Modelling Complex Queuing Situations with Markov Processes

Jason Randal Thorne, ***School of IT, Charles Sturt Uni., NSW 2795, Australia****

***Abstract*–This article comments upon some new developments in the field of Queuing Theory and Markov Processes. Matrix–analytic methods are used to solve particular kinds of a Markov Processes. The level dependent quasi–birth–death process (LDQBD) is an example of a matrix-analytic model, and in this article we give the background on how queues are modelled with Markov Processes which leads to some new mathematics to overcome problems in modelling particular situations. We conclude by outlining new methods used to numerically calculate results for a LDQBD process.**

***Index Terms*–Markov Process, Level Dependent Quasi–Birth–Death, Queuing Theory, Matrix–Analytic Methods.**

I. INTRODUCTION

Some of the seminal work on matrix–analytic methods was done by Evans [1], which has been extensively developed by Neuts (see [2]) and others, who considered quasi birth–death processes (QBDs) which are a matrix extension of birth–death processes. In a QBD the state space is divided into subsets known as levels. From any level, the process can only move up or down to an adjacent level or to another state within the current level. An important feature of the QBD is that there is a homogeneous structure over the levels. That is, the behaviour of the process is essentially the same regardless of which level the process happens to be in. Two matrices, usually denoted R and G, play a major role in the general theory, and several algorithms have been proposed to evaluate them numerically. The matrices R and G are solutions of non-linear matrix equations, and most of the methods initially produced for their evaluation used a fixed point iteration method that converged linearly. A major break through in these methods was published in 1993 by Latouche and Ramaswami [3], who described a new iterative algorithm which was quadratically convergent and efficient. Based on probabilistic interpretation, it had good numerical stability characteristics. This method was called the Logarithmic Reduction algorithm. QBDs have been extended to the case where they have an inhomogeneous structure. That is, the behaviour of the process is dependent upon the particular level that the process is in. These models are called level dependent quasi–birth–death processes (LDQBDs). Recently Bright and Taylor [4] and Bright [5] adapted the Logarithmic Reduction algorithm to determine the R_k and G_k matrices for these LDQBD processes, where the k subscript indicates that these matrices are also level dependent.

This article presents the necessary theory for LDQBDs and briefly comments upon the new algorithms for calculating R_k and G_k. First we successively give examples to be modelled by a more sophisticated Markov Process and reasons why the extra sophistication is necessary. The analytical solutions to these models and what these solutions mean is then presented which is followed by a brief comment on some new techniques for their numerical solution. The paper concludes with results obtained by applying the numerical techniques to examples. These results test the accuracy of the new solutions.

II. A BIRTH–DEATH PROCESS

The birth–death process is a special case of a Markov process and encompasses a wide range of models such as the Poisson process and the M/M/1 queue. In this section we will define a birth–death process and give an example of its application.

Definition 1 *The continuous-time Markov process* $\{X(t) : t \geq 0\}$ *is a* **birth–death process** *if the only two possible transitions are* $n \to n+1$ *with birth rates* $q(n, n+1),\ n \geq 0,$ *and* $n \to n-1$ *with death rates* $q(n, n-1),\ n \geq 1$*. The standard infinitesimal generator matrix for a birth–death process is given by*

$$\mathbf{Q} = \begin{bmatrix} -q(0) & q(0,1) & 0 & 0 & \cdots \\ q(1,0) & -q(1) & q(1,2) & 0 & \cdots \\ 0 & q(2,1) & -q(2) & q(2,3) & \cdots \\ 0 & 0 & q(3,2) & -q(3) & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad (1)$$

*ISBN (1-86467-114-9)

where

$$q(j) = q(j, j-1) + q(j, j+1),\ \forall j \in S.$$

*As the process can not skip adjacent states, we say the process is "***skip free***" in the states.*

The equilibrium equations of a birth–death process are

$$\pi(j)q(j, j-1) = \pi(j-1)q(j-1, j)$$

and are satisfied by the equilibrium distribution given by

$$\pi(j) = \pi(0) \prod_{r=1}^{j} \frac{q(r-1, r)}{q(r, r-1)} \tag{2}$$

The following example illustrates a special case of the birth–death process, namely, the $M/M/1$ queue.

Example: 1 Suppose that the arrival process, that is the stream of customers arriving at a queue, forms a Poisson process of rate λ. Suppose further that there is a single server and that customers' service times are independent of each other and of the arrival process and are exponentially distributed with mean μ^{-1}. Such a queue is called the simple or M/M/1 queue, the M's indicating the memoryless (exponential) character of the inter-arrival and service times and the final digit indicating the number of servers. Let $X(t)$ be the number of customers in the system at time t. Then it follows from Definition 1 that $X(t)$ is a birth-death process with transition rates

$$q(j, j+1) = \mu,\ j \in S \tag{3}$$
$$q(j, j-1) = \lambda,\ j \in S. \tag{4}$$

The process can be represented by the *state transition diagram* in Figure 1. where the number in the boxes

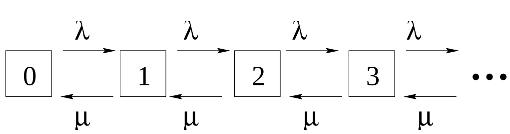

Figure 1: The State Transition Diagram

represents the possible states, and the values by the arrows represents the rate of transition between the states in the direction indicated.

This can be modelled by a birth–death process with the following infinitesimal generator matrix derived from equations (1), (3) and (4).

$$\mathbf{Q} = \begin{bmatrix} -\lambda & \lambda & 0 & \cdots \\ \mu & -(\lambda+\mu) & \lambda & \cdots \\ 0 & \mu & -(\lambda+\mu) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

A. Limitations of the Birth–Death Model

Birth–death processes work well in modelling situations that are skip free in the states, such as customers in a line or a fish population in a lake. There are, however, many situations with the Markov property where the birth–death process is inadequate. The following example was taken from Neuts [6, page 20] and is a variant of the M/G/1 queue.

Example: 2 Customers arrive at a service unit according to a Poisson process of rate λ. Services occur in groups, with the group size dependent on the queue length according to the following rule. Let there be i customers waiting at the completion of a service. If $0 \leq i < L$, the server remains idle until the queue length reaches L and then starts serving all L customers. If $L \leq i < m$, a group of size i enters service, and if $i \geq m$, a group of size m is served. It is assumed that the lengths of service of successive groups are conditionally independent.

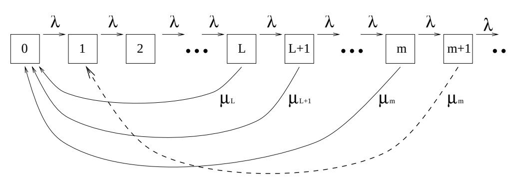

Figure 2: State Transition Diagram for Example 2

The bivariate sequence of the queue lengths following departures and the times between departures defines a Markov process. However, by examining the state transition diagram in Figure 2 it can be seen that the transitions are not skip free in the states, so this situation can not be modelled by a birth–death process (see Definition 1). In the next section we will see how the problem of modelling this situation –and many others– was overcome.

III. QUASI–BIRTH–DEATH PROCESSES

The following definition was formed from the explanation given in Latouche, Pearce and Taylor [7].

Definition 2 *A continuous time* **Quasi–Birth–Death (QBD)** *process is a continuous time Markov process whose infinitesimal generator matrix is of the block partitioned form*

$$\mathbf{Q} = \begin{bmatrix} Q_1^{(-1)} & Q_0 & 0 & 0 & \cdots \\ Q_2 & Q_1 & Q_0 & 0 & \cdots \\ 0 & Q_2 & Q_1 & Q_0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \quad (5)$$

After partitioning the states into subsets $l(i) = \{(i,j); i \geq 0,\ 1 \leq j \leq m\}$ called **levels***, position j within the level is termed the* **phase***. The process can jump down one level, stay in the same level or jump up one level and the rate that these transitions occur are given by the $m \times m$ matrices Q_2, Q_1, and Q_0 respectively. The process is said to be skip free between levels.*

The trick with modelling Example 2 is to redefine the state space so that we have levels that are skip free. This can be done by making the levels successive sets of m customers, then the possible state transitions have been transformed to that represented by Figure 1 on page 2. But describing the number of customers in the system has now become two dimensional, because, although m customers are served at a time, they arrive one at a time. This means that when there are $i.m+j$ customers in the system, it is represented by the state (i,j), where $0 \leq j \leq m-1$. Incorporating this state space with the elements of $\mathbf{Q}$ in equation (5) gives the following sub-matrices.

$$Q_1^{(-1)} = \begin{bmatrix} S_1 & 0 \\ S_2 & S_3 \end{bmatrix} \quad (6)$$

whose components are defined by

$$S_1 = \begin{bmatrix} -\lambda & \lambda & 0 & \cdots & 0 & 0 \\ 0 & -\lambda & \lambda & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -\lambda & \lambda \\ 0 & 0 & 0 & \cdots & 0 & -\lambda \end{bmatrix}, \quad (7)$$

an+ $L \times L$ matrix.

$$S_2 = \begin{bmatrix} \mu_L & 0 & \cdots & 0 \\ \mu_{L+1} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mu_m & 0 & \cdots & 0 \end{bmatrix}, \quad (8)$$

an $(m-L+1) \times L$ matrix with the only non-zero column being the first, and

$$S_3 = \begin{bmatrix} -\lambda & \lambda & 0 & \cdots & 0 & 0 \\ 0 & -\lambda & \lambda & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -\lambda & \lambda \\ 0 & 0 & 0 & \cdots & 0 & -\lambda \end{bmatrix}. \quad (9)$$

an $(m-L+1) \times (m-L+1)$ matrix. The three matrices Q_2, Q_1 and Q_0 that repeat throughout $\mathbf{Q}$ in equation (5) for this example are all $m \times m$ and are defined as follows.

$$Q_2 = \begin{bmatrix} \mu_m & 0 & \cdots & 0 \\ 0 & \mu_m & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mu_m \end{bmatrix}, \quad (10)$$

$$Q_1 = \begin{bmatrix} -\lambda & \lambda & 0 & \cdots & 0 & 0 \\ 0 & -\lambda & \lambda & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -\lambda & \lambda \\ 0 & 0 & 0 & \cdots & 0 & -\lambda \end{bmatrix}, \quad (11)$$

and Q_0 is $m \times m$ with the only non-zero entry being λ in the bottom left hand entry.

A. Limitations of the Quasi–Birth–Death Model

The Quasi–Birth–Death model effectively models the behaviour of processes that are skip free in the levels if the transition rates are independent of the level the process is in (after the first level). Such models are encountered in areas such as telecommunications and manufacturing processes. However, there are many situations, (in biological systems for instance) where the transition rates are most definitely dependent upon the level the system is in, and Quasi–Birth–Death processes become inappropriate to model the behaviour. The following simple example illustrates this point.

Example: 3 Consider a colony of algae (referred to as a population). At any given time, each member of the population can either give birth to one new member, or it can die. The temperature affects rates of birth and death within the population, and to model the affect of this factor, it has been quantified to 2 different *phases*, where 1 is cold and 2 is hot. Assuming the temperature is uniformly distributed throughout the population, the new birth and death rates for an individual are

$$\begin{aligned} \lambda_j &= \lambda(j) \\ \mu_j &= \mu(j) \end{aligned}$$

where $\lambda(j)$ and $\mu(j)$ are functions that give the birth and death rates at temperature j respectively. The other possible events that can occur are a change in temperature. We will assume that these changes occur instantaneously at rate γ_1 from cold to hot and rate γ_2 from hot to cold. All events are considered to occur separately. Two things need to be represented in the state description, namely, the number of algae present n, and the temperature factor of the environment j, so a point (n, j) in the state space $S = \{(n, j),\ n \in \mathcal{Z}^+,\ j \in \{1, 2\}\}$ represents a population size of n algae living at a temperature of j. Because each individual in the population can give birth or die, as the population size increases, the birth and death rates increase. Thus the rates for the system with population n and temperature j are given by

$$\lambda_{nj} = n\lambda(j) \qquad (12)$$
$$\mu_{nj} = n\mu(j). \qquad (13)$$

The QBD process can cope with modelling the behaviour of a population given that the transition rates change with a factor (such as temperature), as it is defined with a two dimensional state space. However, the model becomes inappropriate when the transition rate changes due to the level (in this case the number of algae present). Overcoming this problem naturally leads to the next section.

IV. LEVEL DEPENDENT QUASI–BIRTH–DEATH PROCESSES (LDQBD)

A level dependent QBD differs from a QBD in that the transition rates and the number of phases at each level can be dependent upon the level the process is in. The following definition is adapted from Bright [5, pages 5-6].

Definition 3 *A continuous time* Level Dependent Quasi–Birth–Death (LDQBD) *is a continuous time two-dimensional Markov process $X(t)$ on the state space $S = \{(k, j); k \geq 0,\ 1 \leq j \leq M(k)\}$ with infinitesimal generator of the blocked partitioned form*

$$\mathbf{Q} = \begin{bmatrix} Q_1^{(0)} & Q_0^{(0)} & 0 & 0 & \cdots \\ Q_2^{(1)} & Q_1^{(1)} & Q_0^{(1)} & 0 & \cdots \\ 0 & Q_2^{(2)} & Q_1^{(2)} & Q_0^{(2)} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \qquad (14)$$

where $Q_0^{(k)},\ k \geq 0$, $Q_1^{(k)},\ k \geq 0$, $Q_2^{(k)},\ k \geq 1$ are matrices of order $M(k) \times M(k+1)$, $M(k) \times M(k)$ and $M(k) \times M(k-1)$ and give the rates of going up one level, staying in the same level or going down one level respectively. We say the process is skip free in the levels.

A LDQBD can model the situation in Example 3 using the infinitesimal generator matrix given in equation (14) with the population size defined as levels, and the temperature as phases within each level. It should be noted here that it is assumed that the process starts with at least one member in the population (ie; in level 1), as this is necessary for the population to propagate. In the following equations the subscript indicates the temperature phase. Thus for $n \geq 1$

$$Q_2^{(n)} = \begin{bmatrix} n\mu(1) & 0 \\ 0 & n\mu(2) \end{bmatrix},$$

$$Q_1^{(n)} = \begin{bmatrix} -(n(\mu(1) + \lambda(1)) + \gamma_1) & \gamma_1 \\ \gamma_2 & -(n(\mu(2) + \lambda(2)) + \gamma_2) \end{bmatrix}$$

and

$$Q_0^{(n)} = \begin{bmatrix} n\lambda(1) & 0 \\ 0 & n\lambda(2) \end{bmatrix}.$$

Also $Q_i^{(0)} = \mathbf{0}$ for $i = 0, 1, 2$, since, if the population dies out it will not recover. Thus level zero is called *absorbing* as the process will remain in this level if it is entered.

V. DETERMINING RESULTS FOR A LDQBD

Meaningful results can be obtained from this mathematical model in terms of two matricies called G_k and R_k. An interpretation of these matricies and how they are used is outlined in the following.

A. The G_k Matrix

The G_k matrix for a LDQBD gives the first passage probabilities downward to each level from the immediately higher level (see Ramaswami [8, page 29]). The (i, j)-th element of the matrix G_k is the probability that, starting in the state (k, i), the process reaches level $k-1$ in finite time, and does so first through the state $(k-1, j)$. An example of a sample path that the probability $[G_k]_{ij}$ includes is given in Figure 3.

B. An Expression for G_k

The matrix G_k can be expressed as the minimal nonnegative solutions to a family of non-linear matrix equations. Ramaswami and Taylor [9, page 158] prove that for a continuous time LDQBD the family of matrices $\{G_k,\ k \geq 1\}$ are the minimal nonnegative solution to the matrix equation

$$Q_2^{(k)} + Q_1^{(k)} G_k + Q_0^{(k)} G_{k+1} G_k = 0,\ k \geq 1 \qquad (15)$$

where $Q_i^{(k)},\ i = 0, 1, 2$ are the sub-matrices of the transition matrix as given in Definition 3.

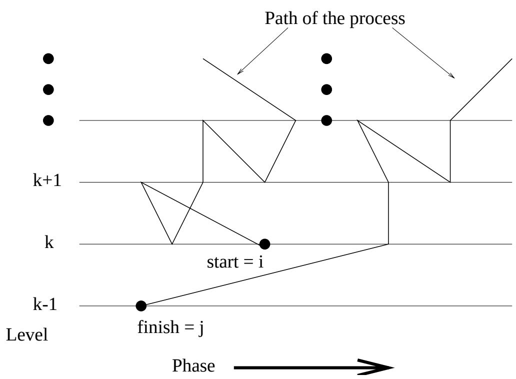

Figure 3: The Process Path Attributed to $[G_k]_{i,j}$

C. The R_k Matrix

The family of matrices R_k is used to find the stationary distribution of a LDQBD. The physical interpretation of this matrix is briefly outlined in the following. For a more detailed description of R_k, see Ramaswami [8, Section 3] and Bright [5, Section 2.4].

Theorem 1 *For a positive recurrent continuous time LDQBD,*

$$\pi_{\mathbf{k}} = \pi_{\mathbf{k-1}} R_{k-1}, \; k \geq 1 \tag{16}$$

where $\pi_{\mathbf{k}}$ is the vector of stationary probabilities of being in one of the phases in state k, and $[R_{k-1}]_{ij}$ is the expected **sojourn time** *in the state (k, j), in units of the mean sojourn time in the state $(k-1, i)$, before returning to level $k-1$, given the process starts in the state $(k-1, i)$.*

D. An Expression for R_k

The family of R_k matrices satisfy a non-linear matrix equation. Ramaswami [8, Section 3, pages 18 and 21] discusses $R_k(s)$ for a continuous time Markov process with a 2 dimensional state space that is skip free upwards in the levels, where $R_k(s)$ is the generating function form of R_k. Generalising Ramaswami's results for a LDQBD process, $R_k(s)$ is given by

$$R_k(s) = Q_0^{(k-1)} \int_0^\infty e^{-st}{}_{k-1}P(k, k : t)\, dt, \; k \geq 1, \tag{17}$$

and satisfy the equations

$$sR_k(s) = Q_0^{(k-1)} + R_k(s) Q_1^{(k+1)} + R_k(s) R_{k+1}(s) Q_2^{(k+1)}. \tag{18}$$

Now, taking s to be zero we get that R_k satisfies

$$Q_0^{(k)} + R_k Q_1^{(k+1)} + R_k [R_{(k+1)} Q_2^{(k+2)}] = 0, \; k \geq 0. \tag{19}$$

R_k is the minimal non-negative solution to the above equation, whose solutions are not necessarily unique.

E. The Relationship Between R_k and G_k

In a continuous time LDQBD, the elements of R_k and G_k both give a measure referring to a return to a level while only visiting higher levels in between. The important difference to note between the two matrices is that $[R_k]_{ij}$ is a relative time and $[G_k]_{ij}$ is a probability. There is a relationship between these two matrices however which is proved in Bright [5, page 21].

Theorem 2 *The family of matrices $\{G_k, \; k \geq 1\}$ and $\{R_k, \; k \geq 0\}$ satisfy the following equations;*

$$G_k = (-Q_1^{(k)} - R_k Q_2^{(k+1)})^{-1} Q_2^{(k)}, \tag{20}$$

$$R_k = Q_0^{(k)} (-Q_1^{(k+1)} - Q_0^{(k+1)} G_{k+2})^{-1}. \tag{21}$$

VI. THE NUMERICAL SOLUTION

Latouche and Ramaswami [3, 1993] presented a major break-through in matrix–analytic methods when they devised the Logarithmic Reduction algorithm to calculate the R and G matricies for a QBD process. This algorithm converged faster than previous algorithms and had good numerical stability characteristics, and recently Bright and Taylor [4] adapted this Logarithmic reduction algorithm to calculate R_k and G_k for a LDQBD (see Bright [5, Chapter 3] for details on these algorithms). Thorne [10] discovered that these algorithms used memory exponentially with the complexity of the problem, and constructed a new recursive algorithm to improve the memory usage of this algorithm. The following shows the results of applying these new algorithms to examples and tests to show their accuracy.

A. Testing the Results

Once the matrices G_k and R_k were determined numerically, a way of validating them was needed. R_k is the minimal non-negative solution to the system of equations (19). Rearranging equation (19) gives

$$R_k = (-Q_0^{(k)})[Q_1^{(k+1)} + R_{k+1} Q_2^{(k+2)}]^{-1}. \tag{22}$$

Thus we can find the value of R_k from R_{k+1}. Similarly, by rearranging equation (15) we can get the following expression for G_k in terms of G_{k+1}.

$$G_k = [Q_1^{(k)} + Q_0^{(k)} G_{k+1}]^{-1} (-Q_2^{(k)}). \tag{23}$$

Because the solutions to these equations are not unique, the relation between R_k and G_k given by equation (20) was used to further verify results.

B. Results

The code for the new algorithm was tested with the LDQBD defined for the algae population in Example 3. If the population of algae is 100, the probability that the population of 99 algae will ever recur needs to be known. The matrix G_{100} can answer this question as $[G_{100}]_{ij}$ gives the probability that starting with a population of 100 at temperature i the population eventually drops down to 99 algae at a temperature j. The matrix R_{100} is used to validate G_{100}. For this model, temperature is factored into 7 different phases and the necessary parameters were defined as follows.

- $\lambda = 1$ for the individual birth rate.
- $\mu = 2$ for the individual death rate.
- $\gamma = 0.5$ for all changes in temperature.

The following output was obtained.

```
The matrix R at k = 100.
0.49275908 0.00459541 0.00011914 0.00000578 0.00000041 0.00000004 0.00000000
0.00229770 0.48598516 0.00660393 0.00021886 0.00001271 0.00000103 0.00000011
0.00003971 0.00440262 0.48176169 0.00852845 0.00034093 0.00002289 0.00000229
0.00000144 0.00010943 0.00639634 0.47772890 0.01034189 0.00047998 0.00003943
0.00000008 0.00000508 0.00020456 0.00827351 0.47386768 0.01205987 0.00067848
0.00000001 0.00000034 0.00001145 0.00031998 0.01004990 0.47020617 0.01438280
0.00000000 0.00000003 0.00000098 0.00002253 0.00048463 0.01232811 0.47999486

The matrix R at k = 100 working backwards from R 101
0.49275908 0.00459541 0.00011914 0.00000578 0.00000041 0.00000004 0.00000000
0.00229770 0.48598507 0.00660393 0.00021886 0.00001271 0.00000103 0.00000011
0.00003971 0.00440262 0.48176166 0.00852845 0.00034093 0.00002289 0.00000229
0.00000144 0.00010943 0.00639634 0.47772893 0.01034189 0.00047998 0.00003943
0.00000008 0.00000508 0.00020456 0.00827351 0.47386762 0.01205987 0.00067848
0.00000001 0.00000034 0.00001145 0.00031998 0.01004989 0.47020617 0.01438280
0.00000000 0.00000003 0.00000098 0.00002253 0.00048463 0.01232812 0.47999495

The difference between matrices is 8.94e-08.

The matrix G at k = 100 from R at k = 100
0.99523067 0.00468449 0.00008168 0.00000300 0.00000017 0.00000001 0.00000000
0.00936897 0.98142225 0.00897258 0.00022491 0.00001053 0.00000072 0.00000007
0.00024503 0.01345887 0.97281790 0.01303239 0.00042019 0.00002369 0.00000205
0.00001198 0.00044981 0.01737653 0.96460491 0.01685312 0.00065700 0.00004663
0.00000085 0.00002633 0.00070032 0.02106640 0.95674372 0.02046732 0.00099511
0.00000008 0.00000215 0.00004738 0.00098550 0.02456078 0.94929302 0.02511121
0.00000001 0.00000024 0.00000478 0.00008161 0.00139315 0.02929641 0.96922386

The matrix G at k = 100
and took      0.000 seconds to calculate.
0.99523085 0.00468449 0.00008168 0.00000300 0.00000017 0.00000001 0.00000000
0.00936898 0.98142219 0.00897258 0.00022491 0.00001053 0.00000072 0.00000007
0.00024503 0.01345887 0.97281790 0.01303239 0.00042019 0.00002369 0.00000205
0.00001198 0.00044981 0.01737652 0.96460479 0.01685311 0.00065700 0.00004663
0.00000085 0.00002633 0.00070032 0.02106640 0.95674378 0.02046732 0.00099511
0.00000008 0.00000215 0.00004738 0.00098550 0.02456078 0.94929308 0.02511121
0.00000001 0.00000024 0.00000478 0.00008161 0.00139315 0.02929641 0.96922386

The matrix calculated from G at k = 101 is
0.99523067 0.00468449 0.00008168 0.00000300 0.00000017 0.00000001 0.00000000
0.00936898 0.98142225 0.00897258 0.00022491 0.00001053 0.00000072 0.00000007
0.00024503 0.01345887 0.97281790 0.01303239 0.00042019 0.00002369 0.00000205
0.00001198 0.00044981 0.01737653 0.96460491 0.01685312 0.00065700 0.00004663
0.00000085 0.00002633 0.00070032 0.02106639 0.95674366 0.02046731 0.00099511
0.00000008 0.00000215 0.00004738 0.00098550 0.02456078 0.94929296 0.02511121
0.00000001 0.00000024 0.00000478 0.00008161 0.00139315 0.02929641 0.96922386

The difference is 1.79e-07.
****END OF REPORT****
```

As can be seen from the output, the matrices generated by the improved memory efficient algorithm are consistent with those calculated by equations (22) and (20). The G_{100} matrix implies that the probability of a return to a population of 99 is very high concentrated around the concurrent event that the temperature is the same as the initial temperature.

VII. Conclusion

This project has been concerned with the presentation of R_k and G_k and implementation of an adaptation of the Logarithmic Reduction algorithm (Bright [5]) to evaluate them in a LDQBD process. The results suggest that the more memory efficient version of these algorithms implemented by Thorne [10] are accurate, though further analysis needs to be done to show how much memory they save.

Acknowledgment

My sincerest thanks go to Nigel Bean for his attentive regard over my efforts and infinite patience in guiding my honours thesis. Thanks also to Peter Taylor, whose common sense, practicality and industry inspired me to do my best.

References

[1] R Evans. Geometric distribution in some two dimentional queuing systems. *Operations Research*, 15:830–846, 1967.

[2] M Neuts. *Matrix–Geometric Solutions in Stochastic Models.* The John Hopkins University Press: ISBN 0-8018-2560-1, 1981.

[3] G Latouche and V Ramaswami. A logarithmic reduction algorithm for quasi-birth-death processes. *J.Appl.Prob*, 30:650–674, 1993.

[4] L Bright and P Taylor. Calculating the equilibrium distribution in level dependent quasi-birth-and-death processes. *Stochastic Models*, 11(3):497–525, 1995.

[5] L. Bright. *Matrix Analytical Methods in Applied Probability.* PhD thesis, The University of Adelaide, 5005, South Australia, 1996.

[6] M Neuts. Solving queues without rouché's theorem. Technical Report 76/2, Department of Statistics and Computer Science, 1976.

[7] G Latouche, C Pearce, and P Taylor. Invariant measures for quasi-birth-and-death processes. *Stochastic Models*, 1997.

[8] V Ramaswami. Matrix analytic methods: A tutorial overview with some extensions and newresults. In Michigan Flint, editor, *Proceedings of the First International Conference on Matrix Analytic Methods in Stochastic Models*, Aug 1995.

[9] V Ramaswami and P Taylor. Some properties of the rate operators in level dependent quasi–births–and–death processes with a countable number of phases. *Stochastic Models*, 12(1):143–164, 1996.

[10] J. Thorne. An investigation of algorithms for calculating the fundamental matrices in level dependent quasi–birth–death processes. Honours Thesis, The University of Adelaide, 5005, South Australia, 1997.