The Performance of a Precedence-Based Queuing Discipline

JOHN N. TSITSIKLIS AND CHRISTOS H. PAPADIMITRIOU

Stanford University, Stanford, California

AND

PIERRE HUMBLET

Massachusetts Institute of Technology, Cambridge, Massachusetts

Abstract. A queuing system with infinitely many servers, and with the following queuing discipline is considered: For any two jobs i and j in the system, such that i arrived later than j, there is a fixed probability p that i will have to wait for j's execution to terminate before i starts executing. This queuing system is a very simple model for database concurrency control via "static" locking, as well as of parallel execution of programs consisting of several interdependent processes. The problem of determining the maximum arrival rate (as a function of p) that can be sustained before this system becomes unstable is studied. It is shown that this rate is inversely proportional to p, and close upper and lower bounds on the constant for the case of deterministic departures are found. The result suggests that the degree of multiprogramming of multiuser databases, or the level of parallelism of concurrent programs, is inversely proportional to the probability of conflict, and that the constant is small and known within a factor of 2. The technique used involves the computation of certain asymptotic parameters of a random infinite directed acyclic graph (dag) that seem of interest by themselves.

Categories and Subject Descriptors: C.4 **[Performance of Systems]**: *design studies*; D.4.8 **[Operating Systems]**: Performance—*queuing theory*; H.2.2 **[Database Management]**: Physical Design

General Terms: Design, Performance, Theory, Verification

Additional Key Words and Phrases: Database concurrency control, queuing theory, random dag, static locking, throughput

1. *Introduction*

Consider the following queuing system. Jobs arrive at a rate λ. There are infinitely many servers that can service these jobs, and we assume that the service times are independent identically distributed variables with known distribution. Ordinarily, we would assign each incoming job to a different server, and the system would not be of any particular interest. Let us assume, however, that the jobs have certain *precedence constraints* between them. In particular, for each arriving job i and each

This research was supported by an I.B.M. Faculty Development Award and by the National Science Foundation.

Authors' current addresses: J. N. Tsitsiklis and P. Humblet, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139; C. H. Papadimitriou, Department of Computer Science, Stanford University, Stanford, CA 94305.

job j in the system at the moment of i's arrival (j is either in the queue or being served), there is a probability p that i cannot start before the service of j is completed. Owing to this constraint, it may not be possible to start servicing a job, despite the availability of servers. Moreover, as the number of jobs in the queue increases, incoming jobs will have a higher probability of having to wait too; this will further increase the size of the queue, and the process is potentially unstable. The question that we would like to investigate is, under what conditions on the arrival and service time distributions, as well as on p, is the queuing system stable.

This queuing system is a very simple attempt at modeling the performance of concurrent systems with conflicts and interdependencies between the processes. One such situation is the scheduling of database transactions so as to preserve consistency [1a, 3, 4]. One common method employed in database concurrency control is *locking*. In the simplest version of locking, once a transaction arrives and requests execution, it predeclares the database entities that it must access and/or update. If there is a conflict with a previous arrival, the transaction joins a queue and waits for that transaction to complete. If the probability of conflict between any two transactions is p, and we assume that there is enough processing power in the system so that the processing times of the transactions are not affected by the degree of multiprocessing, we have the queuing system under consideration. We should note that there are more clever ways to do locking. For example, we may acquire locks in a dynamic way (with the possibility of introducing deadlocks); or, we can start a transaction each time no active transaction conflicts with it, instead of waiting for all previous arrivals that conflict with it to execute. However, we think that our simple model captures the essentials of the queuing situation arising in database concurrency control, and deserves study, especially because of its conceptual simplicity.

Another application of our model is parallel computation. Suppose that a program consists of many processes, with the following pattern of interdependency: Consider a process i and another process j written before i was. Then there is a probability p that process i needs data computed by process j. If such a pattern of dependencies exists, then the effective parallelism of the program is captured by the maximum ratio of the arrival-to-service rate (what we call *throughput*[1]) of our queuing system.

In Section 2 we present a mathematical model for this system. If the arrival and service times are both exponential, then we have a Markov process with arbitrary directed acyclic graphs (dags) as states. This makes a direct approach to this problem virtually impossible. For this reason, we focus on the special case in which service times are deterministic. Then the states of the process are finite sequences of integers, much simpler than dags. In Section 3 we study in more detail the case of deterministic service times. We also consider a related process in which jobs arrive into the system but are never served, so that the size of the queue grows to infinity. From a slightly different point of view, this latter process corresponds to forming a random graph. We show that the throughput of the original system can be compactly characterized in terms of the asymptotic expected depth of this random graph.

On the basis of the results of Section 3, we obtain in Section 4 upper and lower bounds on the throughput of the original system by exploiting certain inequalities

[1] In the queuing-theoretical literature the term "throughput" is used to describe the ratio of the arrival rate divided by the *total* service rate, for all processors. Our use is therefore slightly nonstandard. However, this deviation is totally justified, since in our model and intended applications the servers are components of a single concurrent system.

on the statistics of the associated random graph. As the probability p tends to zero, the throughput converges to infinity. However, we are able to bound the throughput, within a constant factor from the inverse of p, for arbitrarily small p. We also discuss briefly the case of exponential service times, in which the same upper bounds, but a weaker lower bound, appear to hold.

2. *The Model*

In this section we present a model of the system under study and introduce some notation. Jobs are indexed by the positive integers in order of arrival; the arrival time of job j is denoted t_j. We assume that, with probability 1, no two jobs arrive simultaneously, that is, $t_{i+1} > t_i$; also that the interarrival times $t_{i+1} - t_i$ are independent, identically distributed, with finite mean. The *arrival rate* is then defined as $\lambda = 1/E[t_{i+1} - t_i]$. For any pair (i, j) of jobs such that $i > j$, let α_{ij} be a 0–1 random variable with mean p. We assume that these random variables are independently distributed both among themselves, as well as with the sequence of interarrival times. Here, the event $\alpha_{ij} = 1$ corresponds to a precedence constraint from i to j. That is, service of job i can only start after service of job j has terminated.

At any time t, there will be a set $V(t) \subset Z^+$ of jobs that have arrived and whose service has not yet terminated. Let

$$V_0(t) = \{i \in V(t) : \text{there is no } j \in V(t) \text{ with } \alpha_{ij} = 1\}$$

be the set of presently serviced jobs. That is, we assume that there are infinitely many servers available, and that, once a job has no precedences leading to other jobs, its service starts. Thus, there are two possibilities for an arriving job i. Either

$$V(t_i) \cap \{j : \alpha_{ij} = 1\} = \varnothing,$$

in which case $i \in V_0(t_i)$ and service starts immediately, or the intersection above is nonempty, and thus i's service will start at time τ_i, where

$$\tau_i = \inf_{t \geq t_i} \{V(t) \cap \{j : \alpha_{ij} = 1\} = \varnothing\}.$$

Let s_i denote the service time of the ith job. Formally,

$$s_i = \sup_{t \geq t_i} \{i \in V(t)\} - \tau_i.$$

We assume that the service times are themselves independent, identically distributed, random variables with finite mean (without loss of generality taken to be 1), and also independent of arrival times and the precedence events. We can now pose the problem of interest in this paper:

Find necessary and sufficient conditions on the distributions of the interarrival and service times, and on p, so that the above defined queuing system is stable in an appropriate sense.

Remark. In simple queuing systems (e.g., for the G/G/1 queue) conditions for stability can be expressed only in terms of the arrival and service rates. Accordingly, we show in Section 3 that (in the case of deterministic service times) stability depends on the interarrival times distribution only through the parameter λ. However, there are reasons to believe that throughput depends on the exact form of the service time distribution, and not just its mean.

To illustrate the difficulty of this problem, let us consider briefly the case of exponential interarrival time and service time distributions. In this case it is

straightforward to obtain a Markov model for the process. In particular, the state $X(t)$ is a dag $(V(t), E(t))$, with $V(t)$ the set of jobs that have arrived but have not yet been serviced, and arc (i, j) is in E iff $\alpha_{ij} = 1$. Although the state transition probabilities between two dags are very easy to compute, they are nevertheless extremely hard to handle. Although the state space is countable, there is no convenient parameterization or ordering of its elements. There seem to be no techniques in the spirit of networks of queues [2] that are applicable. The direct approach through the state transition equations is hopeless. This motivates the study of *deterministic* service times. In this case, even though the Markov property is lost, the state of the queue admits a much simpler description, and interesting results can be obtained.

3. *The Asymptotic Depth of Random Dags*

In this section we assume that the service times s_i are deterministic and in fact equal to 1. In this case, our model can be considerably simplified. The basic observation is that the time at which an arriving job i will start service is determined to within one time unit by the precedences. In particular, the waiting time is approximately equal to the *length of the longest chain* leading from i to other jobs in $V(t_i)$.

More formally, to each arriving job i we assign an index $n(i)$ equal to

$$n(i) = \max(\{\lfloor t_i \rfloor\} \cup \{n(j): \alpha_{ij} = 1\}) + 1.$$

Intuitively, a job i is guaranteed to complete service in the interval $[n(i), n(i) + 1)$. Finally, $l(t)$ is the *depth of the queue at time t*, that is, $l(t) = \max_{t_i \le t} n(i) - \lfloor t \rfloor$. Note that $l(t) = 1$ iff $V(t) = V_0(t)$, that is, all jobs in the system are simultaneously serviced. We can now define the *renewal times* of our system. The zeroth renewal time R_0 is zero, and for all $k > 1$,

$$R_{k+1} = \inf\{t: t \ge R_k,\ l(t) \le 1, \text{ and there is a } \tau \in [R_k, t] \text{ such that } l(\tau) > 1\}.$$

Define now the system to be *transient* if $\lim_{t\to\infty} l(t) = \infty$, and the system to be *positively recurrent* if there is an A such that for all k we have $E(R_{k+1} - R_k) \le A$. Notice that, even though we do not have a Markov model, this definition is similar to the traditional classification of chains in a countable state space. Also, the two cases of this definition are not exhaustive, in that the system can be "null recurrent." We shall show, however, that this may happen only for a single threshold value of the arrival rate λ.

We are thus led to studying the statistics of longest chains in random dags. Let $G = (Z^+, E)$ be the infinite directed graph of the precedence constraints; that is, $(i, j) \in E$ if $\alpha_{ij} = 1$. We denote by $G[m, n]$ the node-induced subgraph of G with vertex set $\{m, m + 1, \ldots, n\}$. We define the *depth* of $G[m, n]$, d_{mn}, to be the length of the longest path in $G[m, n]$. We next define $\beta_n = E[d_{1n}]/n$ and $\beta^* = \liminf_{n\to\infty} \beta_n$.

Intuitively, β^* is the rate of increase of the longest chain in the queue per unit arrival. On the other hand, the depth of the queue tends to decrease by one per unit time owing to service. For this reason, it is quite natural to expect that the throughput of the system is exactly $1/\beta^*$. This is made precise in the following theorem:

THEOREM 1

(1) *The sequence* $\{\beta_n\}$ *converges to* β^* *from above.*
(2) *The limit* $\lim_{n\to\infty}(d_{mn}/n)$ *exists almost surely, and is equal to* β^* *for any m.*

(3) *If* $\lambda > 1/\beta^*$, *then the system is transient.*
(4) *If* $\lambda < 1/\beta^*$, *then the system is positive recurrent.*

PROOF

(1) Notice that, from the definition of d_{mn} it follows that, for all $a \le b \le c$, we have $d_{ab} + d_{b+1,c} \ge d_{ac}$. Therefore, $d_{1,kn} \le \sum_1^k d_{(i-1)n+1,in}$. Noting that the terms of the sum are all independent and identically distributed, we take expectations and divide by kn to obtain $\beta_{kn} \le \beta_n$. Also, it is easy to check that, with positive probability, this inequality is strict. We conclude that $\beta_n > \beta^*$ for all n.

We can similarly show a more general inequality, namely,

$$(a + b)\beta_{a+b} \le a\beta_a + b\beta_b.$$

Now fix some $\epsilon > 0$, and choose an n_0 such that $\beta_{n_0} \le \beta^* + \epsilon/2$. Then choose a k_0 such that $\beta_m/k_0 < \epsilon/2$ for all $m = 1, \ldots, n_0$. Then, for all $k \ge k_0$ and $m \le n_0$, we have, using the previous inequality,

$$\beta_{kn_0+m} \le \frac{kn_0\beta_{kn_0}}{kn_0 + m} + \frac{m\beta_m}{kn_0 + m} < \beta^* + \epsilon.$$

It follows that β_n indeed converges to β^*.

(2) Fix an integer n. Then

$$\limsup_{k\to\infty} \frac{d_{1k}}{k} = \limsup_{k\to\infty} \max_{m\le n} \frac{d_{1,kn+m}}{kn + m}$$
$$= \limsup_{k\to\infty} \frac{d_{1,kn}}{kn} \le \frac{1}{n} \limsup_{k\to\infty} \frac{1}{k} \sum_{j=0}^{k-1} d_{jn+1,(j+1)n} = \beta_n.$$

The latter equality follows from the weak law of large numbers, since the addends are independent and identically distributed random variables with means $n\beta_n$. Since this inequality holds for all n, we conclude that $\limsup_{k\to\infty}(d_{1k}/k) \le \beta^*$.

To complete the proof of part (2), we show a converse inequality. Fix an integer n. Let r_k be the largest integer less than nk that has depth in G equal to the maximum d_{nk}. For $k \ge 2$ we define m_k to be the number of nodes in the interval $[(k-1)n + 1, kn]$ that have in G paths to r_{k-1}. Notice that, by the choice of r_k, the m_k's are independent and identically distributed random variables, since they depend in the same manner on disjoint sets of α_{ij}'s. □

LEMMA 1

(*i*) $E[m_k] \ge n - 1/p^2$.
(*ii*) *For all* $k \ge 2$, $d_{1,kn} \ge d_{1,(k-1)n} + d_{(k-1)n+1,kn} - (n - m_k)$.

PROOF

(i) Let $u_1, u_2, \ldots$ be, in increasing order, all nodes with paths to r_{k-1}. Then we have that $n - m_k \le \sum_{j=0}^{\infty} (u_{j+1} - u_j - 1)$. Now notice that

$$\Pr(u_{j+1} - u_j = l) = (1 - p)^{(l-1)(j+1)}(1 - (1 - p)^{j+1}),$$

and, therefore,

$$E[u_{j+1} - u_j] = \frac{1}{1 - (1 - p)^{j+1}}.$$

So, our estimate for $n - m_k$ above yields

$$E[n - m_k] \leq \sum_{j=0}^{\infty} \left(\frac{1}{1 - (1-p)^{j+1}} - 1 \right) \leq \frac{1}{p^2}.$$

(ii) Consider the path that achieves $d_{(k-1)n+1,kn}$. Of its $d_{(k-1)n+1,kn}$ vertices, all but $n - m_k$ have paths to r_{k-1}. Thus, the last element of the chain has depth in G at least $d_{1,(k-1)n} + d_{(k-1)n+1,kn} - (n - m_k)$. □

Now, to show the inverse inequality for (2), repeated applications of Lemma 1(ii) yield $d_{1,nk} \geq \sum_{j=1}^{k} d_{(j-1)n+1,jn} - \sum_{j=2}^{k} (n - m_j)$. Then, proceeding as in the proof of the other direction,

$$\liminf_{k\to\infty} \frac{d_{1k}}{k} = \liminf_{k\to\infty} \min_{m\leq n} \frac{d_{1,kn+m}}{kn+m}$$

$$= \liminf_{k\to\infty} \frac{d_{1,kn}}{kn} \geq \frac{E[d_{1n}]}{n} - \frac{E[n - m_k]}{n} \geq \beta_n - \frac{1}{np^2}.$$

Since this inequality holds for all n, it follows that $\liminf_{k\to\infty} d_{1k}/k \geq \liminf_{k\to\infty}(\beta_n - 1/np^2) = \beta^*$.

(3) Let us assume that $\lambda > 1/\beta^*$, and let $A(t)$ be the total number of arrivals prior to time t. By part (2) of the theorem, we have that $\lim_{t\to\infty}(d_{1,A(t)}/A(t)) = \beta^*$ (almost surely), and also that $\lim_{t\to\infty}(A(t)/t) = \lambda$ (almost surely), and thus $\lim_{t\to\infty}(d_{1,A(t)}/t) = \beta^*\lambda > 1$ (almost surely). Therefore, $\lim_{t\to\infty}(d_{1,A(t)} - \lfloor t \rfloor) = \infty$ (almost surely). Notice now that the quantity in the limit is a lower bound on the depth of the queue at time t, which establishes that the system is indeed transient.

(4) To show this last part, we need a lemma.

LEMMA 2. *Suppose that* $\lambda\beta^* < 1$. *Then, there is some* $t \geq 0$ *such that* $E[d_{1,A(t)}]/t < 1$.

PROOF. Let $\epsilon > 0$ be small enough so that $\lambda(\beta^* + \epsilon) + \epsilon < 1$. Choose n_0 such that, for all $n \geq n_0$, $\beta_n \leq \beta^* + \epsilon$. Finally, choose t large enough so that $t \geq n_0$, and $\Pr(A(t) < n_0) < \epsilon$. Also note that $d_{1,A(t)} \leq A(t)$, for all t. With the above, we obtain

$$\begin{aligned} E[d_{1,A(t)}] &= E[d_{1,A(t)} \mid A(t) \leq n_0]\Pr(A(t) \leq n_0) \\ &\quad + E[E[d_{1,A(t)} \mid A(t)] \mid A(t) > n_0]\Pr(A(t) > n_0) \\ &\leq n_0\epsilon + (\beta^* + \epsilon)E[A(t) \mid A(t) > n_0]\Pr(A(t) > n_0) \\ &\leq t\epsilon + (\beta^* + \epsilon)E[A(t)] = t(\epsilon + (\beta^* + \epsilon)\lambda) < t. \end{aligned}$$ □

So, we have found a time t so that $E[d_{1,A(t)}]/t < 1$. We shall use this to show that the system is recurrent. We do this as follows. We simplify our queuing discipline, making it suboptimal, and thus less likely to be recurrent. In particular, we consider periods of length t. During each period, we service only the jobs that were left from the previous one, whereas the new arrivals are constrained to wait for those left from previous periods. This is obviously suboptimal, since it imposes an additional constraint on the problem. Then it is easy to check whether the depth x_k of the queue at the beginning of the kth period obeys the equation

$$x_{k+1} = \begin{cases} (x_k - t) + y_k & \text{if } x_k > t, \\ y_k & \text{otherwise,} \end{cases}$$

where y_k is the length of the longest chain in the arrivals during this period. These are independent and identically distributed random variables, with mean less than

t, by Lemma 2. It is easy to verify that this system is positive recurrent, that is, that the expectation of the length of runs of indices k with $x_k > t$ is finite. If this bound is K, then the expectation of the periods between two consecutive renewals in our system is at most Kt. This completes the proof of the last part of Theorem 1. □

In view of Theorem 1, the value $\lambda^* = 1/\beta^*$ will henceforth be called the *throughput* of the system. Theorem 1 says nothing about the case $\lambda = 1/\beta^*$; it could be that, for this value, the system is neither transient nor positive recurrent. Also, note that the stability of the system depends on λ alone, and not on other moments of the interarrival times. We conjecture that this holds for any service time distribution.

Theorem 1 provides a constructive way for obtaining lower bounds on the throughput of the queuing system: For any fixed n, β_n is such a bound. For *upper* bounds, however, β_n is useless. In what follows, we define a quantity that is helpful in that respect.

Let P_n denote the *path*, that is, the dag with nodes $\{1, \ldots, n\}$ and arcs $\{(i+1, 1): i = 1, \ldots, n-1\}$. We define

$$\gamma_n = \frac{E[d_{1,2n} \mid G[1, n] = P_n]}{n}.$$

Let us explain the meaning of γ_n. First, β_n is the average depth increase when n nodes are added to the dag, *given that the process started from the most favorable conditions*, namely, depth 0. As n goes to infinity, the effect of the favorable initial conditions gradually diminishes, and β_n approaches its limit β^*. In the same spirit, γ_n corresponds to the average depth increase when n nodes are added, starting from the *most unfavorable* conditions, namely, totally ordered nodes. The following result can be obtained by arguments similar to those in the proof of Theorem 1 (its proof is therefore omitted).

THEOREM 2. *The sequence* $\{\gamma_n\}$ *converges to* β^* *from below.*

COROLLARY

(1) *If* $\lambda > 1/\gamma_n$ *for some* n, *then the system is transient.*
(2) *If* $\lambda < 1/\beta_n$ *for some* n, *then the system is positive recurrent.*

4. *Upper and Lower Bounds on the Throughput*

The previous section suggests methods for estimating the throughput of our system. One could estimate β^* by simulation. More interestingly, numerical values of β_n and γ_n can be calculated for small values of n. Unfortunately, this is not a realistic approach for sufficiently small values of p—and this is the range of interest for our applications. For example, as p goes to zero, it is easy to see that β^* converges to zero. On the other hand, β_n satisfies $\beta_n \geq 1/n$. So, if we fix the value of n and evaluate β_n as a function of p, we are not going to observe the true behavior of β^* as p goes to zero. For this reason, different tools are required to capture this behavior. Before proceeding to a result of this type, let us make a few observations. It should be clear that $\beta_n \geq p$ for all n, and thus $\beta^* \geq p$ or $\lambda^* p \leq 1$. This is a fundamental limitation of the throughput. With this in mind, the exact value of $\lambda^* p$ (which is between 0 and 1) may be viewed as the *efficiency* of the system. Moreover, it is better to think in terms of $\lambda^* p$ instead of λ^*, because the former quantity remains bounded as p goes to zero (whereas the latter goes to infinity). A natural question is then whether $\lambda^* p$ is bounded away from 0 as p goes to zero. This is settled next.

THEOREM 3. *For all* $p \neq 0$, $\lambda^* p \geq 1/e + p/2$.

PROOF. Let $X_n(k)$ denote the number of nodes in $G[1, n]$ that have depth k or more (i.e., the length of the longest path starting at the node is at least k). □

LEMMA 3. *For all* n *and* k, $E[X_n(k)] \leq p^{k-1}\binom{n}{k}$, *where* $\binom{i}{j} = 0$, *for* $j > i$.

PROOF. Induction on k. It holds for $k = 1$. Suppose it holds for some k. Then

$$E[X_n(k+1)] = \sum_{i=1}^{n} E[1 - (1-p)^{X_{i-1}(k)}] \leq p \sum_{i=1}^{n} E[X_{i-1}(k)]$$

$$\leq p \sum_{i=1}^{n} p^{k-1}\binom{i-1}{k} = p^k\binom{n}{k+1}. \qquad \square$$

Now,

$$\beta_n = \frac{E[d_{1n}]}{n} \leq \frac{1}{n}\left(k \Pr(X_n(k) = 0) + n \Pr(X_n(k) > 0)\right) \leq \frac{k}{n} + p^{k-1}\binom{n}{k}.$$

Recall that the latter inequality is true for all k and n and note that

$$\binom{n}{k} \leq \frac{(n - ((k-1)/2))^k}{k^k e^{-k}}(1 + o(1)).$$

Choosing the first term to be any number larger than $p/(e^{-1} + p/2)$, we observe that the second term tends to zero as k grows. It follows that $\beta^* \leq p/(e^{-1} + p/2)$, or, equivalently, $\lambda^* p \geq e^{-1} + p/2$. □

Recall that there is a ready upper bound: $\lambda^* p \leq 1$. It is easy to derive a tighter bound, and we give a simple derivation. The basic observation is the following: With a nonzero probability, there are at least two nodes of maximum weight in $G[1, n]$. So, the probability that the next node increases the depth by more than one is bounded below by a number larger than p. The precise statement is the following:

THEOREM 4. *For all* $p \in (0, 1]$, $\lambda^* p \leq (3 - 2p)/(4 - 4p + p^2)$. (*Notice that this implies that* $\lambda^* p$ *tends to a quantity less than* 0.75 *as* p *goes to zero.*)

PROOF. We study the growth of the average depth of a family of graphs $G'[1, n]$ that are bound to have less average depth than $G[1, n]$. The reason for this is that the graph $G'[1, n]$ has smaller depth than $G[1, n]$ *for each sample path.* A graph $G'[1, n]$ has n nodes, just as $G[1, n]$, and is of one of the following two types (see Figure 1): either (a) a set of isolated nodes, together with a path starting with the nodes, say, k and l, or (b) a set of isolated nodes with two paths starting at the nodes k and l, and converging to a single path at their second node. Once we have $G'[1, n]$, we can define $G'[1, n+1]$ as follows:

(1) Suppose that $G'[1, n]$ is of type (a). Then we have three cases. (i) If $\alpha_{n+1,k} = 1$ (probability p), then node $n + 1$ is appended to the chain. The graph remains of type (a), only with a chain longer by one. (ii) If $\alpha_{n+1,k} = 0$ but $\alpha_{n+1,l} = 1$ (probability $p(1-p)$), then $n + 1$ is appended to l to form a graph of type (b). (iii) If neither holds, then $n + 1$ becomes an isolated node, and $G'[1, n+1]$ is a graph of type (a).

(2) Suppose that $G'[1, n]$ is a graph of type (b). Then, if either $\alpha_{n+1,k}$ or $\alpha_{n+1,l}$ is one, node $n + 1$ is appended to one of the paths, the end of the other path becomes an isolated node, and $G'[1, n+1]$ is of type (a). Otherwise, $n + 1$ becomes an isolated node.

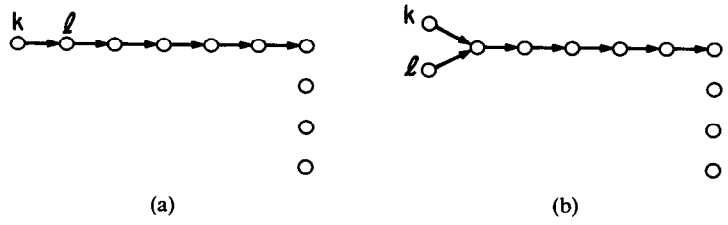

FIGURE 1

Notice that the growth of $G'[1, n]$ is a Markov process, with states (a) and (b) corresponding to the types of graphs, and with steady-state probabilities π_a and π_b satisfying

$$\pi_a = p\pi_a + (1 - p^2)\pi_a + (1 - (1 - p)^2)\pi_b.$$

Also, using the fact that $\pi_a + \pi_b = 1$, we get

$$\pi_a = \frac{2 - p}{3 - 2p}, \qquad \pi_b = \frac{1 - p}{3 - 2p}.$$

It follows that the probability that $n + 1$ increases the depth is (at steady state) $p\pi_a + (1 - (1 - p)^2)\pi_b = (4 - 4p + p^2)/(3 - 2p)p$.

This is the incremental depth of $G'[1, n]$ (call it d'_n/n). Notice that it is a lower bound for β_n. It follows that the limit of the latter quantity is bounded above by the former, and thus we obtain the theorem. □

COROLLARY. $0.37 \leq lim_{p\to 0}\lambda^* p \leq 0.75$.

By a more complicated argument we can establish that $\lim_{p\to 0}\lambda^* p \leq \log 2 \approx 0.69$; we also have experimental evidence that $\lim_{p\to 0}\lambda^* p = e^{-1}$.

Let us finally discuss briefly the case in which the service times are exponentially distributed with mean one (as opposed to deterministic and equal to one). It is possible to do an analysis of this case along very similar lines. It is not hard to show that, in this case as well, if $\lambda > 1/\beta^*$, then the system is unstable (i.e., the same upper bound holds). For lower bounds, we conjecture that a sufficient condition for the system to be positive recurrent is $\lambda < 1/ep\log(1/p)$. That is, we feel that a similar lower bound still holds, although it is weaker by a factor of $\log(1/p)$.

Note Added in Proof. Bruce Hajek has kindly communicated to us a proof that the limit of $\lambda^* p$ is actually equal to $1/e$, as p tends to zero.

The proof goes as follows: Consider a modification of our process in which an incoming job can only be blocked by the M deepest jobs in the queue. Let $\hat{\beta}(M)$ be the expected depth per unit time of the corresponding random digraph. Clearly, $\hat{\beta}(M) \leq \beta^*$.

Consider now a further modification: If an incoming job is blocked by some of the M deepest jobs (probability $1 - (1 - p)^M$), then it is attached to any of these M jobs with equal probability. Let $\beta(M)$ be the corresponding expected depth per unit time. Notice that with this modification jobs are attached to positions at smaller depth, on the average. Hence, $\beta(M) \leq \hat{\beta}(M) \leq \beta^*$.

We now use the main result of [1] to obtain $\beta(M) = s(M)[1 - (1- p)^M]$, where the sequence $s(M)$ has the property that $\lim_{M\to\infty} Ms(M) = e$. Putting everything

together,

$$\frac{1}{\lambda^* p} = \frac{\beta^*}{p} \geq \frac{s(M)}{p}\,[1 - (1-p)^M] = Ms(M)\,\frac{1-(1-p)^M}{Mp}.$$

Now let $p \to 0$ and then let $M \to \infty$. We then have $[1 - (1-p)^M]/(Mp) \to 1$, which leads to the desired result.

ACKNOWLEDGMENT. We thank the referee for reading the manuscript so carefully.

REFERENCES

1. ALDOUS, D., AND PITMAN, J. The asymptotic speed and shape of a particle system. In *Probability, Statistics and Analysis*, J. F. C. Kingman and G. E. H. Reuter, Eds. Cambridge University Press, Cambridge, England, 1983.

1a. BERNSTEIN, P. A., AND GOODMAN, N. Concurrency control in distributed database systems. *ACM Comput. Surv. 13*, 2 (June 1981), 185–221.

2. KLEINROCK, L. *Queuing Systems*, Vol. I: *Theory*. Wiley, New York, 1975.

3. PAPADIMITRIOU, C. H. The serializability of concurrent database updates. *J. ACM, 26*, 4 (Oct. 1979), 631–653.

4. ULLMAN, J. D. *Principles of Database Systems*, 2nd ed. Computer Science Press, Rockville, Md., 1983.

RECEIVED JUNE 1984; REVISED APRIL 1985; ACCEPTED OCTOBER 1985