Two-State Imprecise Markov Chains for Statistical Modelling of Two-State Non-Markovian Processes

Matthias C. M. Troffaes  
Department of Mathematical Sciences, Durham University, UK  
MATTHIAS.TROFFAES@DURHAM.AC.UK

Thomas Krak  
ELIS – FLip, Ghent University, Belgium  
THOMAS.KRAK@UGENT.BE

Henna Bains  
Department of Engineering, Durham University, UK  
HENNA.BAINS@DURHAM.AC.UK

Abstract

This paper proposes a method for fitting a two-state imprecise Markov chain to time series data from a two-state non-Markovian process. Such non-Markovian processes are common in practical applications. We focus on how to fit modelling parameters based on data from a process where time to transition is not exponentially distributed, thereby violating the Markov assumption. We do so by first fitting a many-state (i.e. having more than two states) Markov chain to the data, through its associated phase-type distribution. Then, we lump the process to a two-state imprecise Markov chain. In practical applications, a two-state imprecise Markov chain might be more convenient than a many-state Markov chain, as we have closed analytic expressions for typical quantities of interest (including the lower and upper expectation of any function of the state at any point in time). A numerical example demonstrates how the entire inference process (fitting and prediction) can be done using Markov chain Monte Carlo, for a given set of prior distributions on the parameters. In particular, we numerically identify the set of posterior densities and posterior lower and upper expectations on all model parameters and predictive quantities. We compare our inferences under a range of sample sizes and model assumptions.

Keywords: imprecise Markov chain, estimation, reliability, Markov assumption, MCMC

1. Introduction

Imprecise Markov chains provide a popular model for stochastic processes under severe uncertainty [13, 28, 7, 5, 21, 7, 6, 29, 27, 26, 18, 20, 24, 8, 10, 14, 15, 11, 16]. From a sensitivity analysis point of view, imprecise Markov chains model a set of stochastic processes, where only the bounds of this set are required to satisfy the Markov and stationarity conditions. Whilst one might think that computing with such a wide range of models is difficult, it turns out that computing expectation bounds is often extremely efficient, and nearly as easy as working with standard Markov chains [7]. This property makes imprecise Markov chains especially attractive for practical applications. Indeed, Markov and stationarity assumptions are frequently made for computational and conceptual ease, even in the face of clear violations of such assumptions [31].

Two-state imprecise Markov chains are particularly attractive as they allow closed analytic expressions for many quantities of interest. For instance, they admit a closed analytic expression for the expectation of any function of the state at any fixed point in time. Moreover, in practical applications, very often we are interested in stochastic processes that can be in one of two states, but where the Markov and stationarity conditions are violated to some extent.

In this paper, we demonstrate how imprecise two-state continuous time Markov chains can be fitted to time series data to model stochastic processes that violate the Markov assumption. We do so by combining two well-known techniques from the literature: the idea that arbitrary distributions of transition times can be modelled using phase-type distributions, which always have a representation as a Markov chain with potentially many states [17], and the idea that we can lump states together to reduce these complicated models back to a simpler two-state model [11]. In doing so, we also gain insight into what shape of processes imprecise Markov chains can model, potentially further opening up new applications for imprecise Markov chains.

We also manage to perform a full uncertainty quantification on the inferences from the fitted imprecise Markov chain, in the sense that the imprecise Markov chain is directly part of the likelihood of the model, and sets of distributions over the parameters of the imprecise Markov chain are fully propagated. This stands in contrast with how imprecise Markov chains have been fitted elsewhere in the literature, where uncertainty in the bounds on the transition rates is normally not propagated [25, 18, 24, 16].

We find that directly fitting a standard two-state Markov chain (precise or imprecise) does not predict system behaviour as correctly as the lumped phase-type model.

Moreover, under a standard model, the influence of the prior will vanish as we get more data, and thereby, so will any imprecision in the posterior. The lumped phase-type model on the other hand will always retain some level of imprecision, as this imprecision is a result of structural uncertainty, and not just due to a lack of data.

The paper is organised as follows. Section 2 gives a brief introduction to Markov chains. Section 3 details how to reduce a many-state Markov chain to a two-state imprecise Markov chain, using lumping. Section 4 presents a simple model that serves as an example of this reduction method. This simple model is also used throughout the remainder of the paper. Section 5 explains how to fit the modelling parameters based on time series data. Section 6 presents a fully worked numerical example. Section 7 concludes the paper.

2. Markov Chains

In this section, we present a minimal introduction to Markov chains.

Let \( S := \{1, \ldots, n \} \) denote a state space, which we assume to be finite. A stochastic process \((X_t)_{t \geq 0}\) is said to be a (homogeneous, continuous time) Markov chain on \( S \) if

\[
P(X_{t+\delta t} = j \mid X_t = i_1, \ldots, X_{t+k} = i_k, X_t = i) = P(X_{\delta t} = j \mid X_0 = i)
\]  

(1)

for every \( 0 < \delta t \in \mathbb{R} \), every \( 0 \leq t_1 < \cdots < t_k < t \in \mathbb{R} \), and every \( i_1, \ldots, i_k, i \in S \). It can be shown that a Markov chain is uniquely determined by its rate matrix \( Q \in \mathbb{R}^{n \times n} \):

\[
Q_{ij} := \lim_{\delta t \to 0} \frac{P(X_{\delta t} = j \mid X_0 = i) - P(X_0 = j \mid X_0 = i)}{\delta t}
\]

(2)

The laws of probability imply that the rows of \( Q \) need to sum to zero, the diagonal elements need to be non-positive, and all other entries need to be non-negative. The expectation of any function \( f \) of \( X_t \), given \( X_0 \), is given by:

\[
E(f(X_t) \mid X_0 = i) = [T_t f]_i
\]

(3)

where

\[
T_t := \exp(Q t) = \lim_{n \to \infty} \left( I + \frac{t}{n} Q \right)^n
\]

(4)

For a two-state Markov chain, the rate matrix \( Q \) is determined by just two parameters \( \lambda \geq 0 \) and \( \mu \geq 0 \):

\[
Q = \begin{bmatrix}
-\lambda & \lambda \\
\mu & -\mu
\end{bmatrix}
\]

(5)

and we have a closed expression for \( T_t \) (see Theorem 2 in the appendix):

\[
T_t = I + \frac{1 - e^{-(\lambda + \mu)t}}{\lambda + \mu} Q.
\]

(6)

3. Induced Imprecise Markov Chain By Lumping

Following [11, Sec. 3], we now explain how to induce imprecise Markov chains through lumping.

Imagine that we would like to reduce the state space to \( \{1, \ldots, m\} \) for some \( m < n \). For this purpose, we define a lumping map \( \Lambda : \{1, \ldots, n\} \to \{1, \ldots, m\} \), as well as the following non-linear operator \( \mathcal{Q} : \mathbb{R}^m \to \mathbb{R}^m \) [11, Sec. 3.3]:

\[
[\mathcal{Q} f]_k := \min_{i \in \Lambda^{-1}(k)} \{ Q(f \circ \Lambda) \}_i
\]

(7)

for all \( f \in \mathbb{R}^m \) and \( k \in \{1, \ldots, m\} \).

The lumped process \((Y_t)_{t \geq 0} \) defined by \( Y_t := \Lambda(X_t) \) then satisfies

\[
[T_t f]_i \leq E(f(Y_t) \mid Y_0 = i) \leq [T_t f]_i
\]

(8)

where

\[
T_t := \lim_{r \to \infty} \left( I + \frac{t}{r Q} \right)^r
\]

(9)

and \( T_t(f) := -T_t(-f) \) for every \( f \in \mathbb{R}^m \). Whilst the process \( Y_t \) is under normal circumstances not Markovian, the expectation bounds in Eq. (8) do correspond to the lower and upper expectation induced by the imprecise Markov chain with lower transition rate operator \( \mathcal{Q} \).

For example, consider a situation where we leave state 1 as it is, and lump all states \( \{2, \ldots, n\} \) into a single state. This corresponds to the lumping map \( \Lambda(1) := 1 \) and \( \Lambda(2) = \cdots = \Lambda(n) := 2 \). As this is the lumping we will use throughout the remainder of the paper, for clarity of exposition, we will denote state 1 of the lumped process by \( \alpha \) and state 2 of the lumped process by \( \beta \). Equation (7) becomes:

\[
[\mathcal{Q} f]_\alpha := Q_{11} f_\alpha + \sum_{j=2}^n Q_{1j} f_\beta = -Q_{11} (f_\beta - f_\alpha)
\]

(10)

\[
[\mathcal{Q} f]_\beta := \min_{i=2}^n \left\{ Q_{i1} f_\alpha + \sum_{j=2}^n Q_{ij} f_\beta \right\} = \min_{i=2}^n Q_{i1} (f_\alpha - f_\beta)
\]

(11)

Further, in this case, because the reduced state space has only two states, we know that we have a closed analytic expression for the lower transition operator \( T_\alpha \).

More precisely, any two-state imprecise Markov chain can be described by just 4 parameters, \( \lambda_\alpha \) and \( \lambda_\beta \) (with \( 0 < \lambda_\alpha \leq \lambda_\beta \)) which bound the transition rate from state \( \alpha \) to state \( \beta \), and \( \mu_\alpha \) and \( \mu_\beta \) (with \( 0 < \mu_\alpha \leq \mu_\beta \)), which bound the transition rate from state \( \beta \) to state \( \alpha \):

\[
[\mathcal{Q} f]_\alpha = \min\{\lambda_\alpha (f_\beta - f_\alpha), \lambda_\beta (f_\beta - f_\alpha)\},
\]

(12)

\[
[\mathcal{Q} f]_\beta = \min\{\mu_\alpha (f_\alpha - f_\beta), \mu_\beta (f_\alpha - f_\beta)\}.
\]

(13)
Then, with the $\lambda$ and $\mu$ that achieve the minimum in Eqs. (12) and (13):

$$\left(\lambda_f, \mu_f\right) := \begin{cases} 
(\lambda_\alpha, \mu_\alpha) & \text{if } f_\alpha \leq f_\beta \\
(\lambda_\beta, \mu_\beta) & \text{if } f_\alpha > f_\beta
\end{cases} \quad (14)$$

and

$$Q_f := \begin{bmatrix} -\lambda_f & \lambda_f \\
\mu_f & -\mu_f \end{bmatrix} \quad (15)$$

it can be shown that (see [10] or Theorem 5 in the appendix for a shorter proof)

$$\mathcal{T}_f = f + \frac{1-e^{-(\lambda_f+\mu_f)t}}{\lambda_f+\mu_f} Q_f f. \quad (16)$$

Comparing Eqs. (10) and (11) with Eqs. (12) and (13), we see that we can interpret our lumped process as one with a precise transition rate

$$\lambda_\alpha = \bar{\lambda} = -Q_{11} \quad (17)$$

for going from state $\alpha$ to state $\beta$, and lower and upper transition rates

$$\mu_\alpha = \min_{i=2}^n Q_{i1} \quad \mu^* = \max_{i=2}^n Q_{i1} \quad (18)$$

for going from state $\beta$ to state $\alpha$.

In conclusion, provided that the bounds in Eq. (8) are sufficiently tight for our needs, we can use an induced two-state imprecise Markov chain to analytically bound arbitrarily complex many-state Markov chains. In particular, this allows us to analytically bound any two-state process where transitions between states follow far more complex distributions, and specifically, phase-type distributions, which are distributions that arise from transition times between any two states in an arbitrary Markov chain.

4. A Simple Model

In this section, we demonstrate, on a very simple model, how we can use the ideas explained so far. We will study how a three-state Markov chain can be used to model a two-state non-Markovian process, and how the lumping of this three-state Markov chain can bound this two-state process. The analysis presented here carries over easily to models that incorporate many more states to capture non-Markovian behaviour.

Before carrying on with the model, we emphasize that with this approach we often do not have a semantics for the additional states in the many-state model. Their existence is only motivated by obtaining a better fit to the observed transition times. For example, we could compute a stationary distribution that also includes those auxiliary states, however this would not necessarily have any meaning to an end user. For ease of presentation and interpretation, in the model that is presented next, the additional state is physically motivated (and is thereby provided with a semantics), although such interpretation is not strictly necessary.

Consider the three-state Markov chain depicted in Fig. 1. Here, we could imagine state 1 to correspond to a working state, and states 2 and 3 to correspond to non-working states. The transition from 1 to 2 represents failure. Time to failure in this model is exponential, as is commonly the case in applications. For repair however, we consider two distinct situations. Either, the system can be quickly repaired remotely with mean time $1/\mu_1$, or, the system needs to be serviced locally to fix the problem, at a much slower repair time $1/\mu_2 + 1/\mu_3 \gg 1/\mu_1$. The ratio $\mu_1/\mu_2+\mu_3$ determines how often quick remote repair is successful.

Although here, we clearly have reliability in mind, these many-state Markov models are commonly used for modelling non-exponentially distributed transition times, for instance in human resource planning [31] and medical sciences [2, 3].

Our process has the following transition rate matrix:

$$Q = \begin{bmatrix} -\lambda & \lambda & 0 \\
\mu_1 & -\mu_1 + \mu_2 & \mu_2 \\
\mu_3 & 0 & -\mu_3 \end{bmatrix} \quad (19)$$

Time to transition from state 2 to state 1 is no longer exponential, but follows a so-called phase-type distribution, because the process potentially has to pass through state 3 before reaching state 1. The specific phase-type distribution for our model here has a cumulative distribution function equal to [17, Lemma 2.2.2]

$$\Phi_{21}(t) = 1 - \begin{bmatrix} 1 & 0 \end{bmatrix} \exp \left( \begin{bmatrix} -\mu_1 & -\mu_2 & \mu_2 \\
0 & -\mu_3 \\
\mu_3 & 0 & -\mu_3 \end{bmatrix} t \right) \begin{bmatrix} 1 \\
1 \end{bmatrix} \quad (20)$$

and so, by Theorem 1 (see the appendix),

$$= 1 - \left( e^{-(\mu_1+\mu_2)t} + \mu_2 e^{-(\mu_1+\mu_2)t} - e^{-(\mu_1+\mu_2)t} \right). \quad (21)$$

This distribution is depicted in Fig. 2. The corresponding probability density function is given by

$$\phi_{21}(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \exp \left( \begin{bmatrix} -\mu_1 & -\mu_2 & \mu_2 \\
0 & -\mu_3 \\
\mu_3 & 0 & -\mu_3 \end{bmatrix} t \right) \begin{bmatrix} \mu_1 \\
\mu_2 \\
\mu_3 \end{bmatrix} \quad (22)$$

Figure 1: Example of a Markov chain.
5. Fitting Parameters

In reality, we do not know the rates $\lambda$, $\mu_1$, $\mu_2$, and $\mu_3$. However, we may have some time to failure and time to repair data, and these data can be used to estimate these parameters.

For $\lambda$, if we have an i.i.d. sequence $x_1, \ldots, x_N$ of observed times to failure, the maximum likelihood estimate is:

$$\hat{\lambda} = \frac{N}{\sum_{i=1}^{N} x_i}$$  \hspace{1cm} (27)

If we have a Gamma$(s, \tau)$ prior on $\lambda$, where the hyperparameter $s$ determines the strength of the prior, and $1/\tau$ is our prior expectation of $\lambda$ before having seen the data, then we can use the posterior expectation of $\lambda$ as an estimate:

$$\hat{\lambda} = \frac{s + N}{s/\tau + \sum_{i=1}^{N} x_i}$$  \hspace{1cm} (28)

For full uncertainty quantification, we can propagate the entire posterior distribution:

$$\lambda \mid x_1 \ldots x_N \sim \text{Gamma} \left( s + N, s/\tau + \sum_{i=1}^{N} x_i \right)$$  \hspace{1cm} (29)

In either case, we can use intervals for the hyperparameters (for example, as in [24]), in case we prefer to represent our prior knowledge more robustly using a set of prior distributions.

For $\mu_1$, $\mu_2$, and $\mu_3$, given an i.i.d. sequence $y_1, \ldots, y_M$ of observed times to repair, a variety of advanced methods have been developed in the literature in order to fit phase-type distributions [1, 22, 12, 23, 30, 19]. However, in general, there is no closed expression for the maximum likelihood estimate, or even for the posterior distribution. In our case, we know that the likelihood does have a closed form:

$$f(y_1 \ldots y_M \mid \mu_1, \mu_2, \mu_3) = \prod_{i=1}^{M} \phi_{21}(y_i)$$  \hspace{1cm} (30)

where we derived $\phi_{21}$ in Eq. (23). Using this expression, we can easily obtain (correlated) samples from the posterior joint distribution of $(\mu_1, \mu_2, \mu_3)$ using Markov chain Monte Carlo, as long as the prior also has a simple analytic form (e.g., we could also use Gamma distributions for these). Here too, if we prefer to represent our prior knowledge more robustly, we could use a set of Gamma distributions, and then simply take lower and upper envelopes over different Markov chain Monte Carlo runs to obtain lower and upper expectations. This allows us to do a full uncertainty quantification.

6. Numerical Example

For reasons of confidentiality, here, we use simulated data based on an approximate fit to data from an actual reliability system. This ensures that our example is still somewhat representative of a real reliability system. At the same time, the use of simulated data allows us to explore a wider range of sample sizes, and allows us to investigate how our fits evolve as more data is available.

Specifically, we studied data generated from the Markov chain depicted in Fig. 1, with $\lambda = 0.01$, $\mu_1 = 4$, $\mu_2 = 0.6$, and $\mu_3 = 0.2$. The generated data for $y_i$ is depicted in Fig. 3.
Due to the structure of Eqs. (31) and (32), intuitively, we expect that the relevant extreme cases are the ones obtained for

$$\{\tau_0, \tau_1, \tau_2, \tau_3\} \in \{(50, 1, 4, 40), (250, 0.1, 0.2, 2)\}$$

so, for ease of presentation, we simply plot the results under these two priors, as these cover the two relevant extreme cases within our analysis.

Marginal posterior densities are depicted in Fig. 4, where we had a relatively small sample size ($N = M = 30$). We note in particular that $\mu_2$ appears to be the hardest parameter to estimate, compared to the other parameters, both in terms of relative posterior variance and in terms of relative posterior imprecision. This is interesting because, through the lumping, our inferences from the two-state imprecise Markov chain do not depend on $\mu_2$. The posterior distributions are generally skewed. The limiting posterior lower expectation of being in state $\alpha$ is around 0.87 (solid line on bottom left), whereas the posterior upper expectation is around 0.9975 (dashed line on bottom right).

Marginal posterior density plots for a larger sample size ($N = M = 300$) are depicted in Fig. 6. Here, the posterior distributions are less skewed, and the influence of the prior has been reduced. The limiting posterior lower expectation of being in state $\alpha$ is around 0.94, whereas the posterior upper expectation is around 0.9976.

For comparison, we also fitted a model where, instead of a phase-type distribution for repair, we simply assumed exponential repair, with $\mu \sim \text{Gamma}(s, s \tau)$ where $\tau \in \{0.1, 40\}$ (this roughly corresponds to the range for the parameters of the phase-type model). The marginal posterior distributions for $\mu$ and $\pi := \frac{\mu}{\lambda + \mu}$ are depicted in Figs. 5 and 7. We can see that the fit is contained within the fit of the lumping model, as we would expect.

However, the exponential model will obviously not predict system behaviour as accurately as the lumped phase-type model. Further, asymptotically, as we get more data, eventually, the influence of the prior will vanish, and thereby, under the exponential model, the imprecision in the posterior will vanish. The lumped phase-type model on the other hand will never rid of all imprecision, as there is imprecision as a result of structural uncertainty (which will never vanish), and not just due to a lack of data.

At this point, the reader may wonder why we would not use the three-state model directly to perform our inferences, since we estimate $\mu_2$ anyway. Whilst this could be workable in the simple three-state model, in a much more complicated setting, working with a simpler (fewer state) model can be advantageous, as mentioned earlier. First, the interpretation of the two-state model may be more convenient, as the auxiliary states may not have a direct semantics. Secondly, a two-state model has a computational advantage. For example, in reliability, many thousands of components may need to be considered at once, and in this case, having a simple two-state model can be particularly attractive.
Figure 4: Posterior densities for the lumped model under both priors (solid and dashed line), for small sample size ($N = M = 30$). The vertical lines indicate posterior expectations.

Figure 5: Posterior densities for the exponential model under both priors (solid and dashed line), for small sample size ($N = M = 30$). The vertical lines indicate posterior expectations.

Figure 6: Posterior densities for the lumped model under both priors (solid and dashed line), for large sample size ($N = M = 300$). The vertical lines indicate posterior expectations.

Figure 7: Posterior densities for the exponential model under both priors (solid and dashed line), for large sample size ($N = M = 300$). The vertical lines indicate posterior expectations.

7. Conclusion

In this paper, we explored how to combine fitting of phase-type distributions and lumping of Markov chains into imprecise Markov chains, to fit two-state imprecise Markov chains to data from a non-Markovian two-state processes.

Through Markov chain Monte Carlo simulation, we demonstrated how to perform a full uncertainty quantification, deriving sets of posterior distributions on relevant quantities of interest, from a set of prior distributions on all parameters. In doing so, we have shown how non-Markovian pro-
cesses can be statistically fitted, in one fell swoop, through robust Bayesian analysis. This analysis stands in contrast with how imprecise Markov chains have been fitted elsewhere in the literature, which typically comprised of identifying lower and upper bounds on the transition rates of a precise Markov chain model, and then using those lower and upper bounds as bounds on the rates of an imprecise Markov chain model (see for instance [25, 18, 24, 16]). Those approaches cannot model non-Markovian behaviour as correctly as the lumped phase-type model. Also, in those approaches, the influence of the prior will vanish as we get more data, and thereby, so will any imprecision in the posterior. The lumped phase-type model on the other hand will always retain some level of imprecision, as this imprecision is a result of structural uncertainty, and not just due to a lack of data.

In our numerical example, the many-state model that we used was still quite simple. In general, to model more complex phase-type distributions for transition times, we may not have a fast closed form expression for the phase-type density. Either, one would have to resort to direct evaluation of a matrix exponential as part of the Markov chain Monte Carlo run, or, one could simply sample directly from the likelihood (which can be done quickly for any phase-type distribution), and then use approximate Bayesian computation [9] to fit the model.

Acknowledgments

We thank all reviewers for their constructive comments on the paper. The first two authors are supported by the H2020 Marie Curie ITN, UTOPIAE, Grant Agreement No. 722734. Henna Bains is funded by an Offshore Renewable Energy (ORE) Catapult Doctoral STUDENTSHIP.

Appendix A. Technical Results

Throughout, we use the fact that for any matrix $A$:

$$\exp(A) = \sum_{n=0}^{\infty} \frac{1}{n!} A^n$$

(38)

**Theorem 1** For every $a, b, c \in \mathbb{R}$ such that $a \neq c$, we have that

$$\exp\left( \begin{bmatrix} a & b \\ 0 & c \end{bmatrix} \right) = \begin{bmatrix} e^a & \frac{b e^{c-a}}{c-a} \\ 0 & e^c \end{bmatrix}$$

(39)

**Proof** We first show by induction that, for every $n \in \mathbb{N}$,

$$\begin{bmatrix} a & b \\ 0 & c \end{bmatrix}^n = \begin{bmatrix} a^n & \frac{b a^n - c^n}{c-a} \\ 0 & c^n \end{bmatrix}$$

(40)

Indeed, clearly, Eq. (40) holds for $n = 0$. Assume Eq. (40) holds for a particular fixed value of $n$. Then,

$$\begin{bmatrix} a & b \\ 0 & c \end{bmatrix}^{n+1} = \begin{bmatrix} a & b \\ 0 & c \end{bmatrix}^n \begin{bmatrix} a & b \\ 0 & c \end{bmatrix} = \begin{bmatrix} a^n & \frac{b a^n - c^n}{c-a} \\ 0 & c^n \end{bmatrix} \begin{bmatrix} a & b \\ 0 & c \end{bmatrix} = \begin{bmatrix} a^{n+1} & \frac{b a^{n+1} - c^{n+1}}{c-a} \\ 0 & c^{n+1} \end{bmatrix}$$

(41)

Now note that

$$ba^n + cb^{a^n-c^n} = b a^n (a-c) + c(a^n-c^n) = b a^n - c^{n+1}$$

(42)

$$b a^n + cb^{a^n-c^n} = b a^n (a-c) + c(a^n-c^n) = b a^n - c^{n+1}$$

(43)

We can now prove Eq. (49). Indeed, by Eq. (38),

$$\exp\left( \begin{bmatrix} a & b \\ 0 & c \end{bmatrix} \right) = \left[ \begin{array}{cc} 1 & 0 \\ 0 & 1 \end{array} \right] + c \left[ \begin{array}{cc} -a & a \\ b & -b \end{array} \right]$$

(49)

where $c := \frac{1-e^{-a-b}}{a+b}$. Indeed, clearly Eq. (50) holds for $n = 1$. Assume Eq. (50) holds for a particular fixed value of $n \geq 1$. Then,

$$\begin{bmatrix} a & a \\ b & -b \end{bmatrix}^{n+1} = \begin{bmatrix} a & a \\ b & -b \end{bmatrix}^n \begin{bmatrix} a & a \\ b & -b \end{bmatrix} = \begin{bmatrix} a^n & \frac{b a^n - c^n}{c-a} \\ 0 & c^n \end{bmatrix} \begin{bmatrix} a & a \\ b & -b \end{bmatrix} = \begin{bmatrix} a^{n+1} & \frac{b a^{n+1} - c^{n+1}}{c-a} \\ 0 & c^{n+1} \end{bmatrix}$$

(51)

$$\begin{bmatrix} a & a \\ b & -b \end{bmatrix}^{n+1} = \begin{bmatrix} a & a \\ b & -b \end{bmatrix}^n \begin{bmatrix} a & a \\ b & -b \end{bmatrix} = \begin{bmatrix} a^n & \frac{b a^n - c^n}{c-a} \\ 0 & c^n \end{bmatrix} \begin{bmatrix} a & a \\ b & -b \end{bmatrix} = \begin{bmatrix} a^{n+1} & \frac{b a^{n+1} - c^{n+1}}{c-a} \\ 0 & c^{n+1} \end{bmatrix}$$

(52)

$$\begin{bmatrix} a & a \\ b & -b \end{bmatrix}^{n+1} = \begin{bmatrix} a & a \\ b & -b \end{bmatrix}^n \begin{bmatrix} a & a \\ b & -b \end{bmatrix} = \begin{bmatrix} a^n & \frac{b a^n - c^n}{c-a} \\ 0 & c^n \end{bmatrix} \begin{bmatrix} a & a \\ b & -b \end{bmatrix} = \begin{bmatrix} a^{n+1} & \frac{b a^{n+1} - c^{n+1}}{c-a} \\ 0 & c^{n+1} \end{bmatrix}$$

(53)

$$\begin{bmatrix} a & a \\ b & -b \end{bmatrix}^{n+1} = \begin{bmatrix} a & a \\ b & -b \end{bmatrix}^n \begin{bmatrix} a & a \\ b & -b \end{bmatrix} = \begin{bmatrix} a^n & \frac{b a^n - c^n}{c-a} \\ 0 & c^n \end{bmatrix} \begin{bmatrix} a & a \\ b & -b \end{bmatrix} = \begin{bmatrix} a^{n+1} & \frac{b a^{n+1} - c^{n+1}}{c-a} \\ 0 & c^{n+1} \end{bmatrix}$$

(54)

$$\begin{bmatrix} a & a \\ b & -b \end{bmatrix}^{n+1} = \begin{bmatrix} a & a \\ b & -b \end{bmatrix}^n \begin{bmatrix} a & a \\ b & -b \end{bmatrix} = \begin{bmatrix} a^n & \frac{b a^n - c^n}{c-a} \\ 0 & c^n \end{bmatrix} \begin{bmatrix} a & a \\ b & -b \end{bmatrix} = \begin{bmatrix} a^{n+1} & \frac{b a^{n+1} - c^{n+1}}{c-a} \\ 0 & c^{n+1} \end{bmatrix}$$

(55)

So, by induction, Eq. (50) must hold for all $n \in \mathbb{N}$. We can now prove Eq. (49). Indeed, by Eq. (38),

$$\exp\left( \begin{bmatrix} a & a \\ b & -b \end{bmatrix} \right) = \left[ \begin{array}{cc} 1 & 0 \\ 0 & 1 \end{array} \right] + \frac{1}{n!} \begin{bmatrix} -a & a \\ b & -b \end{bmatrix}^n$$

(56)

So, by induction, Eq. (50) must hold for all $n \in \mathbb{N}$. We can now prove Eq. (49). Indeed, by Eq. (38),

$$\exp\left( \begin{bmatrix} a & a \\ b & -b \end{bmatrix} \right) = \left[ \begin{array}{cc} 1 & 0 \\ 0 & 1 \end{array} \right] + \frac{1}{n!} \begin{bmatrix} -a & a \\ b & -b \end{bmatrix}^n$$

(57)
= \sum_{n} c_n \begin{bmatrix} -a & a \\ b & -b \end{bmatrix} n^a \\
= -\left( a - b \right) \begin{bmatrix} -a & a \\ b & -b \end{bmatrix} \begin{bmatrix} a + b \\
= -(a - b) \begin{bmatrix} -a & a \\ b & -b \end{bmatrix} \begin{bmatrix} a + b \\

Next, we want to prove that our expression for $T_f$ in Eq. (16) indeed corresponds to the lower transition operator. To do so, however, we will not use the definition of Eq. (9). Instead, we will take Eq. (16), that is (

as a definition of $T_f$, and then prove that this operator solves a specific differential equation. It follows then from [20, Sec. 2.6] that the solution to this differential equation is the lower transition operator.

**Lemma 3** For every $f \in \mathbb{R}^2$, we have that

$$
[T_f]_\alpha - [T_f]_\beta = (f_\alpha - f_\beta) e^{-\left(\lambda_f + \mu_f\right) t}.
$$

**Proof**

$$
[T_f]_\alpha - [T_f]_\beta = (f_\alpha + \frac{1}{\lambda_f} + \mu_f) - (f_\beta + \frac{1}{\lambda_f} + \mu_f) = (f_\alpha - f_\beta) (f_\alpha - f_\beta) e^{-\left(\lambda_f + \mu_f\right) t}.
$$

**Lemma 4** $\lambda_{T_f} = \lambda_f$ and $\mu_{T_f} = \mu_f$.

**Proof** $\lambda_f$ and $\mu_f$ are determined solely by the sign of $f_\alpha - f_\beta$. Since, by Lemma 3, the sign of $[T_f]_\alpha - [T_f]_\beta$ is the same as the sign of $f_\alpha - f_\beta$, we have that $(\lambda_f, \mu_f) = (\lambda_{T_f}, \mu_{T_f})$.

**Theorem 5** The operator $T_f$, as defined in Eq. (60), solves

$$
\frac{d}{dt} [T_f] = Q [T_f]
$$

with initial condition $T_0[f] = f$.

**Proof** We see that the initial condition $T_0[f] = f$ is satisfied. Evaluating the left-hand side of Eq. (67) we have:

$$
\frac{d}{dt} [T_f] = e^{-\left(\lambda_f + \mu_f\right) t} Q [T_f]
$$

Evaluating the right-hand side of Eq. (67) and using Lemma 4 we have:

$$
Q [T_f] = Q_{T_f} [T_f]
$$

By Eq. (50), $Q_{T_f} = -\left(\lambda_f + \mu_f\right) Q_f$. Therefore,

$$
Q [T_f] = \left( Q_f - \frac{1}{\lambda_f + \mu_f} \right) (\lambda_f + \mu_f) Q_f f
$$

$$
= e^{-\left(\lambda_f + \mu_f\right) t} Q_f f
$$

**Appendix B. Stan Model Specification**

```stan
functions {
  real phasetype_lpdf(real a, real b, real c, real d) {
    real expa = exp(a * t);
    real expb = exp(b * t);
    real expc = exp(c * t);
    real expd = exp(d * t);
    return expa + expb + expc + expd;
  }
  real min2(real a, real b) {
    return min(a, b);
  }
}

real phasetype_lpdf(real a, real b) {
  real tmp[2];
  tmp[1] = a;
  tmp[2] = b;
  return min2(tmp[1], tmp[2]);
}
}
data {
  // data
  int NX;
  int NY;
  vector[NX] Y;
  int YM;
  vector[NY] Y;
  // hyperparameters
  real S;
  real T0;
  real T1;
  real T2;
  real T3;
}
parameters {
  real<lower=0> lambda;
  real<lower=0> mu1;
  real<lower=0> mu2;
  real<lower=0> mu3;
  real<lower=0> lambda;

  lambda = gamma(S, S + T0);
  mu1 = gamma(S, S + T1);
  mu2 = gamma(S, S + T2);
  mu3 = gamma(S, S + T3);
  for (n in 1:NY)
    Y[n] = phasetype(lambda, mu1, mu2, mu3);
}
model {
  lambda - gamma(S, S + T0);
  mu1 - gamma(S, S + T1);
  mu2 - gamma(S, S + T2);
  mu3 - gamma(S, S + T3);
  for (n in 1:NY)
    X[n] = exponential(lambda);
    Y[n] - phasetype(mu1, mu2, mu3);
}
generated quantities {
  real lmu = min2(mu1, mu2);
  real upi = upi / (lambda + lmu);
  real upi = upi / (lambda + lmu);
}
```
References


