Minimisation of the update response time in a distributed database system

Hoda W. Maalouf, Mustafa K. Gurcan

Communications and Signal Processing Group, Department of Electrical and Electronic Engineering, Imperial College of Science Technology and Medicine, Exhibition Road, London, UK SW7 2BT

Received 19 November 2001; received in revised form 28 February 2002; accepted 8 April 2002

Abstract

This paper is concerned with the minimisation of the update response time in distributed database systems (DDBS) where resequencing algorithms are used to preserve the consistency of the databases. The authors proposed a novel resequencing technique, in reference [7], namely hop-by-hop resequencing with batch processing for connecting the first stage to the second stage of a two-stage DDBS. In this paper, a stochastic upper bound is calculated for the system response time and it is shown that the system response is smaller than that for a one-stage system of the same size and parameters.

Keywords: Distributed databases; Resequencing; Batch processing

1. Introduction

The basic architecture of a distributed database system (DDBS) consists of database sites connected to each other via a communication network. At each database site, there is a computer running one or both of the software modules, namely data access software which supervises user interactions with the database, and data storage software which stores and manages the physical data at each site [1,4]. In the case of full replication, an update transaction has to be broadcast from an access site to all the storage sites of the DDBS. The update transactions to be processed on a given data item may arrive in differing orders at distinct storage sites. The integer timestamp ordering (TO) algorithm, known as the Lelann’s ticketing algorithm [6], can be used in packet switched networks where messages may arrive out of sequence. The global order is established in the following way [1,6]. The access sites are ordered in a virtual ring configuration. On this ring, M tokens circulate (one token for every data item in the database). These tokens are in charge of sequentially delivering the timestamp. When receiving the token, each access
site numbers sequentially all the updates which have accessed the site since the last token’s visit and then forwards the token to its successor in the virtual ring. Hence, each update transaction accessing the DDBS is marked with a unique integer number, its timestamp. Finally, resequencing queues are implemented on each storage site (one in front of each data item) in order to enforce this global order. TO \cite{1,4,5} is a widely used technique to ensure that, in spite of any potential network desequencing, the update transactions are processed in the same order at all storage sites.

The update response time in a DDBS is defined as the delay between the arrival of update messages at the access sites and the updating of all replicas in the distributed database, i.e. the delay between the ‘fork’ that corresponds to the time an update is split into $Q$ parts with each part going to a different storage site, and the ‘join’ that takes place when all storage sites have committed. An extensive amount of literature exists on the analysis of resequencing systems \cite{2}. However, most of it is limited to simple network models. Very few papers consider resequencing systems with fork–join synchronisation constraints. The main contributions in this domain can be found in \cite{1–3,10}.

Baccelli \cite{1} and Varma \cite{10} analysed the queuing network associated with the TO algorithm, which operates for a one-stage network as shown in Fig. 1. The number of storage sites and data items are denoted by $Q$ and $M$, respectively. The communication delays between the access site and the storage sites are modelled by $Q$ infinite server queues, while each storage site $S^q$, $1 \leq q \leq Q$ is represented by $M$ “resequencing” buffers followed by $M$ single server queues $S^q_1, \ldots, S^q_M$. A transaction is said to have concluded when each of $Q$ storage sites finishes the execution. By using ideas from queuing theory and stochastic ordering theory, Baccelli \cite{1} showed that for a one-stage distributed database network, the response time, is upper bounded by $\ln Q/p$, where $Q$ is the number of storage sites, assumed to be very large ($\rightarrow \infty$), and $p$ is a constant dependent on the system parameters.

The system analysed in \cite{1,10} is a one-stage DDBS as opposed to the two-stage DDBS investigated in this paper. The two-stage DDBS is shown in Fig. 2. It is assumed that the token passing mechanism of Lelann’s ticketing algorithm is implemented at access sites. It is further assumed that the allocation of timestamps is very fast compared to the arrival of the updates. The propagation topology for the updates involves two stages: the first stage of communication is from the access sites to $K$ gateway controllers, and the second stage is from the gateway controllers to the end-users’ $L$ storage sites. The
The proposed system considers fully replicated databases. An update is replicated to produce $K$ new updates and is immediately sent to the input queue of the communication link in order to broadcast to all gateway controllers. Update customers that arrive from the communication medium are resequenced in a resequencing buffer with respect to the TO, defined among the different access sites by the token mechanism. In the proposed system, it is assumed that a batch server will serve periodically and together the waiting in-sequence update customers. Each gateway controller will then generate new timestamps by maintaining a counter and will issue a new timestamp for every batch. This newly formed update customer is replicated to produce $L_k$ new customers, which are immediately sent to the input queue of the communication link in order to broadcast to all the end-users’ storage sites. At storage sites, servers will update customers according to a “resequencing” service discipline. Once serviced, all customers leave the queuing network and a “join” takes place when all replicas of a given data item have been updated.

This arrangement enables the resequencing delay to be controlled at the second stage, to minimise the processing time at end-users and hence reduce the overall update response time. This novel resequencing strategy is referred to as hop-by-hop resequencing with batch processing [7]. The idea of batch processing was motivated by the fact that the departure process in the resequencing buffer is a batch departure [7]. The proposed technique is adaptive by nature, since with any changes in network parameters the batch-processing period $T$ can be changed to minimise the end-to-end delay.

This paper shows that hop-by-hop resequencing with batch processing provides an improved resequencing strategy in two-stage DDBS. The paper also presents a method for calculating the update response time for a two-stage DDBS with hop-by-hop resequencing and batch processing. It is shown that the response time is smaller than that of a one-stage DDBS. Finally, the paper identifies the optimum number of gateway controllers that minimises the update response time. The paper is divided into six main sections. Section 2 provides a brief description of a two-stage DDBS in terms of a queuing network.
network model. In Section 3, an upper bound for the expectation value of the update response time in a two-stage DDBS is calculated. In Section 4, the two-stage DDBS performance is compared with that of a one-stage DDBS. The numerical results are explained in Section 5 and finally in Section 6, the main results of the paper are summarised.

2. Queuing network model

This section provides the main assumptions considered while analysing the queuing network model of a two-stage replicated database network maintained by TO algorithm. The arrival of the update transactions from every access site is assumed to be a Poisson process, then their sum is also Poisson. A single arrival time process represents the superposition of the arrival streams of update customers in each access site operating on a particular data item as shown in Fig. 2. The variable $\lambda$ will identify the global arrival rate for the update customers. The number of gateway controller storage sites and end-user storage sites will be denoted by $K$ and $L$, respectively. $L$ is equal to $L_1 + \cdots + L_k$, where $L_k$ is the number of the end-user storage sites in network $k$. The total number of stage sites, $Q$, is given by $Q = K + L$. The system of Fig. 2 can be represented using a queuing network model as shown in Fig. 3, which can be divided into two major sections: I and II each of which is sub-divided into two sections a and b.

I-a Resequencing due to an $M/M/\infty$ disordering network: the analysis of this sub-system was undertaken, as pointed out earlier, in [1]. In Section 3.1, we give a brief summary of the results for the upper-bound delay calculations.

I-b Periodic batch processor: the batch service time at the gateway controller is assumed to be negligible and will be taken equal to 0. The exact mathematical analysis of this sub-system is shown in

Fig. 3. Overall queuing network model.
reference [7] where an expression for the interdeparture time density function from the batch server is calculated. This interdeparture time is given in Appendix B.

II-a Resequencing due to a G/M/∞ disordering network: the analysis of this sub-system is undertaken in Section 3.2 in order to provide the upper-bound delay for the two-stage network.

II-b Destination server queue: the analysis of this sub-system is undertaken in Section 3.2 as part of the upper-bound resequencing delay calculations.

In the system model used, the general variables \( \sigma, \alpha, \tau \) and \( d \) identify the service time to update the data items, the communication delay, the inter-arrival time of data streams and the link delay, respectively.

Variables \( \Sigma, A, T \) and \( D \) are used to define the distribution functions for the random variables \( \sigma, \alpha, -\tau \) and \( d \), respectively. It is assumed that the communication delay \( \alpha \) has an exponential distribution with service time \( \mu \). The characteristics of the distribution functions \( \Sigma, A, T \) and variables \( \sigma, \alpha, -\tau \) are presented in Appendix A. The characteristics of the distribution functions \( D \), for stages I and II, are discussed in Appendices B and D. The system delay, \( R \), for each of the first and second stages of the network given in Fig. 3 is equal to the maximum value of link delay plus the service time. A subscript, \( n \), is used in conjunction with these variables to identify the \( n \)th update item. The superscripts, 1, 2, \( k \) and \( l \), for these variables are used to refer to stages I and II of the system model, the gateway controller number, \( k \), and also the number, \( l \), of storage sites.

The total number of data items will be denoted by \( M \) and the \( m \)th data item will be identified as \( E_m \), \( 1 \leq m \leq M \). A single arrival time process represents the superposition of the arrival streams of update customers in each access site operating on a particular data item \( E_m \).

It is assumed that the allocation of timestamps is very fast compared to the arrival of updates, which is in line with the assumptions made by Baccelli [1]. The analysis presented here will be limited to the interaction of fork, join and resequencing mechanisms occurring in the system as shown in Fig. 3. Once each update is created and timestamped, \( K \) copies of it are sent to the input queue of the communication link. Resequencing and batch processing are performed at gateway controllers. Therefore, each gateway controller storage site \( S_k \), \( 1 \leq k \leq K \) is represented by a set of \( M \) resequencing buffers and batch server queues \( S_k^1, \ldots, S_k^M \).

Once serviced, every batch is given a new timestamp. This newly formed update customer is used to produce \( L_k \) new customers that are immediately sent to the input queue of the communication link. Resequencing and batch processing are performed at gateway controllers. Therefore, each gateway controller storage site \( S_k \), \( 1 \leq k \leq K \) is represented by a set of \( M \) resequencing buffers and batch server queues \( S_k^1, \ldots, S_k^M \).

Queue \( S_k^m \), \( 1 \leq m \leq M \) will serve update customers according to a resequencing service discipline. In all the analysis that follows, the index, \( m \), will be suppressed as in our analysis and in our results an averaging process is carried out.

3. Mathematical analysis

In this section, an analysis is presented for an upper bound for the expected value of the update response time for the network model shown in Fig. 3. The terminology “first and second stage of the network” will identify the sections shown in Fig. 3 and labelled correspondingly. The delay of the \( n \)th update transaction in the \( i \)th link of the first stage will be denoted by \( R_{i,k}^n, 1 \leq k \leq K \). The variable \( R_{i,k}^{n,l} \), \( 1 \leq k \leq K \) and \( 1 \leq l \leq L_i \), will denote the delay of the \( n \)th update in the \( i \)th link of sub-network \( k \) in the second stage. Let the variable \( R_n \) denote the overall system response time of the \( n \)th update transaction, i.e. the delay between the first fork and the last join in the network for the \( n \)th update. The system response time \( R_n \)
is given by

$$R_n = \max_{1 \leq k \leq K} \left( R_{k1}^1 + R_{k1}^2 \right) \leq \max_{1 \leq k \leq K} \left( R_{k1}^1 \right) + \max_{1 \leq k \leq K} \left( R_{k1}^2 \right). \quad (1)$$

In the next two sections, the average value of $R_n$ is shown to have an upper bound given by

$$E[R_n] \leq \frac{\max(1, \ln K)}{p_1} + \frac{\ln L}{p_2} \quad (2)$$

where $K$ is the number of gateway controller storage sites, $L$ the number of end-user storage sites, and $p_1$ and $p_2$ the constants dependant on the system parameters (see also Appendices A and B).

### 3.1. Calculation of the upper bound for stage one

In the case of a two-stage network, the first stage is same as the one-stage network described in [1]. Hence, the delays experienced by update customers when traversing the first stage are as in [1] and notation will now be introduced in order to associate the one-stage analysis developed by Baccelli to the two-stage network analysed in this paper. Every data item has an associated positive real sequence $\{\alpha_k\}$, $n = 0, 1, 2, \ldots$, where $\alpha_0 = 0 < \alpha_1 < \cdots < \alpha_n < \cdots \in \mathbb{R}^+$, representing the global arrival process of update customers that operate on the data item at the various access sites. The communication delay experienced by an update customer is denoted by $\alpha_k \in \mathbb{R}^+$, and the service time required to update the item created by the $n$th update customer on site $S_k$ is denoted by $\sigma_k$. The variable $\tau_n$ is defined as the $n$th inter-arrival time of the stream $\{\alpha_k\}$, and the link delay $d_k \in \mathbb{R}^+$ is defined as the delay between the $n$th arrival date of $\alpha_k$ and the beginning of the service of this $n$th update customer in queue $S_k$. The variables $\Sigma^\tau, A^\tau, T^\tau, D^\tau_k$ are used to define the distribution functions for the random variables $\Sigma^\tau, \Sigma^\tau - \tau_n, d_k^\tau$, respectively. The variables $\Sigma^\tau, \Sigma^\tau - \tau_n$ have independent identically distributed (i.i.d.) exponential distributions with parameters $\sigma, \mu$ and $\lambda$, respectively. The characteristics of the variables and their distribution functions are given in Appendix A.

For each sub-network, $k$, the link delay, $d^\tau_{k,n+1}$, is either equal to the communication delay, $\alpha_k$, if the previous data item has already been served, or it is equal to the delay experienced by the previous data item, $d^\tau_{k,n} + \sigma^\tau$, and minus the inter-arrival time, $\tau_n$, if the previous data item is still being served when the $(n+1)$th data item arrives to the server queue. This gives the link delay $d^\tau_{k,n+1}$ as

$$d^\tau_{k,n+1} = \max(\alpha_{k,n+1} - \tau_n, d^\tau_{k,n} + \sigma^\tau - \tau_n). \quad (3)$$

The system response time $R^1_n$ for the $n$th update is equal to the maximum value of link delays plus the service times experienced by $n$th update over $k$ different links and is given by [1]:

$$R^1_n = \max_{1 \leq k \leq K} \left( d^\tau_{k,n} + \sigma^\tau \right). \quad (4)$$

As there are many links, it is necessary to calculate the maximum value for the link delays plus the service time, and to use distribution functions for the system response time calculations. For this reason, variable $d_l(R^1_n)$ is used to define the distribution function of $R^1_n$. In the analysis, the steady-state distribution functions $df(R^1_n)$ and $D^1_n$, corresponding to the distribution functions $df(R^1_n)$ and $D^1_n$, respectively, are also used. It is not straightforward to obtain the distribution function $D^1_n$ for the variable $d_l^n$ from the distribution functions of the variables given in Eq. (3) due to the statistical dependence of some of the
variables. To avoid the statistical dependence problem, a new variable \( \hat{d}_n \) and its distribution function \( \hat{D}_n \) in terms of distribution functions \( \Sigma_k, A_k, T \) and \( \hat{D}_{k-1} \) are defined as follows:

\[
\hat{D}_n = A_n(\hat{D}_{n-1} * \Sigma^k + T^k)
\]

and the steady-state distribution function is given by

\[
\hat{D}_\infty = A_\infty(\hat{D}_\infty * \Sigma^k + T^k),
\]

where \( * \) symbolises convolution. In order to obtain the steady-state distribution function, \( \hat{D}_\infty \), it is necessary to calculate the steady-state distribution function \( A_\infty \) for the communication delay experienced by an update. This is achieved by considering the Laplace–Stieltjes transform \( A_\infty(s) \), which is of the form, \( \mu/(\mu + s) \) and by letting

\[
U = \hat{D}_\infty * \Sigma^k + T^k,
\]

so that the steady-state distribution function can be expressed as \( \hat{D}_\infty = (1 - e^{-\mu t})U \). Baccelli [1] showed that this value of the distribution function, \( \hat{D}_\infty \), identifies the theoretical upper bound for the distribution function, \( D_\infty \), so that the following relationship holds for the distribution functions:

\[
D_\infty \leq \hat{D}_\infty.
\]

This relationship can be used to express the distribution function, \( f_d(R(1)) \), of the system response time in terms of the steady-state distribution function \( \hat{D}_\infty \) as

\[
f_d(R(1)) \leq \prod_{k=1}^{K} \hat{D}_\infty * \Sigma^k.
\]

The convolution given on the right-hand side of Eq. (9) is calculated using the Laplace–Stieltjes transform of the steady-state distribution function, \( \hat{D}_\infty \), of the delay using

\[
\hat{D}_\infty(s) = \frac{1}{\Sigma^k(s)T^k(s)} - \frac{s}{\mu + s}\hat{D}_\infty(\mu + s)\Sigma^k(\mu + s)T^k(\mu + s).
\]

In Appendix D, we present the mathematical steps involved in producing Eq. (10) when Eq. (6) is used as a starting point. By using Eq. (9) and showing that the first singularity \( -p_1 \) of \( \hat{D}_\infty(s) \Sigma^k(s) \) is a pole of order 1, Baccelli [1] showed that the asymptotic expression (as \( t \to \infty \)) for \( \hat{D}_\infty * \Sigma^k(t) \) is given by:

\[
\hat{D}_\infty * \Sigma^k(t)(1 - C \exp(-p_1 t)(1 + o(1))).
\]

Using Eqs. (9) and (11), Baccelli [1] obtained the expected value of the system’s response time as

\[
E[\max_{1 \leq k \leq K} R(1)] = E(R(1)) \leq \frac{\ln K}{p_1} (1 + o(1)).
\]

In order to compare the upper bound for the mean response time given by Eq. (12), simulation results were obtained using the block oriented network simulator (BONeS) package. The BONeS software packages is a commercially available graphical environment for simulation of a broad range of communication networks. BONeS uses a hierarchical block diagram modelling technique where the top layer in the hierarchy represents the network topology and the lower layers represent increasingly complex details of
the network architecture and protocols. In Fig. 4, graph 1 is that of Baccelli's [1] theoretical upper bound and graph 2 shows the simulation results. The Y-axis identifies the normalised mean response time. As can be seen, there is good agreement between the analytical derivations and simulation results.

3.2. Second stage

In a two-stage network, the variables used are similar to those in a one-stage network with some minor differences appearing in their subscripts. For example, $\alpha_k^n$ (in the first stage) represents the communication delay experienced by a customer in the $k$th link leading to the $k$th gateway controller in the first stage, while $\alpha_{k,l}^n$ (in the second stage) corresponds to the communication delay in the link between the $k$th gateway controller and the $l$th storage site in the $k$th sub-network of the second stage. In what follows, all the variables related to the second stage will have an extra $l$ subscript.

As for the case of the first stage, and in order to find an upper bound for $E[\max_{1 \leq k \leq K, \ 1 \leq l \leq L_k} R_{\infty}^{2,k,l}]$ (denoted by $E(\hat{R}_{\infty}^{2})$), it is necessary to find the inverse Laplace–Stieltjes transform of $\hat{D}_{\infty}^{2,k,l}(s)$, where $\hat{D}_{\infty}^{2,k,l}(s)$ is the Laplace–Stieltjes transform of a distribution function $\hat{D}_{\infty}^{2,k,l}(t)$ defined in terms of $\hat{\Sigma}_{\infty}^{2,k,l}$, $A_{\infty}^{2,k,l}$, and $T_{\infty}^{2,k,l}$ for the second stage of the network. The general form of $\hat{D}_{\infty}^{2,k,l}(s)$ in terms of $\hat{\Sigma}_{\infty}^{2,k,l}$, $A_{\infty}^{2,k,l}$, and $T_{\infty}^{2,k,l}$ remains the same as for the first stage, only that the form of $T_{\infty}^{2,k,l}$ changes because of the different nature of arrival process to the second stage, which has the following probability density function [7]:

$$\psi(t) = \sum_{n=1}^{\infty} a_n \delta(t - nT)$$  

for the periodic batch departure process, where $\delta(t - nT)$ is the Dirac delta function given at $t = nT$ and $a_n$ the probability that two consecutive batches are separated by $nT$ units of times. As in Eq. (10), the
Laplace–Stieltjes transform of the distribution function of the link delay is given by
\[
\hat{D}_{\infty}^{k,l,2}(s) = \frac{1}{\sum^{k,l,2}(s) - 1} \hat{D}_{\infty}^{k,l,2}(\mu + s) \Sigma^{k,l,2}(\mu + s) T^{k,l,2}(\mu + s).
\] (14)

Using the approach given in Section 3.1, the distribution function of the system response time for the second stage in terms of the steady-state distribution function is obtained by rewriting Eq. (9) for the second stage as
\[
df(R(\infty)) \leq \prod_{k,l=1}^{K,L} \hat{D}_{\infty}^{k,l,2} \ast \Sigma^{k,l,2}(t).
\] (15)

From Eq. (15), it is concluded that the distribution function, \(df(R(\infty))\), of the system response time of the second stage is smaller than or equal to a product of distribution functions \(\hat{D}_{\infty}^{k,l,2} \ast \Sigma^{k,l,2}(t)\). As each distribution function \(\hat{D}_{\infty}^{k,l,2} \ast \Sigma^{k,l,2}(t)\) has a value less than unity for all values of time \(t\), the distribution function, \(df(R(\infty))\), of the second stages system response time satisfies the relationship
\[
df(R(\infty)) \leq \max_{k,l} \hat{D}_{\infty}^{k,l,2} \ast \Sigma^{k,l,2}(t),
\] (16)

when Eq. (15) is satisfied. In Appendix B, the Laplace–Stieltjes transform, \(\hat{D}_{\infty}^{k,l,2}(s) \Sigma^{k,l,2}(s)\), of \(\hat{D}_{\infty}^{k,l,2} \ast \Sigma^{k,l,2}(t)\) is examined and, it is shown that the singularity of \(\hat{D}_{\infty}^{k,l,2}(s) \Sigma^{k,l,2}(s)\) is a pole of order \(n\) with real part \(-p_2\). Using the approaches given in references [1,2,9], it can be shown that for \(t\) approaching infinity, the inverse Laplace–Stieltjes transform of \(\hat{D}_{\infty}^{k,l,2} \ast \Sigma^{k,l,2}(t)\) gives the asymptotic value of the distribution function \(\hat{D}_{\infty}^{k,l,2} \ast \Sigma^{k,l,2}(t)\) as
\[
\hat{D}_{\infty}^{k,l,2} \ast \Sigma^{k,l,2}(t) = G(t) = 1 - C t^{n-1} e^{-p_2 t} \left(1 + o(1)\right) \text{ as } t \to \infty,
\] (17)

where \(C\) is a constant. To obtain the expected value of the system response time from Eqs. (16) and (17), we examine the asymptotic distribution function \(G(t)\) using an approach introduced by Baccelli [1]. As time goes to infinity (i.e. for \(t \to \infty\)), and assuming that \(G(t)\) satisfies the property that
\[
\lim_{t \to \infty} \frac{1 - G(ct)}{1 - G(t)} = 0, \quad \text{for all } c > 1,
\] (18)

then the expected value of the system response time satisfies the relationship (see e.g. [3])
\[
E(R(\infty)) = m_L (1 + o(1)),
\] (19)

where \(m_L\) is the value of \(t\) such that
\[
G(m_L) = 1 - \frac{1}{L}.
\] (20)

The distribution function \(G(t)\) as given in Eq. (17) does indeed satisfy (18) as
\[
\lim_{t \to \infty} \frac{1 - G(ct)}{1 - G(t)} = \lim_{t \to \infty} \frac{(ct)^{n-1} \exp(-p_2 ct)}{t^{n-1} \exp(-p_2 t)} = \lim_{t \to \infty} t^{n-1} \exp(-(c-1) p_2 t) = 0
\] (21)
for every $c > 1$. This, together with Eq. (19), leads to the inequality that

$$E\left(\max_{1 \leq k \leq K} R_{k,l}^{2,1}\right) = E(R_{\infty}^{2,1}) \leq \frac{\ln L}{p_1} (1 + o(1))$$

(22)

as proved in Appendix C.

From Eqs. (1), (12) and (22) and with approximations given in Appendix E, when $K \ll L$, $K \geq 1$ (say $K = 1, 2, 3$) and $L \to \infty$, it is possible to obtain the upper bound for the total system response time as:

$$E(R_{\infty}) \leq E(R_{\infty}^{1,1}) + E(R_{\infty}^{2,1}) \leq \max(1, \ln K) + \frac{\ln L}{p_1} + \frac{\ln L}{p_2}.$$  

(23)

Computer simulations were carried out to check the validity of the upper bound for the mean response time of a two-stage network and also establish whether Eq. (23) is valid for a small number of gateway controllers. Again the simulations were obtained with the BONEs package. In Fig. 4, graphs for a one-stage and a two-stage network are included. For both systems, the total number of gateways and storage sites is 15, i.e. $Q = K + L = 15$. For the single-stage network, graph 2 (simulation) and graph 1 (analytical upper bound) in Fig. 4 show that Baccelli’s upper bound holds for the given value of $Q = 15$. Similarly, graphs 3 and 4 confirm that the analytical upper bound for the two-stage network presented in Fig. 4 holds for the values of $K = 3$ and $L = 12$ chosen. Most importantly, for the chosen values of $Q$, $K$ and $L$ where $Q = K + L$, the normalised mean response time for the two-stage network is far lower, and hence more desirable, than that for the one-stage network. It should be noted that for the value of $K = 1$ gives the result $\max(1, \ln K)/p_1 = 1/p_1$ which is equal to the expected delay for the first stage of the network. The contribution to the overall system response time from the first stage of the two-stage network becomes negligible and the response of the second stage $\ln L/p_2$ dominates the overall delay as $L \gg K$.

The reduction in average response time for the two-stage network is obtained by ensuring that $p_2 > p_1$. This occurs due to the change in arrival process of packets from the first stage to the second stage of the two-stage network, $T^{*\rightarrow 1}(s)$, caused by the batch processing of the gateway controllers. Eq. (B.7) in Appendix B shows the derivation of $p_2$ as a function of $\Sigma^{s,1,2}(s)$ and $T^{*\rightarrow 2}(s)$, from Eq. (B.5), $T^{*\rightarrow 2}(s)$ is a function of the batch-processing period, $T$ and the average arrival rate, $\lambda$. It is expected that the value of $p_2$ should vary as $T$ and $\lambda$ are changed. Figs. 5 and 6 show the variation of the ratio $p_2/p_1$ as a function of $\lambda$ and $T$. The figures imply that for a fixed value of $T$, an increase in $\lambda$ (or vice versa) will result in an increase in the ratio $p_2/p_1$. As identified in [7], the derivations reported in this paper are valid for the range $T \geq 0.1$ and are indeterminable in the range $T < 0.1$. Thus the improvements reported here apply only to the range $T \geq 0.1$ (Fig. 6). In [7], the optimum range for $T$ is given to be $0.5 \leq T \leq 1.5$. The value of $T$ used in Fig. 4 is $T = 1$.

4. Two-stage versus one-stage

Section 3 above discusses the possibility of obtaining a reduction in the average response time in a DDBS by using the two-stage network model rather than the one-stage network model, where the $Q$ databases in the one-stage network are split into two categories, $K$ gateway controllers and $L$ database sites, in the two-stage model. As the existing system with $Q$ databases and service time $1/\mu$ is partitioned
into two stages, the service times for stage one and two satisfy $1/\mu_1 = 1/\mu_2 = 1/\mu$. The purpose of this is to reduce the overall mean response time by maintaining the ratio $p_2/p_1 > 1$ as discussed in Section 3. The question now posed is to determine the optimal way of sharing the $Q$ databases among the $K$ gateway controllers and $L$ database sites such that the average response time is minimised, as well as determining the upper bound on $K$ that guarantees the average response time is less for the two-stage network.

Using the results, that the average response time of a two-stage network is upper bounded by \[ \max(1, \ln K)/p_1 + \ln L/p_2 \] from Eq. (23), a certain range of values for $K$ and $L$ (always satisfying $Q = K + L$) are chosen such that this upper bound is less than $\ln Q/p_1$ (which is the system response upper bound for one-stage network as identified by Baccelli [1]). $p_1$ and $p_2$ are constants and $Q$ is assumed to be very large (i.e. $\to \infty$). In other words, it is possible to have a set of $K$ values, which
will satisfy the following relationship:

\[
\frac{\max(1, \ln K)}{p_1} + \frac{\ln L}{p_2} \leq \frac{\ln Q}{p_1}.
\] (24)

\(p_1\) is the same for both sides of Eq. (24) as the first stage of the two-stage network is the same as the one-stage Baccelli model. Eq. (24) implies that if the variable \(p_2\) is greater than \(p_1\), to minimise the mean response time the number of gateways, \(K\) must be kept as small and \(L\) as close to \(Q\) as possible. As \(K\) increases, the improvement of a two-stage network over a one-stage network decreases. Thus the argument from Eq. (24) implies that to minimise the mean response time, \(K\) must be kept at its minimum value.

For a two-stage network that satisfies the constraint \(Q = K + L\), \(K\) must lie in the range \(1 \leq K \leq Q\).

Hence the value of \(K\) that minimises the mean response time is \(K = 1\).

As \(K\) increases, the improvement of the two-stage network over a one-stage network decreases. There exists an upper bound on the value of \(K\) that keeps Eq. (24) satisfied. By replacing \(\max(1, \ln K)/p_1\) with \(\ln K/p_1\) in Eq. (24), this value can be obtained in terms of the ratio \(p_2/p_1\) as follows:

\[
\frac{p_2}{p_1} \ln Q > \frac{\ln K}{p_1} + \frac{\ln(Q - K)}{p_2} = \frac{\ln(Q - Q^{1/r})}{\ln Q - 1/r \ln Q}.
\] (25)

In order to find the upper bound, \(K\) can be expressed in terms of \(Q\) as \(K = Q^{1/r}\), where \(r > 1\), and Eq. (25) can be rewritten as

\[
\frac{p_2}{p_1} \ln Q > \frac{p_2}{p_1} \ln Q^{1/r} + \ln(Q - Q^{1/r}).
\] (26)

This upper bound for \(K\) exists when \(r\) satisfies the following relationship:

\[
\frac{p_2}{p_1} > \frac{\ln(Q - Q^{1/r})}{\ln Q - 1/r \ln Q}.
\] (27)

and therefore,

\[
\frac{r}{r - 1} \leq \frac{p_2}{p_1}.
\] (28)

For large values of \(Q\) (e.g. \(Q = 100\) or 1000), and small values of \(K\), the corresponding \(r\) values are greater than 2. The resultant value of \(Q^{1/r}\) becomes much smaller than \(Q\), and the ratio \(\ln(Q - Q^{1/r})/\ln Q\) becomes close to unity. Hence Eq. (28) approximates to

\[
\frac{r}{r - 1} \leq \frac{p_2}{p_1}.
\] (29)

There is a minimum value of \(r\) satisfying the inequality given in Eq. (29), all values of \(r\) greater than this minimum value of \(r\) will satisfy Eq. (29). This also means that if a value of \(K\) satisfies Eq. (24), then all smaller values of \(K\) (\(=Q^{1/r}\)) will also satisfy Eq. (24).

In order to illustrate the theoretical findings, the upper-bound response time for a one-stage network is compared with a two-stage network with 3, 2 and 1 gateway controllers in Fig. 7. It is noticed that two-stage becomes significantly better for arrival rates larger than 0.3, and that the system with 1 gateway controller gives better performance than a system with 2 or 3 gateway controllers. This confirms the previous discussions that the best improvement in response time is obtained when \(K\) is minimised, i.e., when the number of gateway controllers is equal to 1. Thus, the best improvement in mean response time is obtained when \(K = 1\).
5. Numerical results

From Eq. (29), it is identified that as $p_2/p_1$ increases, the minimum value of $r$ that satisfies (29) decreases and vice versa. This means that as $p_2/p_1$ increases, the maximum value of $K$ that satisfies Eq. (23), increases. For example if $Q$ is relatively large and $(p_2/p_1) \approx 2$, then a value of $r = 2$ (and above) is in the range required by (29), and therefore, taking $K = Q^{1/r} = Q^{1/2}$ (or smaller), would be a good choice for our purposes. Figs. 8 and 9 are plots of the main analytical equations established in the paper. The plots were obtained using the well known the Matlab analytical simulation package. Fig. 8
shows, as a function of the ratio $p_2/p_1$, the value $K_{p_2/p_1}$ below which the inequality (29) is satisfied for some large values of $Q = 1000$.

Fig. 9 shows the percentage improvement in the upper bound of the response time of a two-stage system with batch processing over that of an equal size one-stage system. For $Q = 100$ and $K \approx 100^{1/4} \approx 3$, there is almost no improvement for small values of the arrival rate (i.e. $\lambda \leq 0.3$), while the improvement is substantial for large values of the arrival rate $\lambda (\lambda \geq 0.7)$.

After having established how the two-stage network can be configured to improve the system response time, we further examined how different resequencing strategies can be used in conjunction with two-stage networks. Fig. 10 shows the results of the BONeS simulations for the following three different resequencing strategies in two-stage networks:

1. End-to-end resequencing: resequencing is performed at end-users only.
2. Hop-by-hop resequencing: both gateway controller and the end-user perform resequencing.
3. Hop-by-hop resequencing with batch processing: periodic batch processors are used after the resequencing buffers in gateway controllers. End-users process and commit the updates in batches.

As Fig. 8 shows, hop-by-hop resequencing with batch processing gives the best performance of the three resequencing strategies in two-stage networks (end-to-end resequencing, hop-by-hop resequencing and hop-by-hop resequencing with batch processing) for the following reasons:

1. Since a periodic batch server is implemented after each resequencing buffer at a gateway controller, then every batch of in-sequence updates forms a new update transaction that is broadcast from the gateway controller to all its subordinates instead of sending every update separately. This can lead to lower traffic in the second stage network and therefore, to a lower disordering among the newly formed updates and finally to lower resequencing delays at end users.
2. Since the updates in our scheme are in fact made up of several single in-sequence updates, it would only be necessary to access the disc once to get the required data, and then we allow this portion of the
These points cannot be directly related to the upper-bound analysis given by Eq. (23) as the analysis is an approximation.

6. Conclusions

In this paper, we prove that two-stage networks with batch processing give much better delay performance than one-stage networks, when the arrival rate is not very small. We also prove that hop-by-hop resequencing with batch processing gives a better performance than hop-by-hop resequencing and end-to-end resequencing for a two-stage network.

We calculate an upper bound for the average response time in a two-stage DDBS, and show that this upper bound is, for certain values of $K$ (number of gateway controllers), smaller than the upper bound found in [1] for a one-stage system. We have shown that the values of $K$ that make a two-stage network significantly better than a one-stage network are small (e.g. $K = 1, 2, 3$) and that the best improvement in a two-stage network over a one-stage network is for $K = 1$ and the percentage of improvement is given by $p_2/p_1$.

A weakness with distributed database control schemes that are based on the Lelann’s ticketing algorithm is the requirement for short-time delays for timestamp allocation. It should be noted that, in general, the assumption about the timestamp allocation process is valid in any distributed environment where access sites are geographically close to each other but far from the storage sites. Such a negligible delay could be achieved by giving the token a higher transmission priority over update transactions since the size of the token is usually small [8] in comparison with the update message.
Over the last decade, there has been significant progress in wide-area network environments such as the Internet. Instead of using a token passing arrangement to identify token numbers, and to overcome the short delay problems, one of the gateway controllers can be organised as a ticket office over the Internet and each access site can obtain a timestamp for their update customers. The Internet makes the realisation of timestamp order-based algorithms a reality, including the two-stage network proposed in this paper. The proposed scheme therefore, is an alternative to existing distributed database control algorithms.

Appendix A

This appendix presents the characteristics of the variables defined in Section 2.

1. The sequence \( \{\tau_n, \sigma_k n, \alpha_k n\}, 1 \leq k \leq K \} \), i.e. the sequence whose \( n \)th element is the \( \mathbb{R}^{MK+M} \) valued random variable, forms an ergodic sequence of non-negative and integrable random variables on the probability space. \( \mathbb{R} \) denotes the set of real numbers.

2. For all data items \( \{[-\tau_n]_{n=0}^{\infty}, [\sigma_k n]_{n=0}^{\infty}, 1 \leq k \leq K\} \), is a set of associated RV's. The sequences \( \{\tau_n\}_{n=0}^{\infty} \) is independent of the sequences \( \{\sigma_k n\}_{n=0}^{\infty}, 1 \leq k \leq K \). The sequence \( \{\sigma_k n\}_{n=0}^{\infty}, 1 \leq k \leq K \) is an independent sequence of i.i.d. RV's.

3. The condition \( E(\sigma_k n) < E(\tau_n) \) holds for all \( 1 \leq k \leq K \).

4. For all data items, the RV's \( \sigma_1 n, \ldots, \sigma_K n \) (resp. \( \alpha_1 n, \ldots, \alpha_K n \)) are identically distributed. The common distribution function \( A \) of the RV's \( \alpha_1 n, \ldots, \alpha_K n \) has a Laplace–Stieltjes transform of the form

\[
A(s) = \frac{\mu}{\mu + s}, \quad \text{Re}(s) \geq 0, \quad \mu > 0, \tag{A.1}
\]

as we assume that \( \alpha_1 n, \ldots, \alpha_K n \) have exponential distributions with parameter \( \mu \), where \( \text{Re}(s) \) denotes the real part of the complex variable \( s \).

The common distribution function \( \Sigma \) of the RV's \( \sigma_1 n, \ldots, \sigma_K n \) has a rational Laplace–Stieltjes transform \( \Sigma^*(s) \) of the form

\[
\Sigma^*(s) = \frac{\sigma}{\sigma + s} \tag{A.2}
\]

as \( \sigma_1 n, \ldots, \sigma_K n \) are assumed to have exponential distributions with parameters \( \sigma \).

Since the customers arrive according to a Poisson process with rate \( \lambda \), the common Laplace–Stieltjes transform \( T^*(s) \) of the RV's \( \tau_n \) has the form

\[
T^*(s) = \frac{\lambda}{\lambda + s}. \tag{A.3}
\]

Note that the parameters \( \mu, \sigma, \lambda \), which are assumed to be independent of the data item \( m \).

By taking \( q \) as the (non-zero) solution of the equation

\[
T^*(-s) \Sigma^*(s) = 1 \tag{A.4}
\]

i.e. of the equation

\[
\frac{\lambda}{\lambda - s} = \frac{\sigma}{\sigma + s} = 1, \tag{A.5}
\]

which is given by \( -q \) where \( q = \sigma - \lambda \) (\( q > 0 \) since \( \lambda/\sigma < 1 \)), and

\[
p_1 = \min(q, \mu). \tag{A.6}
\]
Using this definition, Baccelli [1] showed that
\[ E[\mathcal{R}_{\infty,m}] \leq \left( \frac{\ln K}{p_1} \right)(1 + o(1)), \]  
(A.7)
when \( K \) is very large (i.e. when \( K \to \infty \)). Therefore,
\[ E[\mathcal{R}_{\infty,m}] \leq \ln K \]
for large \( K \).  
(A.8)

Appendix B

As for the case of a one-stage network, and in order to find an upper bound to \( E(R_{\infty}^{(2)}) \) for the second stage, we are interested in finding the inverse Laplace–Stieltjes transform of \( \hat{D}_{\infty}^{(2)}(s) \), where \( \hat{D}_{\infty}^{(2)}(s) \) is the Laplace–Stieltjes transform of a distribution function \( \hat{\mathcal{D}}_{\infty}^{(2)}(t) = \mathcal{D}_{\infty}^{(2)}(t) \) defined in terms of \( \Sigma_{\infty}^{(2)}, \mathcal{A}_{\infty}^{(2)}, \) and \( T_{\infty}^{(2)} \). The general form of \( \hat{D}_{\infty}^{(2)}(s) \) in terms of \( \Sigma_{\infty}^{(2)}, \mathcal{A}_{\infty}^{(2)} \), and \( T_{\infty}^{(2)} \) remains the same as for the first stage since the same idea regarding the definition of \( \hat{D}_{\infty}^{(2)}(s) \) in the first stage applies in the second stage. It is only the form of \( T_{\infty}^{(2)} \) that changes because of the different nature of arrival process to the second stage. We first start by calculating \( T_{\infty}^{(2)} \).

Let batches depart in time slots of duration \( T \). Hence two non-empty batches can only be separated by an integer multiple of \( T \). Therefore, the probability density function \( \psi(t) \) that two non-empty batches are separated by time \( t \) must be a sum of Dirac delta functions concentrated at \( T, 2T, 3T, ... \). Thus \( \psi(t) \) is given by
\[ \psi(t) = a_1 \delta(t-T) + a_2 \delta(t-2T) + \cdots + a_n \delta(t-nT) + \cdots = \sum_{n=1}^{\infty} a_n \delta(t-nT), \]  
(B.1)
where \( \delta(t-iT) \) is the Dirac delta function concentrated at \( t = iT \). The Laplace–Stieltjes transform of \( T_{\infty}^{(2)}(s) \) is
\[ T_{\infty}^{(2)}(s) = \int_{-\infty}^{\infty} e^{-st} \psi(t) \, dt = \sum_{n=1}^{\infty} a_n \int_{-\infty}^{\infty} e^{-st-nT} \delta(t) \, dt. \]  
(B.2)
If we change variables to \( r = t-nT \) with \( dr = dt \), then
\[ T_{\infty}^{(2)}(s) = \sum_{n=1}^{\infty} a_n \int_{-\infty}^{\infty} e^{-sr} \delta(r) \, dr. \]  
(B.3)
Using the property of the Dirac delta function that
\[ \int_{-\infty}^{\infty} f(x) \delta(x) \, dx = f(0), \]  
(B.4)
we find that
\[ T_{\infty}^{(2)}(s) = \sum_{n=1}^{\infty} a_n \left( e^{-snT} \right) \bigg|_{t=0} = \sum_{n=1}^{\infty} a_n e^{-snT}. \]  
(B.5)
In practice, however, the sum from \( n = 1 \) to \( \infty \) is only a finite sum from \( n = 1 \) to a maximum finite value \( N \) which can be taken as large as needed to ensure that for all practical purposes the results from these two functions (one including an infinite sum, and the other a sum up to \( N \)) are almost the same (i.e. differing by an arbitrary small quantity). Therefore, we can take \( T^{-2} \) to be of the form

\[
T^{-2}(s) = \sum_{n=1}^{N} a_n e^{-nsT}.
\]  (B.6)

Now, as for the case of a one-stage network \([1]\), we calculate the inverse Laplace–Stieltjes transform of \( \hat{D}^{k,l}_\infty(s) \) in terms of \( \Sigma^{k,l,2} \), \( A^{k,l,2} \), and \( T^{\infty-2} \) (where \( A^{k,l,2} = \mu(s + \mu) \), \( \Sigma^{k,l,2}(s) = \sigma/(\sigma + s) \) and \( T^{\infty-2}(s) \) now has the form given in (B.6) with \( s \) replaced by \(-s\)).

Similarly to Eq. (D.7) we have

\[
\hat{D}^{k,l,2}_\infty(s) = \frac{1}{\Sigma^{k,l,2}(s)T^{\infty-2}(s) - 1} \frac{s}{\mu + s} \hat{D}^{k,l,2}_\infty(s + \mu) \Sigma^{k,l,2}(s + \mu) T^{\infty-2}(s + \mu).
\]  (B.7)

Therefore, to find the inverse Laplace–Stieltjes transform of \( \hat{D}^{k,l,2}_\infty(s) \) or, more precisely, the asymptotic form of this inverse transform, we need to find the singularities of \( \hat{D}^{k,l,2}_\infty(s) \) from Eq. (B.7).

These singularities of \( \hat{D}^{k,l,2}_\infty(s) \) are either those of \( T^{\infty-2}(s + \mu) \), or those of \( \Sigma^{k,l,2}(s + \mu) \), or of \( s/(s + \mu) \), or of \( 1/(\Sigma^{k,l,2}(s)T^{\infty-2}(s) - 1) \), or of \( \hat{D}^{k,l,2}_\infty(s + \mu) \). These will now be considered.

Since we are only interested in an asymptotic form for the inverse transform of \( \hat{D}^{k,l,2}_\infty(s) \), we need only to locate the singularity of \( \hat{D}^{k,l,2}_\infty(s) \) that has the largest real part, which (the real part) must be a negative real number, as \( \hat{D}^{k,l,2}_\infty(s) \) must be a distribution function for \( 0 < t < \infty \) that approaches 1 as \( t \to \infty \), and 0 as \( t \to 0 \). From (B.6), \( T^{\infty-2} \) is an entire function \([9]\) and therefore, has no singularities in the complex plane. Next, the term \( \Sigma^{k,l,2}(s + \mu) \) is examined in the full form \( \sigma/(\sigma + s + \mu) \) to identify its only singularity at \( s = -\sigma - \mu \). Also, the singularity of \( s/(s + \mu) \) is at \( s = -\mu \).

The singularities of \( \hat{D}^{k,l,2}_\infty(s + \mu) \) are exactly those of \( \hat{D}^{k,l,2}_\infty(s) \) translated by \(-\mu \). Therefore, the singularity of \( \hat{D}^{k,l,2}_\infty(s) \) that has the largest real part is certainly not among those of \( \hat{D}^{k,l,2}_\infty(s + \mu) \) since these are the same singularities of \( \hat{D}^{k,l,2}_\infty(s) \) moved a distance \( \mu \) to the left, and hence they cannot contain the singularity of \( \hat{D}^{k,l,2}_\infty(s) \) with largest real part. Thus, it is not necessary to consider the singularities of \( \hat{D}^{k,l,2}_\infty(s + \mu) \).

Finally, we find that a singularity of the remaining term \( 1/(\Sigma^{k,l,2}(s)T^{\infty-2}(s) - 1) \) in the complex plane is a singularity of the function

\[
\Sigma^{k,l,2}(s)T^{\infty-2}(s) - 1 = \frac{\sigma + s}{(-s + a_1s e^{\sigma T} + a_2s e^{2\sigma T} + \cdots + a_N s e^{N\sigma T}) - \sigma}.
\]  (B.8)

The right-hand side of (B.8) is the ratio of two entire functions (i.e. of two functions that are analytic in the entire complex plane) and therefore, from the theory of complex functions \([9]\), we know that the singularities of this function must be poles of order \( \geq 1 \) situated at the zeros of the denominator.

It now follows from preceding arguments that the singularities of \( \hat{D}^{k,l,2}_\infty(s) \) are all poles of order \( \geq 1 \), and that the set of all real parts of these singularities must be bounded from above by a (real) number that has to be negative. Otherwise, as mentioned above, if this real number were to be positive the asymptotic behaviour of \( \hat{D}^{k,l,2}_\infty \) would not be a distribution function.
Therefore, there exists a negative number \(-p_2\) (where \(p_2 > 0\)) such that the real part of any pole of \(\hat{D}_{k,l} \ast \Sigma^{k,l} \ast T\) is smaller than or equal to \(-p_2\). The term \(-p_2\) can be chosen to be the largest such number.

From Eq. (16), it is identified that the asymptotic behaviour of the term \(\hat{D}_{k,l} \ast \Sigma^{k,l}(t)\) needs to be examined. Since the singularity of \(\hat{D}_{k,l} \ast \Sigma^{k,l}(s)\) with highest real part is either \(-p_2\) or \(-\sigma\), the highest of \(-p_2\) and \(-\sigma\) is denoted again by \(-p_2\). This proves that \(\hat{D}_{k,l} \ast \Sigma^{k,l}(s)\) has a singularity, which is a pole of order \(n\) with real part \(-p_2\).

**Appendix C**

Using Eqs. (17) and (20), the distribution function \(G(t)\) at \(t = mL\) is obtained as follows:

\[
G(mL) = 1 - CmL e^{-p_2 mL(1 + o(1))} = 1 - \frac{1}{L}. \tag{C.1}
\]

Simplifications lead to

\[
CmL e^{-p_2 mL(1 + o(1))} = \frac{1}{L}. \tag{C.2}
\]

Taking logarithms of both sides gives

\[
n \ln mL - p_2 mL + \ln (1 + o(1)) = - \ln CL. \tag{C.3}
\]

As \(L \to \infty\), \(\ln (1 + o(1)) \to 0\) and \(n \ln mL\) becomes small compared to \(mL\), therefore, the relation \(-p_2 mL = -n \ln CL\), and simplifications lead to

\[
mL \approx \frac{\ln (CL)}{p_2} = \frac{\ln C}{p_2} + \frac{\ln L}{p_2}. \tag{C.4}
\]

As \(L \to \infty\), the constant term \(\ln C/p_2\) becomes negligible compared with \(\ln L/p_2\). Therefore, \(mL \approx \ln L/p_2\) as \(L \to \infty\). Hence

\[
E(R^{(2)}_{n \leq k}) \approx \frac{\ln L}{p_2(1 + o(1))}. \tag{C.5}
\]

As \(L \to \infty\) and \(R^{(2)}_{n} = \max_{1 \leq k \leq K} R^{2,k}_{n}\), then

\[
E \left[ \max_{1 \leq k \leq K} R^{2,k}_{n} \right] \leq \frac{\ln L}{p_2}. \tag{C.6}
\]

**Appendix D**

In this appendix, we present the mathematical steps involved in producing Eq. (10) when Eq. (6) is used as a starting point. Using Eq. (6), the distribution function can be written as

\[
\hat{D}_{n} = A_{n} \ast (\hat{D}_{n} \ast \Sigma^{T} \ast T^{-1}). \tag{D.1}
\]
where \( A^\infty_k \) has the form \( 1 - e^{-\mu t} \) and its Laplace–Stieltjes transform, \( A^\infty_k(s) \), is \( \mu/(\mu + s) \). If we let

\[
U = \hat{D}^\infty_k \Sigma^\infty T^-,
\]

then Eq. (D.1) can be rewritten as \( \hat{D}^\infty_k = A^\infty_k U = (1 - e^{-\mu t})U \) and the Laplace–Stieltjes transform of the distribution function is given by

\[
\hat{D}^\infty_k(s) = \int_0^\infty e^{-st}d\left[U - \int_0^\infty e^{-s(x+\mu)}U\right]dt.
\]

Simplifications lead to

\[
\hat{D}^\infty_k(s) = U(s) - U^*(\mu + s) - \frac{s}{\mu + s} U^*(\mu + s).
\]

Replacing \( U^*(s) \) by \( \hat{D}^\infty_k(s) \Sigma^\infty(s)T^- \), we find that

\[
\hat{D}^\infty_k(s) = \hat{D}^\infty_k(s) \Sigma^\infty(s)T^- - \frac{s}{\mu + s} \hat{D}^\infty_k(s) \Sigma^\infty(s + \mu s)T^-.
\]

Hence

\[
\hat{D}^\infty_k(s) = \frac{1}{\Sigma^\infty(s)T^-} - \frac{s}{\mu + s} \hat{D}^\infty_k(s + \mu s)T^-.
\]

Appendix E

Appendices A and C present the mathematical derivations for the upper bound of the mean response time for stages 1 and 2, respectively. Eqs. (A.8) and (C.6) show the approximations for the upper bounds when \( K \) and \( L \) are large, i.e. \( (K \to \infty, L \to \infty) \). Using these equations, the upper bound for the two-stage network given that \( K \) and \( L \) are large, \( (K \to \infty, L \to \infty) \), is given as

\[
E(R^\infty) \leq E(R^\infty_1) + E(R^\infty_2) \leq \left( \frac{\ln K}{p_1} + \frac{\ln L}{p_2} \right) (1 + o(1)).
\]

However, in the case where \( K \) can take small values, say 1, 2, 3 and \( L \) still remains large \( (L \to \infty, L \gg K) \), Eq. (E.3) is no longer appropriate. We include the lower bound of the first stage as \( 1/p_1 \), where from Eq. (A.6), \( p_1 \) represents the mean batch-processing service rate for the first stage. This lower bound on the mean response time of the first stage implies that delay of the first stage cannot be smaller than the
service time of this stage. Taking the mean delay for the first stage equal to \(1/p_1\) and using Eqs. (2), (A.7) and (C.5), the total delay for all values of \(K\) and for \(L\) large values of \(L\) is

\[
E(R_\infty) \leq E(R_1^\infty) + E(R_2^\infty) \leq \left( \frac{\max(1, \ln K)}{p_1} + \frac{\ln L}{p_2} \right) (1 + o(1)). \tag{E.2}
\]

Hence for \(K \ll L, K \geq 1\) (say \(K = 1, 2, 3\)) and \(L \to \infty\), the dominant term in the delay equation becomes \(\ln L/p_2\) and the expected delay can be approximated to

\[
\lim_{L \to \infty} \left( \frac{\max(1, \ln K)}{p_1} + \frac{\ln L}{p_2} \right) (1 + o(1)) = \frac{\max(1, \ln K)}{p_1} + \frac{\ln L}{p_2}. \tag{E.3}
\]

References


Hoda W. Maalouf was born in Lebanon in 1964. She received the Diploma of Eng. in Electrical Engineering from Ecole Supérieure d’Ingénieurs de Beyrouth, Lebanon, in 1987, the M.Sc. degree in Communications and Signal Processing, in 1990, and the Ph.D. degree in Electrical Engineering, in 1998, both from Imperial College of Science, Technology and Medicine, London, UK. She worked as a research assistant in DRIVE I CIDER and DRIVE II FRAME CEC sponsored projects at Imperial College between 1990 and 1995. She is currently an assistant professor and Chairperson of Computer Science Department, at Notre Dame University, Lebanon. Her research interests include performance evaluation of distributed and real-time systems, computer interconnection networks and multimedia in wireless and fixed networks. Her e-mail address is: hmaalouf@ndu.edu.lb.
Mustafa K. Gurcan obtained B.Sc. and Ph.D. degrees in Electrical and Electronic Engineering from the University of Birmingham in 1978 and 1985 respectively. Dr. Gurcan joined Philips Research Laboratories in Redhill, Surrey, United Kingdom in 1982 and was a principal scientist until June 1988. Since leaving Philips, Dr. Gurcan has been working in the department of Electrical and Electronic Engineering, Imperial College of Science, Technology and Medicine, United Kingdom as a senior lecturer and Head of the Mobile Communication Research Group. His research interests are in the areas of digital data transmission, wireless communication and network resource allocation and optimisation. Dr. Gurcan is a member of IEEE and IEE.