Markov chains has been noted by several authors; in particular, Aldous (1987) states a restricted form of Theorem 3.4 for random walks on

regular graphs. Our characterisation based on the conductance seems to provide a cleaner and more natural formulation of this connection. Very recently, Lawler and Sokal (1988) have independently discovered results similar to those presented in this section but in a rather more general context.

Theorem 3.4 allows us to investigate the rate of convergence of a time-reversible chain by examining the structure of its underlying graph. For rapid mixing, this will typically involve deriving bounds of the form $\Phi(G) = \Omega((\lg N)^{-k})$ as the number N of states increases, for some constant k. The exciting feature of this characterisation is that, with a bit more work, suitable conductance bounds may actually be derived analytically for a number of interesting chains. In this way we are able for the first time to establish the rapid mixing property for chains which lack a high degree of symmetry and which have not proved amenable to analysis by other methods. In the next section, we show how this can be done for the chain based on the tree of derivations mentioned at the end of the last section. Further applications may be found in Jerrum and Sinclair (1988) and Sinclair (1988).

4. Reductions

We return now to the main theme of this paper, namely the construction of an efficient almost uniform generator for a self-reducible relation given only very approximate counting information.

Let $R \subseteq \Sigma^* \times \Sigma^*$ be self-reducible, and $x \in \Sigma^*$ a problem instance with $R(x) \neq \emptyset$. As advertised in Section 2, our aim is to set up an ergodic Markov chain $\mathscr{MC}(x)$ whose states are the vertices of the tree of derivations $T_R(x)$ and whose stationary distribution is uniform over the leaves of the tree.

The chain is based on an elaboration of the standard reduction from uniform generation to exact counting indicated at the end of Section 2. We may view the counter in this reduction as assigning to each edge of the tree of derivations an integer weight equal to the number of leaves in the subtree rooted at its lower end; the process is then a transient Markov chain in which the transition probabilities from any vertex (state) to its children are proportional to the corresponding edge weights. Suppose now that the process is no longer constrained to move downwards but may also backtrack from any vertex to its parent, the transition probabilities to *all* adjacent vertices being proportional to the edge weights: thus from any internal vertex upward and downward movements are equally likely. To eliminate periodicity, we add to each state a self-loop probability of $\frac{1}{2}$.

Viewing this process as a symmetric random walk with reflecting barriers on the *levels* of the tree, it is easy to see that it converges rapidly (essentially in time polynomial in the depth of the tree) to a stationary distribution which is uniform over levels and also uniform over leaves. Hence a short simulation of the chain generates leaves almost uniformly, and the probability of failure can be made small by repeated trials. Now suppose that we have available only an *approximate* counter for R, so that the edge weights in the tree are no longer accurate. Then we have grounds for optimism that this procedure might still work efficiently: the hope is that, since each edge weight influences transitions in both directions, the process will actually be self-correcting.

Suppose then that we are given a polynomially time-bounded approximate counter $\mathscr{C}$ for R within ratio $\rho(n) = 1 + O(n^{\alpha})$ for some $\alpha \in \mathbf{R}$. Thus the error ratio in $\mathscr{C}$ need not even be constant, but may increase polynomially with the problem size. Note first that, since R is self-reducible, $\mathscr{C}$ can always be modified so as to give an exact answer (which will be either 0 or 1) when its input is an atom; also, its output may always be rounded up to the nearest integer at the cost of adding at most 1 to ρ. We shall assume throughout this section that $\mathscr{C}$ incorporates these modifications. We may also assume without loss of generality that ρ is monotonically increasing. To begin with, we shall consider the case where $\mathscr{C}$ is deterministic; the additional technical problems posed by randomised counters will be dealt with later.

For a problem instance x as above, let V, E be the vertex and edge sets respectively of $T_R(x)$, and set $m = l_R(x)$, $r = \rho(|x|)$. Note that both m and r are polynomially bounded in $|x|$, and that the depth of the tree is at most m. For each edge $(u, v) \in E$ define the quantity

$$f(u, v) = \begin{cases} \mathscr{C}(\text{inst}(u)), & \text{if } v \text{ is the parent of } u; \\ \mathscr{C}(\text{inst}(v)), & \text{otherwise.} \end{cases} \tag{8}$$

(Recall that inst$(\cdot)$ gives the problem instance associated with any vertex of the tree.) Since $\mathscr{C}$ is deterministic, $f: E \to \mathbf{N}^+$ is a well-defined function on E. The crucial property to bear in mind is that for any edge $e \in E$, $f(e)$ approximates within ratio r the number of leaves in the maximal subtree below e.

Next we define for each vertex $v \in V$ a *degree*

$$d(v) = \sum_{u:(u,v) \in E} f(u, v). \tag{9}$$

Note that $d(v) \geq 1$ for all $v \in V$, and that $d(v) = 1$ if v is a leaf because $\mathscr{C}$ is

exact for atoms. For each ordered pair v, u of vertices, the transition probability p_{vu} from v to u is then defined to be

$$p_{vu} = \begin{cases} f(u, v)/2d(v), & \text{if } (u, v) \in E; \\ 1/2, & \text{if } u = v; \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

Thus there is a non-zero transition probability between two states iff they are adjacent in the tree. The self-loop probability $\frac{1}{2}$ ensures that the chain is aperiodic. It is clearly also irreducible, and hence ergodic, and it is a simple matter to verify that the stationary distribution $\pi' = (\pi_v)_{v \in V}$ is proportional to the degrees, i.e.,

$$\pi_v = \frac{d(v)}{D} \qquad \forall v \in V, \tag{11}$$

where $D = \sum_{v \in V} d(v)$.

Let us first check that sampling from V according to the distribution π' does in fact give us an efficient generation procedure for R, so that the approach of the previous section applies. Since leaves of the tree correspond to solutions, while other vertices must necessarily correspond to failure of the generator, we have to verify that π' is uniform and sufficiently large over the leaves. (Recall that an almost uniform generator must have bounded failure probability.) Uniformity follows directly from the fact that $d(v) = 1$ for all leaves v, so $\pi_v = 1/D$. That this is not too small is a consequence of the following lemma.

LEMMA 4.1. *In the stationary distribution of $\mathcal{MC}(x)$, the probability of being at a leaf is at least $1/2rm$.*

Proof. Observe that the degree sum D over the tree $T_R(x)$ may be written

$$D = \sum_{v \in V} d(v) = 2 \sum_{e \in E} f(e).$$

Now consider the collection of edges at some fixed level of the tree. By (8) the weight of each such edge approximates within ratio r the number of leaves in the maximal subtree rooted at its lower end. Since these subtrees are disjoint, the aggregated weight of all edges at this level is at most $r \#R(x)$. Summing over all levels of the tree yields the bound

$$D = 2 \sum_{e \in E} f(e) \leqslant 2rm \#R(x). \tag{12}$$

Since $\pi_v = 1/D$ for each leaf v, the stationary probability of being at a leaf is

$$\sum_{\text{leaves } v} \pi_v = \frac{\#R(x)}{D} \geqslant \frac{1}{2rm},$$

as claimed. ∎

Recall that m and r are each polynomially bounded in $|x|$, so the failure probability can be reduced to $\frac{1}{2}$ by repeating the entire experiment only polynomially many times.

We now adress the trickier question of the rate of convergence of the chain $\mathscr{MC}(x)$, assuming for the moment that an efficient stepwise simulation from some initial state can be performed. It is easy to see that the chain is time-reversible by virtue of detailed balance.[2] We will therefore try to estimate the conductance Φ of its underlying graph, as defined in the previous section, and appeal to Theorem 3.4.

LEMMA 4.2. *Let G be the underlying graph of the Markov chain $\mathscr{MC}(x)$ defined above. Then the conductance of G satisfies*

$$\Phi(G) \geqslant (4r^2m)^{-1}.$$

Proof. Note first that in G each edge $(u, v) \in E$ has weight $w_{uv} = f(u, v)/2D$, while the loop at v has weight $d(v)/2D$ and all other edges have weight zero. In what follows, we will identify subsets of V with the subgraphs of $T_R(x)$ which they induce. If $S \subseteq V$ is a subtree (connected subgraph) of $T_R(x)$, we let root(S) denote the vertex of S at minimum distance from root(V), the root of $T_R(x)$.

In order to bound the conductance of G, we claim that it suffices to consider flows out of all subtrees S with root$(S) \neq$ root(V). (Informally, the process will converge fast because it is quite likely to emerge from any such subtree, travelling upwards, within a small number of steps.) To see this, note first that $\Phi(G) \geqslant \min \Phi_S$, where the minimisation is over all non-empty subsets $S \subseteq V$ with root$(V) \notin S$. But we may write any such S as the union $T_0 \cup \cdots \cup T_l$ of disjoint subtrees no pair of which is connected by an edge in $T_R(x)$, and we have

$$\Phi_S \equiv \frac{F_S}{C_S} = \frac{\sum_i F_{T_i}}{\sum_i C_{T_i}} \geqslant \min_i \frac{F_{T_i}}{C_{T_i}} = \min_i \Phi_{T_i}.$$

Hence it is clear that $\Phi(G) \geqslant \min \Phi_S$, where the minimisation is now over all subtrees S of $T_R(x)$ with root$(S) \neq$ root(V), as claimed.

[2] This is actually also an immediate corollary of the fact that $\mathscr{MC}(x)$ is a *tree process*, i.e., the edges corresponding to non-zero transition probabilities form a tree.

A lower bound on Φ_S for such subtrees is readily obtained. We may assume without loss of generality that S is maximal. Then the flow out of S is just $F_S = f(\mathrm{cut}(S))/2D$, where $\mathrm{cut}(S)$ is the cut edge connecting S to the rest of the tree. But since $f(\mathrm{cut}(S))$ approximates the number of leaves $L(S)$ in S within ratio r, the flow is bounded below by

$$F_S \geqslant \frac{L(S)}{2rD}. \tag{13}$$

On the other hand, summing edge weights in the subtree S as in the proof of Lemma 4.1, we may easily derive the bound

$$\sum_{v \in S} d(v) = f(\mathrm{cut}(S)) + 2 \sum_{e \in E(S)} f(e) \leqslant 2rmL(S), \tag{14}$$

where $E(S)$ is the set of edges in S. Since $C_S = \sum_{v \in S} d(v)/D$, putting (13) and (14) together yields

$$\Phi_S = \frac{F_S}{C_S} \geqslant \frac{1}{4r^2m},$$

which completes the proof of the lemma. ∎

Since both m and r are at most polynomial in the problem size $|x|$, the bound in Lemma 4.2 is sufficient to ensure that the chains $\mathscr{MC}(x)$ are rapidly mixing. More precisely, for each $x \in \Sigma^*$ with $R(x) \neq \varnothing$, let $\Delta^{(x)}(t)$ denote the r.p.d. of $\mathscr{MC}(x)$ over the whole state space V after t steps. Then we have:

LEMMA 4.3. *There exists a function* $q: \Sigma^* \times \mathbf{R}^+ \to \mathbf{N}$ *such that* $q(x, \varepsilon)$ *is polynomially bounded in* $|x|$ *and* $\lg \varepsilon^{-1}$, *and for each* $x \in \Sigma^*$ *with* $R(x) \neq \varnothing$,

$$\Delta^{(x)}(t) \leqslant \varepsilon/2 \qquad \textit{for all } t \geqslant q(x, \varepsilon).$$

Proof. For each such x, the chain $\mathscr{MC}(x)$ satisfies the conditions of Theorem 3.4. Furthermore, we have seen that $\min_{v \in V} \pi_v = 1/D$, which by (12) is bounded below by $(2rm|\Sigma|^m)^{-1}$. (Note that solutions are strings of length m over the alphabet Σ, so $\#R(x) \leqslant |\Sigma|^m$.) Applying Theorem 3.4, and using the bound on $\Phi(G)$ obtained in Lemma 4.2, we get

$$\Delta^{(x)}(t) \leqslant 2rm|\Sigma|^m \,(1 - (32r^4m^2)^{-1})^t.$$

The function q defined by

$$q(x, \varepsilon) = 32r^4m^2(\ln(2rm) + m \ln|\Sigma| + \ln(2/\varepsilon))$$

then clearly satisfies the requirements of the lemma. ∎

We are now in a position to state the first major result of this section.

THEOREM 4.4. *Let $R \subseteq \Sigma^* \times \Sigma^*$ be self-reducible. If there exists a polynomially time-bounded (deterministic) approximate counter for R within ratio $1 + O(n^{\alpha})$ for some $\alpha \in \mathbf{R}$, then there exists a fully polynomial almost uniform generator for R.*

Proof. Let $\mathscr{C}$ be the approximate counter for R as specified above. We proceed to construct an almost uniform generator $\mathscr{G}$ for R which uses $\mathscr{C}$ as an oracle.

On inputs $\langle x, \varepsilon \rangle \in \Sigma^* \times \mathbf{R}^+$, $\mathscr{G}$ initially calls $\mathscr{C}$ with input x and halts with output ? if $\mathscr{C}(x) = 0$, which is the case if and only if $R(x) = \varnothing$. Otherwise, $\mathscr{G}$ simulates the Markov chain $\mathscr{M}\mathscr{C}(x)$ defined above, starting at the root of $T_R(x)$. From their definition in (10), the transition probabilities from any state can be computed by appropriate calls to $\mathscr{C}$ since we may easily keep track of the problem instance labels of the vertices. (Note that we are also inferring the structure of the tree locally in the process.) The simulation halts after $q(x, \varepsilon)$ steps, where q is the function specified in Lemma 4.3, outputting the corresponding solution if the final state is a leaf and ? otherwise. Since the degree of the tree is bounded by a polynomial in $|x|$ and all problem instance labels have size at most $|x|$, each step can be simulated in polynomial time. Together with the bound on q from Lemma 4.3, this ensures that $\mathscr{G}$ always halts in time bounded by a polynomial in $|x|$ and $\lg \varepsilon^{-1}$.

Now let $\mathscr{G}(x, \varepsilon)$ be the output random variable of $\mathscr{G}$ on inputs $\langle x, \varepsilon \rangle$. Clearly $\mathscr{G}$ only ever outputs valid solutions, so $\Pr(\mathscr{G}(x, \varepsilon) = y) = 0$ if $y \notin R(x)$. Moreover, since the chain has been allowed to evolve for sufficiently many steps, we may deduce from Lemma 4.3 that

$$(1 - \varepsilon/2)\frac{1}{D} \leqslant \Pr(\mathscr{G}(x, \varepsilon) = y) \leqslant (1 + \varepsilon/2)\frac{1}{D}$$

for any solution $y \in R(x)$, where D depends only on x and is defined at (11). Assuming as we may that $\varepsilon \leqslant 1$, this ensures that the bias is within the required bound ε. Finally, if $R(x) \neq \varnothing$ Lemma 4.1 implies that

$$\Pr(\mathscr{G}(x, \varepsilon) = ?) \leqslant (1 - 1/2rm)(1 + \varepsilon/2).$$

Assuming further that $\varepsilon \leqslant 1/rm$, this bound can be reduced to $1/e < 1/2$ using only $(2rm)^2$ iterations of the procedure $\mathscr{G}$. ∎

If the approximate counter in Theorem 4.4 is *randomised* as defined in Section 2, so that it may occasionally produce arbitrarily bad results, the reduction still goes through but at the cost of some tiresome technicalities. We summarise the proof in this case.

THEOREM 4.5. *The result of Theorem* 4.4 *still holds even if the approximate counter for R is randomised.*

Proof (sketch). Let x be a problem instance for which $R(x) \neq \varnothing$. As before, assume ρ is monotonic and set $m = l_R(x)$, $r = \rho(|x|)$. We begin by considering the intermediate case where the counter $\mathscr{C}$ is randomised but always produces estimates which are within ratio r of their correct values. We again define a Markov chain $\mathscr{M}\mathscr{C}(x)$ on the tree $T_R(x)$, whose transition probabilities are determined as follows. Suppose the process is currently at vertex v, and let U be the set of children of v. For each $u \in U \cup \{v\}$, make a call $\mathscr{C}(\text{inst}(u))$ to the counter and denote the result $c(u)$; then make a further independent set of calls $\mathscr{C}(\text{inst}(u))$ for the same vertices u and denote their sum $d(v)$. Finally, make a transition to an adjacent vertex u with probability

$$\begin{aligned} &c(u)/4r^2 d(v), && \text{if } u \text{ is a child of } v; \\ &c(v)/4r^2 d(v), && \text{if } u \text{ is the parent of } v, \end{aligned} \tag{15}$$

and remain at v otherwise. (Note that the factor $1/4r^2$ ensures that these transitions are always well defined, and that there is a self-loop probability of at least $\frac{1}{2}$ in each state; we have used $\frac{1}{4}$ rather than $\frac{1}{2}$ for consistency with the second part of the proof.) Clearly, if $\mathscr{C}$ is deterministic this reduces (except for a uniform factor of $1/2r^2$) to the original chain. In the randomised case, it is easy to see that the transition probability p_{vu} from v to u is actually the expectation

$$\mathrm{E}\left(\frac{f(u,v)}{4r^2\, d(v)}\right) = \frac{1}{4r^2}\,\mathrm{E}\,(f(u,v))\,\mathrm{E}\left(\frac{1}{d(v)}\right),$$

where the random variable $f(u, v)$ is defined as in (8) and is independent of $d(v)$. The stationary distribution π' therefore satisfies

$$\pi_v \propto 1/\mathrm{E}(d(v)^{-1}) \qquad \forall v \in V,$$

and the fact that $\mathscr{C}$ is exact for atoms implies that $d(v) = 1$ with probability 1 for leaves v. The chain is clearly still time-reversible, and the rest of the proof goes through essentially as in the deterministic case, with $d(v)$ and $f(u, v)$ replaced by $1/\mathrm{E}(d(v)^{-1})$ and $\mathrm{E}(f(u, v))$ respectively.

Now suppose that the counter may in addition produce arbitrarily bad results with some small probability δ: by Lemma 2.1 we may assume that $\delta \leqslant 2^{-p(|x|)}$ for all problem instances in the tree, where p is any desired polynomial. Since we are no longer able to infer the structure of $T_R(x)$ with certainty, we must now work in the larger self-reducibility tree $\tilde{T}_R(x)$ (cf. Section 2). We let V, $\tilde{V}$ denote the vertex sets of $T_R(x)$ and $\tilde{T}_R(x)$ respec-

tively. Note that $\tilde{V}\setminus V$ consists of a union of disjoint maximal subtrees of $\tilde{T}_R(x)$. Some modifications to the transition probabilities are also necessary. At vertex $v \in \tilde{V}$, we compute values $c(u)$, for $u \in U \cup \{v\}$, and $d(v)$ as before, where now U is the set of children of v in $\tilde{T}_R(x)$. If $d(v) = 0$ then we make a transition to the parent of v (if it exists) with probability $\frac{1}{4}$, and remain at v with probability $\frac{3}{4}$. Otherwise, we test whether $\sum_u c(u) > 4r^2 d(v)$: if so, we remain at v; if not, we make a transition to a neighbouring vertex with probabilities as in (15). (Note that the self-loop probability in each state is at least $\frac{1}{2}$.) Once again, the leaves of $T_R(x)$ are treated as a special case.

This chain is clearly ergodic on some subset of $\tilde{V}$ containing V, namely those states which communicate with the root. Henceforth we redefine $\tilde{V}$ to include only such states. The chain is also still time-reversible because it is a tree process. Let us first observe that the new vertices in $\tilde{V}\setminus V$ have negligible effect. All transitions from V to $\tilde{V}\setminus V$ occur with at most tiny probability δ, so if started in V the process is unlikely to leave V during the course of the simulation. Should it enter a subtree in $\tilde{V}\setminus V$, however, the random variable $d(v)$ at the root vertex v will take the value 0 with probability very close to 1, thus causing the chain to leave the subtree rapidly. In fact, it is not hard to see that the stationary probability π_v of a vertex $v \in \tilde{V}\setminus V$ is at most $O(\delta^k)$, where k is the distance of v from V in $\tilde{T}_R(x)$. As a result, the total weight of $\tilde{V}\setminus V$ in the stationary distribution is small. Furthermore, the large exit probability from subtrees $S \subseteq \tilde{V}\setminus V$ ensures a lower bound on Φ_S similar to that in the proof of Lemma 4.2.

Examination of the transition probabilities *within* V reveals that we can view this portion as a chain of the restricted kind described in the first part of the proof whose transition probabilities have been perturbed by a factor in the range $(1 \pm \delta')$, where δ' depends on δ and can be made exponentially small. It is then easy to see that the stationary probabilities of states in V undergo similarly small perturbations in the range $(1 \pm \delta')^m$. As a result, a lower bound as in the proof of Lemma 4.2 also holds for subtrees S with root$(S) \in V$, and so for all subtrees, which again implies that the conductance $\Phi(G)$ is suitably bounded below. Assuming that the simulation starts at the root, we therefore get rapid convergence over the subset V of the state space,[3] which is sufficient since V includes all leaves of $T_R(x)$. A test applied to leaf labels ensures that no non-solutions are output. ∎

Remark. There is actually a simpler way to prove Theorem 4.5, though the resulting algorithm is less natural and the process is no longer strictly a Markov chain. Note that the simulation of Theorem 4.4 can still be

[3] More precisely, we are using the r.p.d. $\Delta_V(t)$ over V here, as defined in Section 3. Theorem 3.4 implies a sufficient condition for rapid mixing with respect to this measure also.

performed using a randomised counter if we arrange to *remember* the outputs of the counter on all previous calls so that each edge weight is computed at most once. Provided all values returned by the counter are accurate within the given ratio, we are effectively in the situation of Theorem 4.4 and our earlier analysis applies. By powering the counter, we can ensure that this condition fails to hold with very small probability, so the effect on the overall process will be negligible.

Theorem 4.5 has an interesting consequence for counting problems. First let us generalise our notion of approximate counting to allow the error ratio in the estimate to be specified as part of the input. If R is a relation over Σ, then a *randomised approximation scheme* for $\#R$ is a probabilistic algorithm $\mathscr{C}$ whose output on input $\langle x, \varepsilon \rangle \in \Sigma^* \times \mathbf{R}^+$ is a non-negative real-valued random variable $\mathscr{C}(x, \varepsilon)$ satisfying

$$\Pr(\mathscr{C}(x, \varepsilon) \text{ approximates } \#R(x) \text{ within ratio } 1+\varepsilon) \geqslant \tfrac{3}{4}.$$

$\mathscr{C}$ is a *fully polynomial* randomised approximation scheme (fpras) if it runs in time bounded by a polynomial in $|x|$ and ε^{-1} for all inputs $\langle x, \varepsilon \rangle$. Note that the definition of fully polynomial here differs from that for almost uniform generators in the absence of a logarithm.

Jerrum, Valiant, and Vazirani (1986) show how to construct a fpras for $\#R$ for a self-reducible relation R given a f.p. almost uniform generator for R. In view of Theorem 4.5, this means that we can bootstrap a very crude counter for R to one with arbitrarily good asymptotic behaviour as follows. Suppose there exists a polynomially time-bounded randomised approximate counter for R within ratio $1+O(n^\alpha)$ for some real α (which we may think of as large). Then by Theorem 4.5 there exists a f.p. almost uniform generator for R, and hence by the above result of Jerrum *et al.* a fpras for $\#R$. (Recall that Jerrum *et al.* establish this only for $\alpha \leqslant -k_R$, a small threshold value as defined in Section 2.) We have therefore proved our next result.

THEOREM 4.6. *Let $R \subseteq \Sigma^* \times \Sigma^*$ be self-reducible. If there exists a polynomially time-bounded randomised approximate counter for R within ratio $1+O(n^\alpha)$ for some $\alpha \in \mathbf{R}$, then there exists a fully polynomial randomised approximation scheme for $\#R$.*

The chief significance of Theorem 4.6 is that it establishes a notion of approximate counting which is *robust* with respect to polynomial time computation, at least for the large class of self-reducible relations: a randomised approximate counter within ratio $1+O(n^\alpha)$ can always be improved to one within ratio $1+n^{-\beta}$ for *any* desired real β with at most a polynomial increase in runtime. Thus we are justified in classifying the

counting problem for a self-reducible relation R as *tractable* if there exists a polynomial time randomised algorithm which with high probability approximates $\#R(x)$ to within some factor of the form $1 + O(|x|^{\alpha})$, with $\alpha \in \mathbf{R}$. We suggest that this notion will be useful in the future classification of hard counting problems, as studied, e.g., by Stockmeyer (1983) and Karp and Luby (1983).

Remarks. (a) The bootstrapping described in Theorem 4.6 actually holds for a rather trivial reason if the relation R has a property which we might call *self-embeddability*. Informally, R is self-embeddable if there exists an efficiently computable function ξ which takes a pair x_1, x_2 of problem instances and embeds them in an instance $\xi(x_1, x_2)$, whose size is at most linear in $|x_1|$ and $|x_2|$ and whose solution set is in (1–1)-correspondence with the product set $R(x_1) \times R(x_2)$. An example is the relation which associates with a directed graph G its set of (directed) Hamiltonian paths: the required embedding function ξ takes a pair G_1, G_2 of graphs and adds a new vertex v, together with edges from v to all vertices of G_1 and from all vertices of G_2 to v. To bootstrap a counter for a self-embeddable relation, given a problem instance x we apply the embedding construction to obtain an instance z with $\#R(z) = \#R(x)^{p(|x|)}$ for some suitable polynomial p; we then use the counter to approximate $\#R(z)$ and take the $p(|x|)$th root of the result, which yields an improved estimate of $\#R(x)$. Although many natural relations turn out to be self-embeddable, there seem to be a number of significant exceptions among self-reducible relations, including DNF-satisfiability and natural *restricted* versions of familiar relations, such as Hamiltonian paths in planar graphs. Moreover, the Markov chain reduction technique presented here can sometimes be applied even in the absence of self-reducibility. Evidence for this is provided by the relation GRAPHS discussed in the next section, which is apparently neither self-embeddable nor self-reducible under the degree restrictions imposed there.

(b) Theorem 4.5 can also be used to derive a surprising bootstrapping result for generators. Specifically, given a polynomially time-bounded generator for a self-reducible relation R which is almost uniform with bias $O(|x|^{-k_R})$, where k_R is a constant as above, it is possible to construct a f.p. almost uniform generator for R (Sinclair, 1988). The new generator of course achieves *exponentially* small bias in polynomial time.

5. Graphs with Specified Degrees

Given a sequence $\mathbf{g} = (g_0, ..., g_{n-1})$ of non-negative integers, is it possible to efficiently generate labelled graphs with vertex set $\{0, 1, ..., n-1\}$ in which vertex i has degree g_i, $0 \leqslant i \leqslant n-1$, such that each graph occurs with

roughly equal probability? We conclude this paper by showing how the approach of the previous section can be used to answer this question affirmatively, provided that the maximum degree does not grow too rapidly with the number of vertices n.

Our motivation for looking at this problem is twofold. First, there is its inherent interest as indicated below. Second, and perhaps more important, it serves to illustrate how our ideas may be used to exploit a much wider class of asymptotic counting results than has hitherto been possible, even for structures which are not self-reducible. We suggest that other natural structures can be handled similarly.

The special case of the problem in which the graphs are *regular*, i.e., $g_i = k$ for all i and some k, is of particular interest and has been considered by several authors. Regular graphs are a natural class to study in their own right and have recently become an important model in the theory of random graphs (Bollobás, 1985). A generation procedure would provide a means of examining "typical" regular graphs with a given number of vertices and given degree and investigating their properties, about many of which little is known. Furthermore, it has recently been shown by Wormald (1987) that generation techniques for labelled graphs with a given degree sequence can be used in the uniform generation of isomorphism classes of regular graphs.

Wormald (1984) gives efficient algorithms for uniformly generating labelled cubic and degree-4 graphs on n vertices. However, these are based on specific recurrence relations and do not generalise easily to higher degrees. A simpler method discussed in (Wormald, 1984), and already implicit in the work of Bollobás (1980), uniformly generates labelled regular graphs of arbitrary degree k, but the probability of failure remains polynomially bounded only if $k = O((\log n)^{1/2})$. When the degree is permitted to increase more rapidly with n, it seems difficult to generate the graphs with anything approaching equal probabilities: in the approach of Tinhofer (1979), for example, the probabilities associated with different graphs may vary widely. Our method, which relies on the reduction to counting developed in the previous section, requires only that $k = O(n^{1/3})$ and achieves a distribution over the graphs which is asymptotically very close to uniform.

In keeping with our general approach, we begin by defining a relation which describes the graphs of interest. For the sake of clarity, we shall not refer in this section to an encoding scheme: it should however be clear how to translate everything into the formal framework of Section 2. A (*labelled*) *degree sequence* on vertex set $[n] = \{0, ..., n-1\}$ is a sequence $\mathbf{g} = (g_0, ..., g_{n-1})$ of non-negative integers such that $\sum_i g_i = 2e(\mathbf{g})$ is even, and a *graph on* $\mathbf{g}$ is a graph with vertex set $[n]$ in which vertex i has degree g_i, $0 \leqslant i \leqslant n-1$. (All graphs here are assumed to be simple and undirected.)

If the vertex set is understood, we shall identify a graph with its edge set. It is actually convenient to generalise the above problem by allowing a set of forbidden edges to be specified. Accordingly, we define the relation GRAPHS which associates with each problem instance of the form $\langle \mathbf{g}, X \rangle$, where $\mathbf{g}$ is a degree sequence on $[n]$ and X is a labelled graph with vertex set $[n]$, the solution set

$$\text{GRAPHS}(\mathbf{g}, X) = \{G: G \text{ is a graph on } \mathbf{g} \text{ having no edge in common with } X\}.$$

We refer to X as an *excluded graph* for $\mathbf{g}$. Although this relation is self-reducible as it stands, we get a more symmetrical structure using the relation R defined by

$$R(\mathbf{g}, X) = \{\langle G, \omega \rangle: G \in \text{GRAPHS}(\mathbf{g}, X) \text{ and } \omega \text{ is an edge-ordering of } G\}.$$

Clearly, we can move freely between these relations since any solution set $R(\mathbf{g}, X)$ contains precisely $e(\mathbf{g})!$ ordered copies of each element of GRAPHS$(\mathbf{g}, X)$.

Next we specify a self-reducibility on R by defining the tree of derivations $T_R(\mathbf{g}, X)$, assuming that $R(\mathbf{g}, X) \neq \varnothing$. In this tree, the object $\langle G, \omega \rangle$ will be derived by successively adding the edges of G in the order determined by ω. More precisely, the partial solution labels of the tree are in (1–1)-correspondence with pairs $\langle \bar{H}, \omega \rangle$, in which $\bar{H}$ is a graph with vertex set $[n]$ which can be extended to at least one graph in GRAPHS$(\mathbf{g}, X)$, and ω is an edge-ordering of $\bar{H}$. The root has label $\langle \varnothing, \varnothing \rangle$, while the children of the vertex with label $\langle \bar{H}, \omega \rangle$ have labels of the form $\langle \bar{H} \cup \{(i, j)\}, \omega + (i, j) \rangle$ for some edge (i, j), where $\omega + (i, j)$ denotes the extension of ω in which (i, j) is the largest element. The problem instance label of a vertex v is determined by its partial solution label $\langle \bar{H}, \omega \rangle$ as follows. Let $\bar{\mathbf{h}} = (\bar{h}_0, ..., \bar{h}_{n-1})$ be the degree sequence of $\bar{H}$, and define $\mathbf{h} = \mathbf{g} - \bar{\mathbf{h}}$, where the subtraction is pointwise. Also, let Y be the subgraph of $X \cup \bar{H}$ obtained by deleting all edges (i, j) for which either $\bar{h}_i = g_i$ or $\bar{h}_j = g_j$. Then the problem instance label of v is $\langle \mathbf{h}, Y \rangle$.

Note that the deletion of redundant constraints from $X \cup \bar{H}$ is not necessary for the consistency of the tree, but it will prove useful later—in the proof of Lemma 5.3—that Y represents only the *essential* excluded graph. From now on, we will in fact assume that all problem instances $\langle \mathbf{g}, X \rangle$ have had redundant constraints removed. In particular, this means that the problem instance label of the root of the tree is just $\langle \mathbf{g}, X \rangle$. It also justifies our use of $e(\mathbf{g})$ as a measure of input size for this problem when stating approximation results below.

Now that we have a tree of derivations for R, Theorem 4.4 will give us an efficient almost uniform generator for R, and hence for GRAPHS, provided we can count these structures with sufficient accuracy. The

counting problem for GRAPHS has received much attention over a number of years, where the aim has chiefly been to extend the validity of asymptotic estimates to a wider range of degrees (see McKay, 1985 for a brief survey). The best result available to date is due to McKay, and we quote this below.

Given a degree sequence $\mathbf{g}$ on $[n]$ and an excluded graph X for $\mathbf{g}$, let $\mathbf{x}=(x_0, ..., x_{n-1})$ be the degree sequence of X, and define $\gamma(\mathbf{g}, X)=\max\{g_{\max}^2, g_{\max}x_{\max}\}$, where $g_{\max}=\max_i g_i$ and $x_{\max}=\max_i x_i$. We shall use γ to express bounds on the degrees involved in the problem. Furthermore, if $g_{\max}>0$ set

$$\lambda(\mathbf{g})=\frac{1}{4e(\mathbf{g})}\sum_{i=0}^{n-1} g_i(g_i-1); \qquad \mu(\mathbf{g}, X)=\frac{1}{2e(\mathbf{g})}\sum_{(i,j)\in X} g_i g_j.$$

THEOREM 5.1 (McKay, 1985). *There exists a positive constant r_0 with the property that, for any problem instance $\langle \mathbf{g}, X\rangle$ with $g_{\max}>0$ and $\gamma(\mathbf{g}, X)\leqslant e(\mathbf{g})/10$, the quantity*

$$\frac{(2e(\mathbf{g}))!}{e(\mathbf{g})!\, 2^{e(\mathbf{g})}\prod_{i=0}^{n-1} g_i!}\exp(-\lambda(\mathbf{g})-\lambda(\mathbf{g})^2-\mu(\mathbf{g}, X)) \tag{16}$$

approximates $\#\text{GRAPHS}(\mathbf{g}, X)$ *within ratio* $\exp(r_0\gamma(\mathbf{g}, X)^2/e(\mathbf{g}))$.

Remarks. (a) Actually, McKay's result is slightly stronger than this: we have stated it in a simplified form which is adequate for our purposes.

(b) The estimate in Theorem 5.1 immediately leads to a simple method, suggested by Wormald (1984) and implicit in the earlier work of Bollobás (1980), for generating graphs whose degrees grow slowly with the number of edges: make g_i copies of vertex i for each i, generate a *pairing* (i.e., a perfect matching in the complete graph on these vertices) uniformly at random, and then collapse the copies to a single vertex again. The result will be a multigraph on $\mathbf{g}$, and the distribution over GRAPHS$(\mathbf{g}, X)$ is uniform, but the procedure may fail since not all the graphs generated in this way will be simple or avoid X. The exponential factor in (16) can be interpreted as approximating the probability that a randomly chosen pairing yields an element of GRAPHS$(\mathbf{g}, X)$. It is then clear from the definitions of λ and μ that, provided $\gamma(\mathbf{g}, X)=O(\log e(\mathbf{g}))$, this probability is polynomially bounded below, so that the method is effective in this range. For regular graphs, this implies a degree bound of $O((\log n)^{1/2})$.

Let us now restate Theorem 5.1 in a more convenient form.

COROLLARY 5.2. *Let Q, B be fixed real numbers with $Q>0$ and $B\geqslant 100Q^4$. Then for all problem instances $\langle \mathbf{g}, X\rangle$ for which either $e(\mathbf{g})\leqslant B$*

or $\gamma(\mathbf{g}, X) \leqslant Q^2 e(\mathbf{g})^{1/2}$, *the quantity* $\#R(\mathbf{g}, X)$ *can be approximated in polynomial time within a constant ratio.*

Proof. We have already observed that $\#R(g, X) = e(\mathbf{g})! \#\text{GRAPHS}(\mathbf{g}, X)$, so we need only approximate the latter. Note that when $e(\mathbf{g}) > B$ the bound on γ ensures also that $\gamma(\mathbf{g}, X) \leqslant e(\mathbf{g})/10$, so we may appeal to Theorem 5.1. The expression in (16) can clearly be evaluated in polynomial time and yields an approximation within the constant ratio $\exp(r_0 Q^4)$ in all relevant cases, except when $g_{\max} = 0$ or possibly when $e(\mathbf{g}) \leqslant B$. The first case is trivial; to handle the second, observe that for fixed B there are only a constant number of instances, up to relabelling of the vertices, for which $e(\mathbf{g}) \leqslant B$, so all counting in this range may be done exactly by explicit enumeration. (Alternatively, in practice any convenient approximation method may be used, subject to the proviso that it yields the answer 0 iff $\#\text{GRAPHS}(\mathbf{g}, X) = 0$: this property can be tested in polynomial time using matching techniques.) ∎

Now let us see whether Corollary 5.2 is powerful enough to allow us to construct a generation algorithm for GRAPHS via the reduction to counting embodied in Theorem 4.4. Ideally, we might hope to handle instances for which $\gamma(\mathbf{g}, X)$ grows as $O(e(\mathbf{g})^{1/2})$. However, this does not follow immediately since the relation R is no longer self-reducible when restricted in this way. In other words, even if $g_{\max}$ and $x_{\max}$ are suitably bounded, the tree $T_R(\mathbf{g}, X)$ will in general contain vertices whose problem instances $\langle \mathbf{h}, Y \rangle$ are *unbalanced* in the sense that the degrees are rather large compared to the number of edges $e(\mathbf{h})$, so that we cannot guarantee reasonable counting estimates over the whole tree. We will overcome this problem by naïvely *pruning* the tree in such a way as to leave only problem instances which do fall within the bounds of Corollary 5.2, though we will have to do a little work to check that the effects of this are not too drastic.

For any pair Q, B of real numbers with $Q > 0$ and $B \geqslant 100Q^4$, we call a problem instance $\langle \mathbf{g}, X \rangle$ (Q, B)-*balanced* if either $e(\mathbf{g}) \leqslant B$ or $\gamma(\mathbf{g}, X) \leqslant Q^2 e(\mathbf{g})^{1/2}$. If $\langle \mathbf{g}, X \rangle$ is (Q, B)-balanced and $R(\mathbf{g}, X) \neq \varnothing$, then the *pruned* tree $T_R^{(Q,B)}(\mathbf{g}, X)$ with respect to Q, B is obtained by deleting from $T_R(\mathbf{g}, X)$ each vertex whose problem instance label is not (Q, B)-balanced, together with the entire subtree rooted at the vertex.

Now consider defining a time-reversible Markov chain $\mathscr{MC}(\mathbf{g}, X)$ on the pruned tree in precisely the same manner as in Section 4, using the counting estimates of Corollary 5.2. Our first claim is that the conductance bound of Lemma 4.2 still holds, so that $\mathscr{MC}(\mathbf{g}, X)$ is rapidly mixing. To see this, imagine a corresponding chain on the complete tree $T_R(\mathbf{g}, X)$ in which *all* counting estimates are within the constant ratio of Corollary 5.2: clearly, in this case the conductance is bounded as in Lemma 4.2. But $\mathscr{MC}(\mathbf{g}, X)$ is obtained from this chain simply by deleting some subtrees

and, as the reader may readily verify, the removal of extremal portions of a Markov chain cannot decrease its conductance. Hence the bound of Lemma 4.2 applies to $\mathscr{MC}(\mathbf{g}, X)$ also.

We turn now to the effect of the pruning operation on the stationary distribution. As before, the distribution will be proportional to the "degrees" $d(v)$ defined as in (9) and can be made uniform over leaves by counting exactly at this level. (When we speak of "leaves" of the pruned tree, we shall always mean those vertices which are also leaves of the original tree $T_R(\mathbf{g}, X)$.) However, since we have lost some leaves by pruning, it is by no means obvious that the *induced* distribution on GRAPHS$(\mathbf{g}, X)$ obtained by forgetting the edge orderings is even close to uniform, or that the failure probability is still bounded. Both these facts will follow from the lemma below, which says that in the pruning process we lose at most a small fraction of the leaves corresponding to any graph in GRAPHS$(\mathbf{g}, X)$ provided that the constants Q, B are suitably chosen.

LEMMA 5.3. *Let $\mathscr{F}$ be a family of problem instances $\langle \mathbf{g}, X \rangle$ satisfying $\max\{g_{\max}, x_{\max}\} = O(e(\mathbf{g})^{1/4})$ and β a real constant. Then there exists a pair of real numbers Q, B as above (which depend on $\mathscr{F}$ and β) such that, for each instance $\langle \mathbf{g}, X \rangle \in \mathscr{F}$ with* GRAPHS$(\mathbf{g}, X) \neq \varnothing$, *and each* $G \in$ GRAPHS$(\mathbf{g}, X)$, *the pruned tree $T_R^{(Q,B)}(\mathbf{g}, X)$ contains at least $e(\mathbf{g})!(1 - e(\mathbf{g})^{-\beta}/4)$ leaves with solution label G.*

We postpone the rather technical proof of this lemma until we have examined its consequences, which constitute the central results of this section.

THEOREM 5.4. *For any fixed real β, there exists a polynomial time algorithm which generates elements of* GRAPHS$(\mathbf{g}, X)$ *almost uniformly with bias at most $e(\mathbf{g})^{-\beta}$, provided that the degrees involved are bounded as $\max\{g_{\max}, x_{\max}\} = O(e(\mathbf{g})^{1/4})$.*

Proof. We assume without loss of generality that $\beta \geqslant 0$ and that $e(\mathbf{g}) > 0$. Let Q, B be real numbers satisfying the conditions of Lemma 5.3 for the given value of β. Assuming that GRAPHS$(\mathbf{g}, X) \neq \varnothing$, simulate the Markov chain $\mathscr{MC}(\mathbf{g}, X)$ as defined above. By the discussion preceding Lemma 5.3, the chain is rapidly mixing so a polynomially bounded simulation suffices to ensure a r.p.d. of at most $e(\mathbf{g})^{-\beta}/4$. But by Lemma 5.3, the stationary distribution of the chain induces a distribution over GRAPHS$(\mathbf{g}, X)$ which is almost uniform with bias at most $e(\mathbf{g})^{-\beta}/4$, since $e(\mathbf{g}) \geqslant 1$ and $\beta \geqslant 0$. The overall bias is then at most $e(\mathbf{g})^{-\beta}$, as required. Finally, again by Lemma 5.3, the stationary probability of being at a leaf is bounded below as in Lemma 4.1 except for an additional factor due to pruning of $(1 - e(\mathbf{g})^{-\beta}/4) \geqslant \frac{3}{4}$. ∎

COROLLARY 5.5. *For any fixed real β, there exists a polynomial time algorithm which generates labelled k-regular graphs on n vertices almost uniformly with bias at most $n^{-\beta}$, provided that the degree is bounded as $k = O(n^{1/3})$.*

We could of course allow the bias in the above algorithm to be specified as part of the input. However, there is no reason to suppose that the resulting generator would be fully polynomial since we can say nothing useful about the behaviour of the counter in Corollary 5.2 for "small" instances as Q and B vary. Thus the polynomial bias claimed here is apparently the best we can achieve in polynomial time. Note that the source of the bias is essentially just the pruning operation on the tree: the effect of the truncation of the Markov chain is exponentially small as in Theorem 4.4, and thus negligible by comparison.

It remains now for us to prove Lemma 5.3. For this we require a preliminary technical result.

PROPOSITION 5.6. *Let Z be a random variable denoting the number of green objects in a random sample (without replacement) of size $s > 0$ from a population of size $m \geqslant 2s$ made up of g green and $b = m - g$ blue objects, and let $\mu = \mathrm{E}(Z) = sg/m$. Then for any real $\alpha > 0$,*

$$\Pr(Z > \alpha\mu) \leqslant s\left(\frac{2e}{\alpha}\right)^{\alpha\mu}.$$

Proof. Note that Z is distributed hypergeometrically with mean $\mathrm{E}(Z) = \mu$ as claimed. Now set $r = \alpha\mu$. If $r < sg/(m-s)$ then the right-hand side of the above inequality is greater than 1 and there is nothing to prove. Assume therefore that $r \geqslant sg/(m-s)$. For each i, $1 \leqslant i \leqslant s$, the probability that the ith choice yields a green object, conditional on the preceding choices, certainly cannot exceed $g/(m-s)$, since there are always at least $m-s$ elements remaining in the pool. Thus for any $r' \in \mathbf{N}$ with $r' \geqslant r$ we have

$$\Pr(Z = r') \leqslant q_{r'} = \binom{s}{r'}\left(\frac{g}{m-s}\right)^{r'}.$$

But we have also

$$\binom{s}{r'} = \frac{s!}{r'!(s-r')!} \leqslant \frac{s^{r'}}{r'!} \leqslant \left(\frac{es}{r'}\right)^{r'},$$

by Stirling's approximation, so that

$$q_{r'} \leqslant \left(\frac{esg}{r'(m-s)}\right)^{r'}.$$

Now the function $f(x) = (c/x)^x$, with $c \in \mathbf{R}^+$, is monotonically decreasing for $c/x \leqslant e$; hence, since $r' \geqslant r \geqslant sg/(m-s)$, we have the bound

$$q_{r'} \leqslant \left(\frac{esg}{r(m-s)}\right)^r \leqslant \left(\frac{2e}{\alpha}\right)^{\alpha\mu} \qquad \forall r' \geqslant r,$$

and consequently

$$\Pr(Z > r) \leqslant \sum_{r'=\lceil r \rceil}^{s} \Pr(Z = r') \leqslant s\left(\frac{2e}{\alpha}\right)^{\alpha\mu},$$

as required. ∎

Proof of Lemma 5.3. By virtue of the asymptotic bounds on $g_{\max}$ and $x_{\max}$, we may choose $Q > 0$ such that $\max\{g_{\max}, x_{\max}\} \leqslant (Q/4)\, e(\mathbf{g})^{1/4}$ for all instances in the family. This implies a lower bound on B of $100Q^4$: further constraints on B will be introduced below. Note that all instances in the family are certainly (Q, B)-balanced.

For problem instances with $e(\mathbf{g}) \leqslant B$ there is nothing to prove as no pruning takes place in the tree $T_R(\mathbf{g}, X)$. So let $\langle \mathbf{g}, X \rangle$ be an instance in the family with $m = e(\mathbf{g}) > B$, and G be any graph in GRAPHS$(\mathbf{g}, X)$, assumed non-empty. In order to estimate the proportion of all $m!$ derivations of G present in the pruned tree $T_R^{(Q,B)}(\mathbf{g}, X)$, we estimate the probability that a *randomly chosen* derivation of G is present. More precisely, consider the random process $(\bar{H}^{(t)})_{t=0}^m$, where $\bar{H}^{(0)} = \varnothing$ and, for $t \geqslant 1$, $\bar{H}^{(t)}$ is a subgraph of G having precisely t edges which is obtained from $\bar{H}^{(t-1)}$ by adding a single edge, all unused edges of G being equiprobable. If we identify $\bar{H}^{(t)}$ with a problem instance label $\langle \mathbf{h}^{(t)}, Y^{(t)} \rangle$ in the tree of derivations as before, then a random derivation $(\bar{H}^{(t)})$ is still present after pruning iff $\langle \mathbf{h}^{(t)}, Y^{(t)} \rangle$ is (Q, B)-balanced for $0 \leqslant t \leqslant m$. The proportion of all $m!$ derivations of G which are present after pruning is therefore just

$$\Pr\left(\bigwedge_{t=0}^{m} (\langle \mathbf{h}^{(t)}, Y^{(t)} \rangle \text{ is } (Q, B)\text{-balanced})\right). \tag{17}$$

We proceed to obtain a lower bound on (17) by showing that, for each t separately, $\langle \mathbf{h}^{(t)}, Y^{(t)} \rangle$ is almost surely (Q, B)-balanced, provided we make B large enough. Clearly, this is just the event that the problem instance corresponding to a randomly chosen t-edge subgraph of G is (Q, B)-balanced. The proof divides into four stages, corresponding to various ranges of values of t.

(i) *If* $0 \leqslant t \leqslant m/2$, *then* $\Pr(\langle \mathbf{h}^{(t)}, Y^{(t)} \rangle$ *is* (Q, B)*-balanced*$) = 1$. We always have $h_{\max}^{(t)} \leqslant g_{\max}$ and $y_{\max}^{(t)} \leqslant x_{\max} + g_{\max}$, so that $\gamma(\mathbf{h}^{(t)}, Y^{(t)}) \leqslant 2\gamma(\mathbf{g}, X)$. Furthermore, for all t in this range, $e(\mathbf{h}^{(t)}) \geqslant e(\mathbf{g})/2$. From our

initial choice of Q, we conclude that $\langle \mathbf{h}^{(t)}, Y^{(t)} \rangle$ is (Q, B)-balanced for all such t.

(ii) *If* $m/2 \leqslant t \leqslant m - m^{5/8}$, *then* $\Pr(\langle \mathbf{h}^{(t)}, Y^{(t)} \rangle$ *is* (Q, B)*-balanced*$) \geqslant 1 - m^{-\beta-1}/4$. Recall that $\bar{H}^{(t)}$ can be viewed as a randomly chosen t-edge subgraph of G, or equivalently, its complement $H^{(t)}$ in G as a randomly chosen s-edge subgraph of G, where $s = m - t$. Now we have

$$\begin{aligned}\gamma(\mathbf{h}^{(t)}, Y^{(t)}) &= \max\{h^{(t)}_{\max} y^{(t)}_{\max}, {h^{(t)}_{\max}}^2\} \\ &\leqslant h^{(t)}_{\max}(x_{\max} + g_{\max}) \leqslant h^{(t)}_{\max} Qe(\mathbf{g})^{1/4}.\end{aligned}$$

Hence $\langle \mathbf{h}^{(t)}, Y^{(t)} \rangle$ will certainly be (Q, B)-balanced if $h^{(t)}_{\max} \leqslant Qe(\mathbf{h}^{(t)})^{1/2} e(\mathbf{g})^{-1/4}$, i.e., if the maximum vertex degree $h^{(t)}_{\max}$ of the random s-edge subgraph $H^{(t)}$ does not exceed $Qs^{1/2}m^{-1/4}$. We can estimate the probability of this event using Proposition 5.6 as follows: let $j \in [n]$ be any vertex with $g_j > 0$. Then if we colour green all g_j edges of G adjacent to j, and all other edges of G blue, the random variable $h^{(t)}_j$ is distributed as the number of green edges in a random sample (without replacement) of s edges of G. We are therefore in the situation of Proposition 5.6, with $Z = h^{(t)}_j$, $g = g_j$, and tail value $\alpha\mu = Qs^{1/2}m^{-1/4}$, where $\mu = sg_j/m$ is the mean of $h^{(t)}_j$. The factor α is quite large, viz.,

$$\alpha = \frac{Qm^{3/4}}{s^{1/2}g_j} \geqslant \frac{\sqrt{2}\, Qm^{1/4}}{g_{\max}} \geqslant 4\sqrt{2},$$

where we have used the facts that $s \leqslant m/2$ and $g_{\max} \leqslant (Q/4)m^{1/4}$. The tail value itself satisfies

$$\alpha\mu = \frac{Qs^{1/2}}{m^{1/4}} \geqslant Qm^{1/16},$$

since also $s \geqslant m^{5/8}$. Proposition 5.6 therefore yields

$$\Pr(h^{(t)}_j > \alpha\mu) \leqslant s\left(\frac{2e}{\alpha}\right)^{\alpha\mu} \leqslant \frac{m}{2} c^{-m^{1/16}},$$

where $c = (2\sqrt{2}/e)^Q > 1$. Thus the probability that *any* vertex degree $h^{(t)}_j$ exceeds the bound is at most $m^2 c^{-m^{1/16}}$, which is less than $m^{-\beta-1}/4$ for all $m > B$ provided B is chosen large enough.

(iii) *If* $m - m^{5/8} \leqslant t \leqslant m - B$, *then* $\Pr(\langle \mathbf{h}^{(t)}, Y^{(t)} \rangle$ *is* (Q, B)*-balanced*$) \geqslant 1 - m^{-\beta-1}/4$. As in (ii) above, let $s = m - t$ and view $H^{(t)}$ as a randomly chosen s-edge subgraph of G. In view of (i), we may assume that $s \leqslant m/2$. By definition of γ, $\langle \mathbf{h}^{(t)}, Y^{(t)} \rangle$ will be (Q, B)-balanced if $h^{(t)}_{\max}$ and $y^{(t)}_{\max}$ are each bounded above by $Qe(\mathbf{h}^{(t)})^{1/4}$. In the case of $h^{(t)}_{\max}$ we proceed via

Proposition 5.6 precisely as in (ii), only this time with tail value $\alpha\mu = Qs^{1/4}$. We find that

$$\alpha \geqslant \frac{Qm}{s^{3/4} g_{\max}} \geqslant 4m^{9/32},$$

since now $s \leqslant m^{5/8}$. Further, $\alpha\mu = Qs^{1/4} \geqslant QB^{1/4}$, so we get the tail estimate

$$\Pr(h_j^{(t)} > \alpha\mu) \leqslant s\left(\frac{e}{2m^{9/32}}\right)^{QB^{1/4}}.$$

Thus the probability that any vertex degree $h_j^{(t)}$ exceeds $Qs^{1/4}$ is at most $m^2(cm)^{-\beta'}$, where $c > 0$ is fixed and β' can be made arbitrarily large by suitable choice of B. By setting B appropriately, we can clearly make this less than $m^{-\beta-1}/8$ for all $m > B$.

A similar argument can be used to handle $y_{\max}^{(t)}$: for a vertex $j \in [n]$ with $g_j > 0$, let $\Gamma(j)$ be the set of vertices adjacent to j in G. At this point we make use of the fact (refer to the definition of problem instance labels in the tree) that $Y^{(t)}$ includes only *essential* excluded edges, i.e., edges (i, k) for which both $h_i^{(t)} > 0$ and $h_k^{(t)} > 0$. From this it is clear that

$$y_j^{(t)} \leqslant |\{i \in \Gamma(j): h_i^{(t)} > 0\}|. \tag{18}$$

Now colour green all edges of G with an endpoint in $\Gamma(j)$, and the remainder blue, and again view $H^{(t)}$ as a random sample of size s from the edge set of G. Each time a green edge is selected, it contributes at most two to the right-hand side of (18). Thus $y_j^{(t)} \leqslant 2Z$, where the random variable Z is the number of green edges in the sample, so the required tail probability may be estimated from Proposition 5.6 with $g = \sum_{i \in \Gamma(j)} g_i \leqslant g_{\max}^2$, and $\alpha\mu = Qs^{1/4}/2$. The bounds on s in this range imply

$$\alpha \geqslant \frac{Qm}{2s^{3/4} g_{\max}^2} \geqslant \frac{8}{Q} m^{1/32},$$

and $\alpha\mu \geqslant QB^{1/4}/2$, so that

$$\Pr(y_j^{(t)} > 2\alpha\mu) \leqslant \Pr(Z > \alpha\mu) \leqslant s\left(\frac{eQ}{4m^{1/32}}\right)^{QB^{1/4}/2}.$$

Exactly as above, this ensures that the probability that any vertex degree $y_j^{(t)}$ exceeds $Qs^{1/4}$ is at most $m^{-\beta-1}/8$ for all $m > B$, provided we make B large enough. Combining the bounds for $h_{\max}^{(t)}$ and $y_{\max}^{(t)}$, we arrive at (iii).

(iv) *If* $t \geqslant m - B$, *then* $\Pr(\langle \mathbf{h}^{(t)}, Y^{(t)} \rangle$ *is* (Q, B)*-balanced*$) = 1$. This is true by definition, since $e(\mathbf{h}^{(t)}) = m - t \leqslant B$.

In view of (i)–(iv), the probability of the conjunction in (17) is now easily seen to be bounded below by $1-m^{-\beta}/4$, as claimed in the lemma. ▮

We conclude our discussion of graphs with specified degrees with some remarks on the counting problem. The reduction in (Jerrum *et al.*, 1986) from approximate counting to almost uniform generation mentioned at the end of Section 4 may be viewed more generally as a means of approximating the number of leaves in a rooted tree T given an almost uniform generator for the leaves in the maximal subtree rooted at any vertex. For any such subtree S, let $L(S)$ denote the number of leaves in S. The idea is to generate leaves of T almost uniformly and compute the fraction s of the sample which belong to the subtree S rooted at some suitably chosen child of root(T): this will be a reliable estimate of the true fraction if the latter is not too small and the sample is large enough. An estimate of $L(T)$ is then obtained by recursively estimating $L(S)$ and multiplying the result by s^{-1}. The aggregate of the sample sizes required to achieve an approximation of $L(T)$ within ratio $1+\varepsilon$ with high probability is bounded by a polynomial function of ε^{-1} and the depth m and maximum degree of T, assuming the generators have bias at most about ε/m.

Now consider the situation of Theorem 5.4: can we apply the above technique to estimate the number of leaves in the pruned tree $T_R^{(Q,B)}(\mathbf{g}, X)$? Note first that an almost uniform generator for the leaves in any maximal subtree S is available since we may simulate just this portion of the Markov chain $\mathscr{MC}(\mathbf{g}, X)$, transitions out of S being censored. Moreover, the reduced chain clearly inherits the rapid mixing property, so the generator will be efficient provided only that the subtree has sufficiently many leaves. It is not hard to see that, by modifying slightly the method in (Jerrum *et al.*, 1986, Theorem 6.4) for selecting a subtree for the recursion, we can ensure that this condition always holds with high probability.

Choosing $\varepsilon = m^{-\beta}/4$ for some $\beta \in \mathbf{R}$, we therefore get a randomised approximate counter which estimates the number of leaves in $T_R^{(Q,B)}(\mathbf{g}, X)$ within ratio $1+m^{-\beta}/4$ in polynomial time. But by Lemma 5.3 this number itself approximates $m!\#\text{GRAPHS}(\mathbf{g}, X)$ within ratio $1+m^{-\beta}/2$, so we are in fact able to approximate $\#\text{GRAPHS}(\mathbf{g}, X)$ within ratio $1+m^{-\beta}$ in polynomial time for any desired $\beta \in \mathbf{R}$. We summarise this discussion in our final theorem.

THEOREM 5.7. *For any fixed real β, there exists a polynomially time-bounded randomised approximate counter for* GRAPHS *within ratio $1+e(\mathbf{g})^{-\beta}$, provided that the degrees are bounded as $\max\{g_{\max}, x_{\max}\} = O(e(\mathbf{g})^{1/4})$.*

Theorem 5.7 implies the existence of a polynomial time algorithmic method for computing the number of labelled graphs with specified degrees (assuming that these are not too large) with a relative error which is smaller than any desired power of the number of edges. The asymptotic behaviour of such a counter thus compares very favourably with available analytic estimates, such as Theorem 5.1. While this is a remarkable theoretical result, we suspect that the various powers and constants accumulated in the reductions will render the method impractical if a high degree of accuracy is required.

Finally, we should observe that the counting problem for GRAPHS is apparently hard to solve *exactly* even under the degree restrictions imposed in this section, so that the approximation approach pursued here is justified. More precisely, we can say that the problem of evaluating #GRAPHS for instances $\langle \mathbf{g}, X \rangle$ whose degrees are bounded as $\max\{g_{\max}, x_{\max}\} = O(e(\mathbf{g})^{1/4})$ is #P-complete. To see this, note first that there is a simple reduction from the well-known #P-complete problem of counting perfect matchings in a graph G, under which the excluded graph X is the complement of G and the degree sequence is $(1, 1, ..., 1)$. The #P-completeness of the restricted version follows from the fact that the former problem remains #P-complete even for very dense graphs G, specifically when G has minimum vertex degree $n - O(n^{1/4})$, as can be shown using a reduction of Broder (1986).

ACKNOWLEDGMENT

The concept of self-embeddability mentioned in Section 4 arose in discussions with Keith Edwards.

RECEIVED October 13, 1987; ACCEPTED May 24, 1988

REFERENCES

ALDOUS, D. (1981), Random walks on finite groups and rapidly mixing Markov chains, *in* "Séminaire de Probabilités XVII," Lecture Notes in Mathematics Vol. 986, pp. 243–297, Springer-Verlag, New York/Berlin.

ALDOUS, D. (1987), On the Markov chain simulation method for uniform combinatorial distributions and simulated annealing, *Prob. Engrg. Inform. Sci.* **1**, 33–46.

ALDOUS, D., AND DIACONIS, P. (1986), Shuffling cards and stopping times, *Amer. Math. Monthly* **93**, 333–348.

ALDOUS, D., AND DIACONIS, P. (1987), Strong uniform times and finite random walks, *Adv. in Appl. Math.* **8**, 69–97.

ALON, N. (1986), Eigenvalues and expanders, *Combinatorica* **6**, 83–96.

ALON, N., AND MILMAN, V. D. (1985), λ_1, isoperimetric inequalities for graphs and superconcentrators, *J. Combin. Theory Ser. B* **38**, 73–88.

BACH, E. (1983), How to generate random integers with known factorisation, *in* "Proceedings 15th ACM Symposium on Theory of Computing," pp. 184–188.

BINDER, K. (1976), Monte Carlo investigations of phase transitions and critical phenomena, *in* "Phase Transitions and Critical Phenomena" Vol. 5b (C. Domb and M. S. Green, Eds.), pp. 1–105, Academic Press, London.

BOLLOBÁS, B. (1980), A probabilistic proof of an asymptotic formula for the number of labelled regular graphs, *European J. Combin.* **1**, 311–316.

BOLLOBÁS, B. (1985), "Random Graphs," Academic Press, London.

BRODER, A. Z. (1986), How hard is it to marry at random? (On the approximation of the permanent), *in* "Proceedings, 18th ACM Symposium on Theory of Computing," pp. 50–58; see also Erratum *in* "Proceedings, 20th ACM Symposium on Theory of Computing, 1988," p. 551.

CAI, J.-Y., AND HEMACHANDRA, L. A. (1986), "Exact Counting Is as Easy as Approximate Counting," Technical Report TR 86-761, Department of Computer Science, Cornell University.

CHEEGER, J. (1970), A lower bound for the smallest eigenvalue of the Laplacian, *in* "Problems in Analysis" (R. C. Gunning, Ed.), pp. 195–199, Princeton University Press, New Jersey.

DODZIUK, J. (1984), Difference equations, isoperimetric inequality and transience of certain random walks, *Trans. Amer. Math. Soc.* **284**, 787–794.

FELLER, W. (1968), "An Introduction to Probability Theory and Its Applications" Vol. I, 3rd ed., Wiley, New York.

GAREY, M. R., AND JOHNSON, D. S. (1979), "Computers and Intractability: A Guide to the Theory of NP-Completeness," Freeman, San Francisco.

GILL, J. (1977), Computational complexity of probabilistic Turing machines, *SIAM J. Comput.* **6**, 675–695.

GUÉNOCHE, A. (1983), Random spanning tree, *J. Algorithms* **4**, 214–220.

JERRUM, M. R., AND SINCLAIR, A. J. (1988), Conductance and the rapid mixing property for Markov chains: The approximation of the permanent resolved, *in* "Proceedings, 20th ACM Symposium on Theory of Computing," pp. 235–244.

JERRUM, M. R., VALIANT, L. G., AND VAZIRANI, V. V. (1986), Random generation of combinatorial structures from a uniform distribution, *Theoret. Comput. Sci.* **43**, 169–188.

KARP, R. M., AND LUBY, M. (1983), Monte-Carlo algorithms for enumeration and reliability problems, *in* "Proceedings, 24th IEEE Symposium on Foundations of Computer Science," pp. 56–64.

KEILSON, J. (1979), "Markov Chain Models—Rarity and Exponentiality," Springer-Verlag, New York.

KIRKPATRICK, S., GELLATT, C. D., AND VECCHI, M. P. (1983), Optimisation by simulated annealing, *Science* **220**, 671–680.

KOLMOGOROV, A. N., AND FOMIN, S. V. (1970), "Introductory Real Analysis," Prentice–Hall, Englewood Cliffs, NJ.

LAWLER, G. F., AND SOKAL, A. D. (1988), Bounds on the L^2 spectrum for Markov chains and Markov processes: A generalization of Cheeger's inequality, *Trans. Amer. Math. Soc.* **309**, 557–580.

MCKAY, B. D. (1985), Asymptotics for symmetric 0–1 matrices with prescribed row sums, *Ars Combin.* **19A**, 15–25.

NIJENHUIS, A., AND WILF, H. S. (1978), "Combinatorial Algorithms," 2nd ed., Academic Press, Orlando, FL.

SCHNORR, C. P. (1976), Optimal algorithms for self-reducible problems, *in* "Proceedings, 3rd International Colloquium on Automata, Languages and Programming," pp. 322–337.

SENETA, E. (1981), "Non-negative Matrices and Markov Chains," 2nd ed., Springer-Verlag, New York.

SINCLAIR, A. J. (1988), "Randomised Algorithms for Counting and Generating Combinatorial Structures," Ph.D. thesis, University of Edinburgh.

STOCKMEYER, L. (1977), The polynomial-time hierarchy, *Theoret. Comput. Sci.* **3**, 1–22.

STOCKMEYER, L. (1983), The complexity of approximate counting, *in* "Proceedings, 15th ACM Symposium on Theory of Computing," pp. 118–126.

TINHOFER, G. (1979), On the generation of random graphs with given properties and known distribution, *Appl. Comput. Sci., Ber. Prakt. Inf.* **13**, 265–297.

VALIANT, L. G. (1979a), The complexity of computing the permanent, *Theoret. Comput. Sci.* **8**, 189–201.

VALIANT, L. G. (1979b), The complexity of enumeration and reliability problems, *SIAM J. Comput.* **8**, 410–421.

WILF, H. S. (1981), The uniform selection of free trees, *J. Algorithms* **2**, 204–207.

WORMALD, N. C. (1984), Generating random regular graphs, *J. Algorithms* **5**, 247–280.

WORMALD, N. C. (1987), Generating random unlabelled graphs, *SIAM J. Comput.* **16**, 717–727.

Printed in Belgium