Calculation of yeast prion curing curves by numerical transform inversion

M. S. Ridout*

Institute of Mathematics, Statistics and Actuarial Science
University of Kent, Canterbury
Kent CT2 7NF U.K.

August 22, 2006

SUMMARY: If the chemical guanidine hydrochloride is added to a dividing culture of yeast cells that carry a yeast prion, the proportion of cells that have the prion gradually decreases over time. Stochastic models to describe this process of ‘curing’ have been developed in earlier work. The present paper investigates the use of numerical Laplace inversion methods to calculate curing curves and contrasts this with current, more direct, approaches. Transform inversion is found to provide an efficient computational approach that allows different models to be investigated with a minimal amount of programming effort. Implementations are available in Matlab and R.

KEY WORDS: age-dependent branching process; fast Fourier transform; Laplace transform; renewal process; Saccharomyces cerevisiae

*email: M.S.Ridout@kent.ac.uk
1 Introduction

Cells of the yeast *Saccharomyces cerevisiae* may be classified as \([PSI^+]\) or \([psi^−]\) according to whether or not they carry the prion form of the protein Sup35p. \([PSI^+]\) cells normally give rise to \([PSI^+]\) offspring so that a population in which all cells are \([PSI^+]\) will remain so indefinitely. However, if the chemical guanidine hydrochloride is added to a culture of \([PSI^+]\) cells, the proportion of cells in the population that are \([PSI^+]\) decreases gradually over time. This process is called *curing*.

Morgan, Ridout & Ruddock (2003; subsequently MRR03) propose a model for curing, developing ideas in Eaglestone *et al.* (2000). Specifically, they derive an expression for the probability, \(p_+(t)\), that a randomly selected cell at time \(t\) is \([PSI^+]\) and use this as the basis of a binomial likelihood to analyse data from curing experiments.

Calculation of \(p_+(t)\) for a given set of model parameters involves a non-trivial computational effort. MRR03 approach the computation directly, but note that numerical transform inversion would provide an alternative approach. The purpose of the present note is to explore this possibility further.

The model of MRR03 treats the cell division process as one of simple binary fission. In fact, *S. cerevisiae* reproduces by an asymmetric budding process. Cole *et al.* (2004; subsequently CMRBT04) extend the model to allow for this more complex method of reproduction. We also investigate numerical transform inversion for this extended model.

The paper is organised as follows. Section 2 outlines the model of MRR03
and Section 3 discusses two approaches to computation, direct calculation and numerical inversion of Laplace transforms. Some details of the transform inversion method used are given in Section 4. Section 5 discusses the extended model of CMRBT04. Sections 6 and 7 present results and conclusions respectively.

2 Model

The model of MMR03 assumes that the population consists entirely of $[PSI^+]$ cells initially, with the number of prions per cell varying as a Poisson variable with expectation $n_0$. Curing is driven by cell division, which is modelled as an age-dependent branching process, in which each cell is assumed to live for a random period of time, termed the *generation time*, at the end of which it is replaced by two new cells. The prions in the original cell are divided randomly between the two offspring cells. It is assumed that all generation times are independent.

We let $f(x)$ denote the probability density function (p.d.f.) of generation time, $X$, and let $F(x)$ denote its distribution function. We also let the random variable $H(t)$ denote the number of renewals in an equilibrium renewal process with inter-occurrence time p.d.f. $f(x)$. MRR03 show that

$$ p_+(t) = \frac{\sum_{h=0}^{\infty} \left\{ 1 - \exp \left( -n_0 2^{-h} \right) \right\} 2^h p_h(t)}{\sum_{h=0}^{\infty} 2^h p_h(t)}, \quad (1) $$

where

$$ p_h(t) = \Pr \{ H(t) = h \}. $$

Strictly, $p_+(t)$ is the expectation of the ratio of the number of $[PSI^+]$ cells
at time $t$ to the total number of cells at time $t$ whereas the right hand side
of equation (1) is the ratio of the corresponding expectations. Thus, the de-
nominator is the expected population size at time $t$, given a single initial cell,
and the numerator is the expected number of $[PSI^+]$ cells at time $t$ (MRR03).
However, particularly when there are a large number of cells initially, as there
are in curing experiments, this approximation is extremely accurate and we
proceed in this paper as if equation (1) is indeed an exact expression. For
a similar discrete-time model this approximation has been shown to be very
accurate even when the initial population size is small (Piau, 2004).

3 Numerical computation

Our main interest is in evaluating equation (1) by numerical transform in-
version, but first, for comparison, we briefly describe the direct computation
approach used by MRR03.

3.1 Direct evaluation

To evaluate equation (1) directly, we first replace the infinite upper limit of the
summations by a finite value, $h_{\text{max}}$ such that $\Pr \{ H(t) > h_{\text{max}} \}$ is negligible
for all $t$ of interest. We discuss the choice of $h_{\text{max}}$ in Section 4.

We then need to evaluate $p_h(t)$ for $0 \leq h \leq h_{\text{max}}$. It is useful to condition on
the time until the first event, $T_1$, which, for an equilibrium process, has p.d.f.

\[ w(t_1) = \frac{1 - F(t_1)}{\mu}, \]
where $\mu = \text{E}(X)$, Cox (1962, §5.2). Then
\[
p_0(t) = \Pr \{ T_1 > t \} = 1 - \int_0^t w(t_1)dt_1.
\]
For $h > 0$ we may note that, conditional on $T_1 = t_1$, the probability of $h$ renewals in $[0, t]$ is the same as the probability of $h - 1$ renewals in an ordinary renewal process in $(t_1, t]$. Thus
\[
p_h(t|t_1) = F_{h-1}(t - t_1) - F_h(t - t_1),
\]
where $F_r(t)$ is the distribution function of the sum of $r$ independent random variables with p.d.f. $f(x)$ (Cox, 1962, §3.1). Consequently,
\[
p_h(t) = \int_0^t p_h(t|t_1)w(t_1)dt_1.
\]
MRR03 evaluated this integral numerically using the trapezium rule, assuming that $X$ had a gamma distribution with p.d.f.
\[
f(x) = \frac{\beta^\alpha x^{\alpha-1}e^{-x/\beta}}{\Gamma(\alpha)}.
\]
For this distribution, computation of $F_r(t)$ involves the incomplete gamma function.

### 3.2 Laplace transform method

We use the notation $g^*(s)$ to denote the Laplace transform of a function $g(x)$. Let $G(t, z)$ denote the probability generating function (p.g.f.) of $H(t)$. Then
\[
G^*(s, z) = \frac{1}{s} + \frac{z - 1}{\mu s^2} \left[ \frac{1 - f^*(s)}{1 - zf^*(s)} \right]
\]
(Cox, 1962; §3.2), from which it is easily shown by induction that
\[
p_h^*(t) = \begin{cases} \frac{1}{s} \frac{1 - f^*(s)}{\mu s^2} & h = 0, \\ \frac{1 - f^*(s)}{\mu s^2} \frac{f^*(s)^h}{(1 - f^*(s))^2} & h > 0. \end{cases}
\]
Let $A(t)$ and $B(t)$ denote the numerator and denominator of equation (1), respectively, with Laplace transforms $A^*(s)$ and $B^*(s)$. Clearly, $B(t) = G(t, 2)$ and hence

$$B^*(s) = \frac{1}{s} + \frac{1}{\mu s^2} \left[\frac{1 - f^*(s)}{1 - 2f^*(s)}\right].$$

(5)

MRR03 give an explicit expression for $B^*(s)$ when $X$ has a gamma distribution. However, there is no simple expression for the Laplace transform of the numerator and therefore, to avoid bias in $p_+(t)$, we approximate the Laplace transforms of both the numerator and the denominator by finite sums with the same upper limit, specifically

$$A^*(s) \approx \left(\frac{1}{s} - \frac{1 - f^*(s)}{\mu s^2}\right) (1 - e^{-n_0}) + \frac{1}{\mu s^2} \sum_{h=1}^{h_{max}} \{f^*(s)\}^{h-1} \{1 - \exp \left(-n_0 2^{-h}\right)\},$$

(6)

and

$$B^*(s) \approx \left(\frac{1}{s} - \frac{1 - f^*(s)}{\mu s^2}\right) \frac{1 - f^*(s)}{\mu s^2} \sum_{h=1}^{h_{max}} 2^h \{f^*(s)\}^{h-1}.$$  

(7)

We may then invert $A^*(s)$ and $B^*(s)$ numerically and calculate $p_+(t)$ as the ratio $A(t)/B(t)$. We now discuss the details of the numerical inversion.

4 Numerical transform inversion

Several numerically stable algorithms exist for inverting transforms of probability density and distribution functions and these are being used increasingly in applied probability (Abate, Choudhury & Whitt, 1999).

Here, although we wish to calculate a probability, this is done via the functions $A(t)$ and $B(t)$. The function $A(t)$ is bounded, but $B(t)$ increases exponentially
with \( t \). Thus, direct inversion \( B^*(s) \) can give poor results for large \( t \). This problem can be alleviated by re-scaling (Choudhury & Whitt, 1997). For the present problem, there is a simple re-scaling that follows from the fact that \( B(t) \) is the expected number of cells at time \( t \) if there is a single initial cell (MRR03). Hence, from the theory of age-dependent branching processes

\[
A(t) \sim A e^{\theta t} \quad (t \to \infty)
\]

under mild regularity conditions, where \( \theta \) is the Malthusian parameter, the positive root of the equation

\[
f^*(s) = 1/2 \tag{8}
\]

(e.g. Harris, 1963; Chapter 6). Thus, instead of inverting the transforms of \( A(t) \) and \( B(t) \) we invert the transforms of \( e^{-\theta t} A(t) \) and \( e^{-\theta t} B(t) \), which are \( A^*(s + \theta) \) and \( B^*(s + \theta) \), respectively.

From amongst the many algorithms for numerical inversion, we have selected the algorithm of den Iseger (2006), because it is extremely accurate, is easy to code and generates values on a regular grid of time points efficiently. This latter feature is convenient for curing experiments, where samples are usually collected at regular intervals. The algorithm belongs to the class of Fourier series methods (Dubner & Abate, 1968) and uses the fast Fourier transform.

We shall not describe the details of the algorithm here, but instead refer to sections 1–4 of den Iseger (2006). Appendix F of Tijms (2003) also provides a useful overview. The algorithm can be extended to the inversion of transforms of functions that have discontinuities and singularities (den Iseger, 2006), but we shall not need this more elaborate version of the algorithm here.
We coded the algorithm in Section 4 of den Iseger (2006) in both Matlab and R. The inputs to the algorithm, as we have implemented it, are $t_{\text{max}}$, the maximum time at which the function is to be evaluated, and a positive integer $M$; the output is a grid of function values for $M$ equally spaced times ranging from 0 to $t_{\text{max}}$ inclusive. For most curing experiments, appropriate choices are $t_{\text{max}} = 31$ (hours) and $M = 32$, which evaluates the curing curve at hourly intervals. Because the algorithm uses the fast Fourier transform, it is generally beneficial to choose $M$ to be a power of two. However, the loss of efficiency for other choices is not great, since much of the computing effort lies in the calculation of transform values. Other internal parameters of the inversion routine were set to the defaults recommended by Den Iseger (2006).

To improve efficiency, calculations in Matlab and R are vectorised where possible. In particular, the routine to calculate the Laplace transform $f^*(s)$ returns a vector of transform values when $s$ is a vector.

To compute $A^*(s)$ and $B^*(s)$ using equations (6) and (7), we need to set $h_{\text{max}}$. To do this, we use the fact that for an equilibrium renewal process, the number of renewals up to time $t$ is distributed asymptotically as $N(\mu/t, \sigma^2 t/\mu^3)$ (Cox, 1962, §3.3). Therefore a suitable setting for $h_{\text{max}}$ is

$$h_{\text{max}} = \text{int} \left( \frac{\mu}{t_{\text{max}}} + c\sigma \sqrt{\frac{t_{\text{max}}}{\mu^3}} \right), \quad (9)$$

where int(.) denotes the integer part and $c$ is chosen so that, based on the asymptotic distribution, $\Pr \{H(t_{\text{max}}) > h_{\text{max}}\}$ is small.

We calculate $\mu = E(X)$ and $\sigma^2 = \text{var}(X)$ by numerical differentiation and the Malthusian parameter $\theta$ is obtained by solving equation (8) numerically, using the Matlab function \texttt{fzero} or the R function \texttt{uniroot}. Alternatively, of
course, distribution-specific code could be supplied for calculating $\mu$, $\sigma^2$ and $\theta$. However, the current code has the advantage that the only change required to modify the distribution of $X$ is to alter the routine for calculating $f^*(s)$.

5 An extended model

CMRBT04 describe a more complicated model that allows for the asymmetrical reproduction of $S$. cerevisiae. At the end of its lifetime, each cell gives rise to a mother cell and a daughter cell. These have different generation time distributions. For a mother cell, the p.d.f. of generation time is denoted by $f_M(x)$. Daughter cells require an additional period of time to develop, with p.d.f. $f_D(x)$, before they can begin reproduction. This initial development period is assumed to be independent of the subsequent reproductive period and the overall generation time is therefore the convolution of two variables with p.d.f.s $f_M(x)$ and $f_D(x)$. Strictly, this is no longer an age-dependent branching process, but it remains true that the population size is asymptotically of the form $Ae^{\theta t}$, where $\theta$ is now the positive root of the equation

$$f_M^*(s) \{1 + f_D^*(s)\} = 1,$$

(Green, 1981; CMRBT04).

As in the original model, we assume that the system is in equilibrium initially, because curing experiments usually involve established and growing cell cultures. In equilibrium, the proportion of cells that are daughter cells is $P = f_M^*(\theta)$ and the Laplace transform of the p.d.f. of the waiting time until
the first cell division is
\[ w^*(s) = \frac{(1 - P) \{1 - f^*_M(s)\}}{s \mu_M} + P \{1 - f^*_M(s)f^*_D(s)\} \frac{\{1 - f^*_M(s)f^*_D(s)\}}{s (\mu_M + \mu_D)}, \] (11)

where \( \mu_M \) and \( \mu_D \) are the expectations of random variables with p.d.f.s \( f_M(x) \) and \( f_D(x) \) respectively.

Finally, at cell division, prions present in the parent cell are assumed to segregate independently between the mother and daughter cells, with each prion having probability \( \pi \) of passing to the daughter cell.

Under these assumptions, CMRBT04 show that
\[ p_+(t) = \sum_{h=0}^{\infty} \sum_{d=0}^{h} Q_{h,d}(t) \left\{ 1 - \exp \left[ -n_0 \pi^d(1 - \pi)^{h-d} \right] \right\} \frac{\{1 - f^*_M(s)f^*_D(s)\}}{\sum_{h=0}^{\infty} \sum_{d=0}^{h} Q_{h,d}(t)}, \] (12)

where
\[ Q^*_{0,0}(s) = \frac{1 - w^*(s)}{s}, \]
\[ Q^*_{1,d}(s) = \frac{w^*(s)}{s} \{1 - f^*_M(s)f^*_D(s)^d\} \quad (d = 0, 1), \]

and more generally, for \( h > 1 \) and \( 0 \leq d \leq h \),
\[ Q^*_{h,d}(s) = \frac{w^*(s)f^*_M(s)^{h-1}f^*_D(s)^{d-1}}{s} \left[ \begin{pmatrix} h-1 \\ d-1 \end{pmatrix} f^*_D(s)\{1 - f^*_M(s)\} + \begin{pmatrix} h-1 \\ d-1 \end{pmatrix} \{1 - f^*_D(s)\} \right], \]

with the convention that \( \begin{pmatrix} h-1 \\ d-1 \end{pmatrix} = 0 \) if \( d = 0 \). The extended model reduces to the original MRR03 model if \( \pi = 1/2 \) and if the development time for daughters is a degenerate random variable taking the value zero with probability one, implying that \( f^*_D(s) \equiv 1 \).

Whilst this model is considerably more complicated than the original model, there are no new conceptual issues in evaluating \( p_+(t) \) by numerical transform inversion. As previously, the infinite summations involved are truncated at
\( h = h_{\text{max}} \). The choice of \( h_{\text{max}} \) can be based conservatively on the distribution of generation time for mother cells, since these are the faster reproducing cells. This is discussed further in the next Section. The main computational challenge is to evaluate the transforms \( Q_{h,d}^*(s) \) efficiently.

6 Results

We first evaluate the performance of the transform inversion method for the MRR03 model, assuming that \( X \) has a gamma distribution, with \( f^*(s) = (1 + s/\beta)^{-\alpha} \). Explicit expressions for \( p_h(t) \) are available when \( \alpha \) is an integer (Cox, 1962; §3.2), enabling \( p_+(t) \) to be evaluated without the need for numerical integration. We evaluated \( p_+(t) \) for integer time points \( t = 0, 1, \ldots, 31 \) for different values of \( \alpha \). The mean generation time \( (\alpha/\beta) \) was fixed at 2 hours. The coefficient of variation of the distribution is \( 1/\sqrt{\alpha} \). The parameter \( n_0 \) was set to 400 (see below).

For transform inversion, we used our program for the more general CMRBT04 model, varying the parameter \( c \), which determines \( h_{\text{max}} \) according to equation (9). We calculated the maximum absolute discrepancy between values of \( p_+(t) \) calculated by transform inversion and the corresponding values calculated directly from equation (1) when the summations were truncated either at the same value of \( h_{\text{max}} \) (MaxErr1) or at a large value of \( h_{\text{max}} = 100 \), for which the directly calculated values were essentially exact (MaxErr2). MaxErr1 is therefore a measure of the accuracy of the transform inversion, whereas MaxErr2 is a measure of the adequacy of the choice of \( c \).
Table 1 gives some numerical results. The transform inversion is extremely accurate, agreeing with the findings of den Iseger (2006). For many purposes, calculation of \( p_+ (t) \) need only be accurate to 3 or 4 decimal places, and for this it appears that choosing \( c = 3 \) will be adequate.

To obtain more guidance on the choice of \( c \), we consider the CMRBT04 model with the mother cell generation time having a gamma distribution with parameters \( \alpha_M \) and \( \beta \) and the additional development time for daughter cells having a gamma distribution with parameters \( \alpha_D \) and \( \beta \). The common value of \( \beta \) means that the distribution of total generation time for daughter cells is also gamma, with parameters \( \alpha_M + \alpha_D \) and \( \beta \), which greatly simplifies direct calculation.

Typical parameter values, based on detailed observation of individual cells using timelapse photography are \( \alpha_M = 30, \alpha_D = 15 \) and \( \beta = 15 \). We kept \( n_0 = 400 \) and set \( \pi = 0.35 \); these are typical estimates from experimental data (L.J. Byrne, personal communication).

Since no exact results are available in this case, we compared the Laplace transform inversion routine for various values of \( c \) with inversion results for \( h_{max} = 50 \). Table 2 shows the results. As expected, the maximum discrepancy decreased as \( c \) increased, but the discrepancies are quite small even with \( c = 0 \).

This is because \( h_{max} \) is determined from the distribution of the number of cell divisions for cells that have no daughter cells in their ancestry. This represents only a very small fraction of the total cell population. The remaining cells in the population are expected to have resulted from fewer cell divisions. With \( c = 0 \), the calculations took approximately half a second in Matlab on
a 3.19 GHz PC with 1 GB of RAM. This is certainly fast enough to make iterative parameter estimation practical; parameter estimation is described in CMRBT04. On the same computer, calculations in R were slower by a factor of about ???.

We also confirmed that the transform inversion method gave very similar results to an existing program based on direct evaluation of equation (12) (Cole, personal communication). Maximum discrepancies between the two approaches were of the order of $10^{-5}$. Useful timing comparisons are not possible, because no attempt has been made to optimise the code for direct evaluation. However, the direct approach is likely to be slower than transform inversion for comparable accuracy, because calculation of each term $Q_{h,d}(t)$ in the direct approach requires numerical integration.

7 Conclusions

Laplace transform inversion offers a fast numerical approach to evaluation of $p_+(t)$. It is also more flexible than direct evaluation, which requires the distribution function of the generation time for daughter cells. In CMRBT04, this is assumed to be a convolution of two gamma variables with common parameter $\beta$, so that evaluation of the distribution function requires a single call to the incomplete gamma function. If the values of $\beta$ differed for the two distributions, evaluation of the distribution function would require much more computational effort. On the other hand, the Laplace transform inversion method places no restrictions on the two distributions, which need not even be from the same family. For example, generation time for mother cells could
have a gamma distribution whilst development time for daughter cells had an inverse Gaussian distribution.

The algorithm of den Iseger (2006) is suited well to this application, because experimental data are usually collected at equally spaced time intervals. Statistical analysis for curing experiments uses data from several sources (CMRBT04), with a binomial likelihood based on $p_+(t)$ providing information about the parameters of key biological interest, $n_0$ and $\pi$. The current algorithms are sufficiently fast for parameter estimation to be practical, even though they are written in Matlab or R, with the advantages of relatively easy maintenance and modification.

Like many inversion algorithms, the den Iseger algorithm requires the transform to be evaluated for complex arguments. Both Matlab and R have complex number facilities and we did not encounter any problems in this respect, though it should be noted that some special functions in these languages do not accept complex arguments. Although one might prefer an algorithm that only required the transform to be evaluated for real arguments, the best-known such algorithm, the Gaver-Stehfest algorithm, requires high precision arithmetic (Abate & Valko, 2004).

The Matlab and R programs are available at [new website]

Acknowledgements

This work forms part of BBSRC grant 96/E18382.
References


Table 1: Performance of numerical transform inversion for the MRR03 model. The probability $p_+(t)$ was calculated at time points $t = 0, 1, \ldots, 31$ by three methods: (a) transform inversion with $h_{\text{max}}$ determined by the value of $c$ given in the Table, (b) direct evaluation with the same value of $h_{\text{max}}$ and (c) direct evaluation with $h_{\text{max}} = 100$. MaxErr1 and MaxErr2 are the maximum absolute discrepancies between the values calculated by methods (a) and (b) and by methods (a) and (c) respectively.

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$c$</th>
<th>$h_{\text{max}}$</th>
<th>MaxErr1</th>
<th>MaxErr2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>28</td>
<td>1.59E-13</td>
<td>4.89E-04</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>36</td>
<td>1.11E-13</td>
<td>1.99E-05</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>44</td>
<td>1.64E-13</td>
<td>7.62E-07</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>22</td>
<td>4.05E-14</td>
<td>1.15E-04</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>26</td>
<td>4.18E-14</td>
<td>7.74E-07</td>
</tr>
<tr>
<td>4</td>
<td>7</td>
<td>30</td>
<td>4.94E-15</td>
<td>5.73E-10</td>
</tr>
<tr>
<td>25</td>
<td>3</td>
<td>18</td>
<td>8.67E-14</td>
<td>1.30E-05</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>20</td>
<td>9.69E-14</td>
<td>8.32E-10</td>
</tr>
<tr>
<td>25</td>
<td>7</td>
<td>22</td>
<td>1.23E-13</td>
<td>1.23E-13</td>
</tr>
</tbody>
</table>
Table 2: Performance of numerical transform inversion for the CMBRT04 model. The probability $p_+(t)$ was calculated at time points $t = 0, 1, \ldots, 31$ by transform inversion using different values of $h_{max}$ determined by the value of $c$ given in the Table. MaxErr is the maximum absolute discrepancy between the calculated values and the corresponding values calculated with $h_{max} = 50$. CPU times are for Matlab code.

<table>
<thead>
<tr>
<th>$c$</th>
<th>$h_{max}$</th>
<th>CPU time (s)</th>
<th>MaxErr</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>16</td>
<td>0.48</td>
<td>4.68E-06</td>
</tr>
<tr>
<td>1</td>
<td>17</td>
<td>0.53</td>
<td>5.72E-08</td>
</tr>
<tr>
<td>3</td>
<td>18</td>
<td>0.59</td>
<td>1.82E-11</td>
</tr>
<tr>
<td>5</td>
<td>20</td>
<td>0.77</td>
<td>3.75E-14</td>
</tr>
<tr>
<td>10</td>
<td>23</td>
<td>1.00</td>
<td>3.55E-14</td>
</tr>
</tbody>
</table>