A heuristic procedure for compact Markov representation of PH distributions

Illés Horváth
MTA-BME Information Systems Research Group
Inter-University Center of Telecommunications
and Informatics, Debrecen
horvath.illes.antal@gmail.com

Miklós Telek
Department of Networked Systems and Services
Budapest University of Technology and
Economics
telek@hit.bme.hu

ABSTRACT

The minimal Markovian representation of PH distributions is an open research problem, which was actively investigated during the last two decades. We present a numerical method for finding small Markovian representation of PH distributions and investigate the general quality of the method by comparing the size of the obtained representation with the size of the representation obtained by alternative methods. Our numerical method intends to find a small Markovian representation. We report examples when the obtained representation is larger than the minimal Markovian representation.

Keywords: PH distribution, minimal representation, feedback Erlang block, unicyclic block.

1. INTRODUCTION

Phase-type (PH) distributions play an important role in Markovian modeling by providing a large class of nonnegative distributions (both discrete and continuous) that allow for a Markov chain-based stochastic interpretation. A PH-distribution is defined as the absorption time of a (discrete or continuous time) Markov chain [12, 13], and is thus characterized by an initial vector and a Markov generator matrix. In the present paper, we focus on the continuous time phase-type distributions, but note that many of the ideas and methods presented may well be applicable to their discrete-time counterparts.

However, the Markov chain representation of a specific PH distribution is not unique; there are many equivalent Markov chain representations for any given PH distribution (even with different sizes). This non-uniqueness raises a number of rather involved research issues. The easiest solution for these technical issues is the use of *canonical representation* [6, 9] – a conveniently defined, minimal, unique Markov chain representation. Unfortunately, canonical representations are available only for order 2 and order 3 general PH distributions [9] and some special subclasses of higher order

PH distributions, e.g. acyclic PH distributions [6]. In spite of the continuous research efforts during the last 2 decades our knowledge on the convenient Markov chain representation of higher order general PH distributions is rather limited. Based on O'Cinneide's characterization theorem [14] Mocanu and Commault proposed a sparse representation, which is unique, sparse (contains a limited number of transitions), but not minimal [10]. Their proposed representation is composed by small Markov chain blocks which represent either one real eigenvalue or a complex conjugate pair of eigenvalues of the PH distribution (which are associated with the poles of the rational Laplace transform of the distribution function). The non-minimal size of the representation is due to the fact that each complex conjugate eigenvalue pair is represented with a Markov chain block of size larger than 2. In this paper we investigate if other compact representations can be obtained by representing more than a single complex conjugate eigenvalue pair by a single small Markov chain block.

1.1 Related works

A very interesting feature of the research effort on the subject of this paper is that there are two partially independently evolving research fields, the theory of Phase type distributions and the theory of linear systems, which independently investigate the same problems with completely different terminology time to time.

On the *field of phase type distribution* continuous (time) models are more commonly considered. After some results on canonical forms O'Cinneide defined a number of open research problems associated with the minimal Markov chain representation of general PH distributions [15]. In this paper he published a conjecture that every PH distribution has a minimal unicyclic representation. While it was shown by a counterexample [8] that the conjecture is false, the unicyclic structure has nice properties for representing PH distributions in an efficient way. Indeed, the counterexample of He and Zhang [8] was such that the combination of 2 unicyclic blocks allowed the minimal representation instead of a single one.

In this work we practically follow the procedure of Mocanu and Commault [10], but we replace their building block (feedback Erlang block) with the more flexible unicyclic block.

On the *field of linear systems* a different terminology (system

response to the impulse function, transfer function) describe similar mathematical concepts in case of single input single output (SISO) systems. On this field the discrete transfer functions and their z-transform are more widespread. A non-negative transfer function of a SISO system is a counterpart of a distribution and the positive representation of the linear system is the counterpart of the Markovian representation [3].

There are several results about the minimal positive representation of specific subset of linear systems [2, 17, 11, 7]. The subsets of linear systems considered in these works are not natural in the analysis of PH distributions. For example the counterpart of the set of order 3 linear systems with distinct real eigenvalues [17] is larger than the set of acyclic PH distribution of order 3 but smaller than the set of PH distribution of order 3. The different motivations behind the evolution of these two fields, the use of different methodologies and the different time axes make the translation of the related results rather difficult. Some of the results have strong geometric interpretations for the location of the eigenvalues, which needs to be translated from the unit circle (in case of discrete time system) to the half plane (in case of continuous time system).

We conclude that the minimal positive representation of a transfer function and the minimal Markov representation of a PH distribution are rather similar problems which were investigated in different communities with very little mutual influence during the last 2 decades, and the roots of the concepts used in this work are present in both fields.

The rest of the paper is organized as follows. In Section 2, we present the theoretical setup for the work: Subsection 2.1 includes the basic definitions, Subsection 2.2 gives a short overview of various representations and transformations in general, while Subsection 2.3 presents a more detailed review of the Mocanu–Commault representation. Subsection 2.4 presents the proposed unicyclic representations and the necessary theory behind them. Section 3 contains the main contribution of the paper: the experimental analysis of Markovian representations with unicyclic blocks in various settings. Section 4 concludes the paper.

2. PRELIMINARIES

2.1 Basic definitions

DEFINITION 1. *A pair* $(\boldsymbol{\alpha}, \mathbf{A})$, *where* $\boldsymbol{\alpha}$ *is a vector of size* $1 \times n$ *and* $\mathbf{A}$ *is a matrix of size* $n \times n$ *is said to be a* vector-matrix pair *of size* n.

DEFINITION 2. *The vector-matrix pair of size* n, $(\boldsymbol{\alpha}, \mathbf{A})$, *is said to be* Markovian *if* $\boldsymbol{\alpha}$ *and* $\mathbf{A}$ *have the following properties:* $\alpha_i \geq 0$, $A_{ii} < 0$, $A_{ij} \geq 0$ *for* $i \neq j$, $\mathbf{A1} \leq 0$, *and* $\mathbf{A}$ *is non-singular.* $\mathbf{1}$ *denotes the column vector of ones of the appropriate size.*

PH distributions can be defined as follows.

DEFINITION 3. *Let* X *be a random variable with cumulative distribution function (CDF)* $F_X(x) = Pr(X \leq x)$. X is PH distributed if there is a finite size Markovian vector-matrix pair, $(\boldsymbol{\alpha}, \mathbf{A})$, for which

$$F_X(x) = Pr(X \leq x) = 1 - \boldsymbol{\alpha} e^{\mathbf{A}x} \mathbf{1}, \qquad (1)$$

In this case we say that X *is phase-type distributed with representation* $(\boldsymbol{\alpha}, \mathbf{A})$, *PH($\boldsymbol{\alpha}, \mathbf{A}$) distributed, for short.*

In Definition 3, vector $\boldsymbol{\alpha}$ is referred to as initial row vector and matrix $\mathbf{A}$ as transient generator matrix. In this paper we assume that $\boldsymbol{\alpha}\mathbf{1} = 1$, which means that there is no probability mass at $t = 0$ and $F_X(0) = 0$. The probability density function (PDF) and the Laplace transform X are

$$f_X(x) = \boldsymbol{\alpha} e^{\mathbf{A}x} (-\mathbf{A}) \mathbf{1}, \qquad (2)$$

$$f_X^*(s) = E(e^{-sX}) = \boldsymbol{\alpha}(s\mathbf{I} - \mathbf{A})^{-1}(-\mathbf{A})\mathbf{1}, \qquad (3)$$

(2) defines the matrix-exponential function associated with the vector-matrix pair $(\boldsymbol{\alpha}, \mathbf{A})$, both for Markovian and non-Markovian vector-matrix pairs, whose Laplace transform is a rational function of s (3). The poles of $f_X^*(s)$ coincide with the eigenvalues of $\mathbf{A}$ with identical multiplicity.

2.2 Different Representations of PH Distributions

The vector-matrix representation of $f_X(x)$ is not unique. There are different vector-matrix pairs, both with identical size and different sizes resulting in the same matrix-exponential function. Also, there might be representations where the vector-matrix pair does not satisfy the nonnegativity criterions $\alpha_i \geq 0, A_{ij} \geq 0$ in Definition 2. Such representations will be referred to as non-Markovian.

E.g., if $(\boldsymbol{\alpha}, \mathbf{A})$ of size n describes a PH distribution with density function $f_X(x)$ and a non-singular matrix $\mathbf{B}$ of size $n \times n$ is such that $\mathbf{B1} = \mathbf{1}$ then $(\boldsymbol{\gamma}, \mathbf{G})$ with $\boldsymbol{\gamma} = \boldsymbol{\alpha}\mathbf{B}$, $\mathbf{G} = \mathbf{B}^{-1}\mathbf{A}\mathbf{B}$ is a different representation of the same distribution with the same size. Further more, if $(\boldsymbol{\alpha}, \mathbf{A})$ represents a PH distribution of size n, $(\boldsymbol{\gamma}, \mathbf{G})$ represents a PH distribution of size m and there exists a matrix $\mathbf{W}$ of cardinality $n \times m$, such that $\boldsymbol{\alpha}\mathbf{W} = \boldsymbol{\gamma}$, $\mathbf{A}\mathbf{W} = \mathbf{W}\mathbf{G}$, $\mathbf{W}\mathbf{1}_m = \mathbf{1}_n$ then $(\boldsymbol{\alpha}, \mathbf{A})$ and $(\boldsymbol{\gamma}, \mathbf{G})$ represent the same distribution [5].

DEFINITION 4. X *is PH distributed with density function* $f_X(x)$. *The* $(\boldsymbol{\alpha}, \mathbf{A})$ *vector-matrix pair of size* n *is said to be a* minimal representation *of* X *if there is no vector-matrix pair of smaller size whose associated matrix-exponential function is* $f_X(x)$.

Methods finding a minimal (not necessarily Markovian) vector-matrix pair are available either directly from $f_X(x)$ (e.g. through Proposition 2.3 in [1]) or from any vector-matrix pair describing the same distribution [5]. Hereafter we assume that a minimal vector-matrix pair is available.

DEFINITION 5. X *is PH distributed with density function* $f_X(x)$. *The* $(\boldsymbol{\alpha}, \mathbf{A})$ *Markovian vector-matrix pair of size* n *is said to be a* minimal Markovian representation *of* X *if there is no Markovian vector-matrix pair of smaller size whose associated matrix-exponential function is* $f_X(x)$.
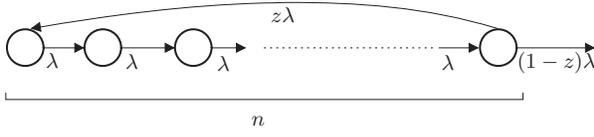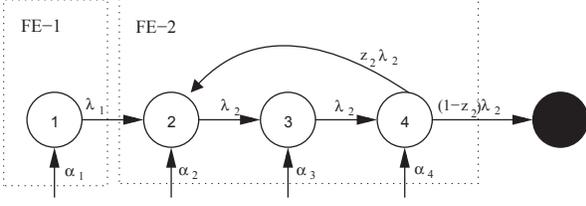
Figure 1: FE-diagonal block.

Figure 2: FE-diagonal representation of a generator with a real eigenvalue (λ_1) and a pair of complex ones.

As opposed to a minimal vector-matrix pair, a minimal Markovian vector-matrix pair is not readily available. The ultimate goal of our research is to find a minimal Markovian $(\boldsymbol{\alpha}, \mathbf{A})$ vector-matrix pair for a PH distribution with density function $f_X(x)$. Towards this ultimate goal, in the current work we try to improve the procedure by Mocanu and Commault which computes a Markovian representation for $f_X(x)$.

2.3 The Mocanu and Commault representation

The procedure for computing the Mocanu and Commault [10] representation is composed of the following steps:

1. compute the eigenvalues of the PH distribution,

2. represent the eigenvalues with a Markovian matrix using FE-diagonal blocks,

3. compute the associated initial vector,

4. extend the representation with additional states if the initial vector contains negative elements.

Step 1 is straightforward.

Step 2. In the Mocanu and Commault representation the real eigenvalues are represented with a single exponential transition (see the first block in Fig. 2) and the pairs of complex eigenvalues by Feedback-Erlang (FE) blocks.

DEFINITION 6. *[10] A* Feedback-Erlang (FE) block *with parameters (b, λ, z) is a chain of b states with transition rate λ and one transition from the bth state to the first state, with rate $z\lambda$ (c.f. Figure 1). The probability $z \in [0, 1)$ is called the* feedback probability.

Feedback-Erlang blocks with length $b = 1$ or feedback probability $z = 0$ are called *degenerate* FE blocks representing

real eigenvalues. A non-degenerate FE block where b is odd has a real eigenvalue and $(b-1)/2$ complex conjugate eigenvalue pairs. A non-degenerate FE block where b is even has 2 real eigenvalues and $(b-2)/2$ complex conjugate eigenvalue pairs. In both cases the eigenvalues are located on a circle in the complex plane whose center is $-\lambda$. The dominant eigenvalue (the one with the largest real part) of the FE block with parameters (b, λ, z) is always real and given by $r = -\lambda \left(1 - z^{\frac{1}{b}}\right)$ [10]. Given the eigenvalues $\sigma_1, \ldots, \sigma_n$ of a PH distribution whose dominant eigenvalue is σ_1, it is possible to compose FE blocks for representing these eigenvalues as follows:

- if σ_j is real, the corresponding FE block is a degenerate block; thus the parameters are:
$$\lambda_j = -\sigma_j, \quad b_j = 1, \quad z_j = 0, \tag{4}$$

- if $\sigma_j = -a_j \pm \mathrm{i}c_j$ is a complex conjugate pair, the parameters are:
$$b_i = \left\lceil \frac{2\pi}{\pi - 2\arctan\left(\frac{c_i}{a_i + \sigma_1}\right)} \right\rceil, \tag{5}$$
$$\lambda_i = \frac{1}{2}\left(2a_i - c_i \tan\frac{\pi}{b_i} + c_i \cot\frac{\pi}{b_i}\right), \tag{6}$$
$$z_i = \left(1 - \frac{a_i - c_i \tan\frac{\pi}{b_i}}{\lambda_i}\right)^{b_i}, \tag{7}$$

where $\lceil x \rceil$ denotes the smallest integer strictly greater than x.

Figure 2 depicts an example of a Markovian generator which is the unicyclic representation of a generator with a real eigenvalue (λ_1) and a pair of complex conjugate ones in FE-diagonal form. In this representation there are two FE blocks, one of length $b_1 = 1$ with rate $q_1 = \lambda_1$, and one of length $b_2 = 3$ with rate λ_2 and feedback probability z_2. The associated generator matrix is

$$\mathbf{G} = \left(\begin{array}{c|ccc} -\lambda_1 & \lambda_1 & 0 & 0 \\ \hline 0 & -\lambda_2 & \lambda_2 & 0 \\ 0 & 0 & -\lambda_2 & \lambda_2 \\ 0 & z\lambda_2 & 0 & -\lambda_2 \end{array} \right).$$

Step 3. Starting from an $(\boldsymbol{\alpha}, \mathbf{A})$ minimal non-Markovian vector-matrix description of the distribution we compute the unicyclic representation of the matrix $\mathbf{A}$, denoted as matrix $\mathbf{G}$. Additionally we compute the initial vector associated with matrix $\mathbf{G}$ as follows. Let n and m ($n \leq m$) be the size of $\mathbf{A}$ and $\mathbf{G}$, respectively. Compute matrix $\widehat{\mathbf{W}}$ of size $n \times m$ as the unique solution to

$$\mathbf{A}\widehat{\mathbf{W}} = \widehat{\mathbf{W}}\mathbf{G}, \qquad \mathbf{W}\mathbf{1} = \mathbf{1}, \tag{8}$$

and based on $\widehat{\mathbf{W}}$ the initial vector is

$$\underline{\boldsymbol{\gamma}} = \underline{\boldsymbol{\alpha}} \cdot \widehat{\mathbf{W}}. \tag{9}$$

Step 4. In case the initial vector contains negative elements, we add a so-called *Erlang-tail* to the matrix and recalculate
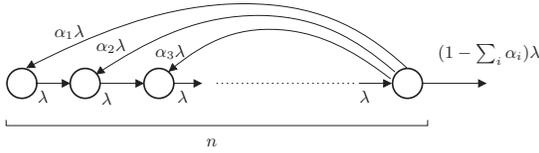
Figure 3: General unicyclic block.

the representation [10, 16]. With a suitable choice of parameters (transition rate and length of the Erlang-tail) this will always result in a Markovian representation, albeit possibly with a large size, depending on the length of the tail. We do not pursue this direction further in the present paper, instead focusing on the block structure. That said, in some cases, the initial vector is already nonnegative before this step, and this last step is not necessary.

2.4 The representation with unicyclic blocks

The term *unicyclic block* comes from [15], where it was supposed to be the structure of the whole generator matrix, not only a block of it. Unicyclic block that we use in this paper is depicted in Figure 3). It has identical intensities towards the absorbing state and may have transitions from the last phase to any previous phase. The main motivation behind the use of unicyclic blocks (UBs) is to represent more eigenvalues with a single block. An unicyclic block has more flexibility than an FE block of the same order, since an FE block is characterized by 2 parameters (λ, z) for a given order n, and an unicyclic block of order n has n parameters $(\lambda, \alpha_1, \ldots, \alpha_{n-1})$. These n parameters allow a given flexibility in representing more than one pair of complex conjugate eigenvalues with a single unicyclic block, which was not the case with FE blocks. It worth noting that an FE blocks is a special UB with $\alpha_1 = z$ and $\alpha_2 = \ldots = \alpha_{n-1} = 0$.

In [8], the authors prove that every PH distribution of size 3 has an UB(3) representation and give an analytic method to calculate the representation. However, to the best of our knowledge, similar methods (or straightforward descriptions of any kind) are unavailable for larger UB. In this work we investigate the flexibility of larger UBs through a numerical method. The complexity and the numerical instability of the applied numerical method increases quickly with the block size, hence we focus on block sizes of 3 (for verifying the procedure), 4, and 5.

It is reported many times (e.g. in [8]) that the eigenvalues of an order 3 PH distribution should be in a triangular of angle $\pi/3$ staring from the dominant eigenvalue as it is in Figure 4. An FE block of order 3 can represent complex conjugate eigenvalue pairs only on the border of this triangular, while a unicyclic block of order 3 can represent any complex conjugate eigenvalue pairs in the triangular. Indeed Figure 4 depicts the feasible region of the complex eigenvalue pairs computed by our numerical procedure when the dominant eigenvalue is -1. The figure also demonstrates that the unicyclic block structure is flexible enough to capture the full flexibility of the complex eigenvalues of order 3 PH distributions.

Our numerical procedure works as follows. We compute the

roots of the characteristic polynomial of matrix

$$
\begin{pmatrix}
-\lambda & \lambda & \cdots & 0 \\
0 & -\lambda & \ddots & 0 \\
0 & 0 & \ddots & \lambda \\
\alpha_1\lambda & \alpha_2\lambda & \cdots & -\lambda
\end{pmatrix}
$$

as a function of $\lambda, \alpha_1, \ldots, \alpha_{n-1}$ and set the parameters such that the roots are the required eigenvalues. Symbolic solutions are available for order 3 and 4, but we use numerical solver in all cases. In case of UB(3) we set one real and one conjugate pair, in case of UB(4) we set two reals and one conjugate pair, and in case of UB(5) we set one real and two conjugate pairs. If the obtained $\lambda, \alpha_1, \ldots, \alpha_{n-1}$ parameters are non-negative the UB is Markovian and exhibits the required eigenvalues.

With the help of this numerical procedure we can depict some eigenvalue regions which can be represented by UB(3), UB(4) and UB(5), such that we fix some eigenvalues and plot the feasible region of the remaining one (which is a conjugate pair in each cases). As already mentioned, the results for UB(3) are aligned with the theory. Feasible regions of the eigenvalues of Markovian UB(4) blocks are depicted in Figure 5. The fixed eigenvalues are depicted with small circles. The order 4 FE block can represent only the conjugate pair at the top and bottom right points, $-1.5 \pm 0.5i$.

Comparing Figure 4 and 5, see Figure 6, we can easily compose an order 4 counterexample of O'Cinneide's unicyclic representation conjecture [15] for which [8] presented an order 6 counterexample. E.g., if the eigenvalues of an order 4 PH are $-1, -2, -3 \pm 0.5i$ then one can represent it by an UB(3) and a single exponential, but cannot represent it as an UB(4). Indeed the intersection of the regions in Figure 4 and 5 are such that the eigenvalues can be represented by both an UB(3) with an additional exponential and an UB(4). An UB(4) block needs only when the complex eigenvalue pair is in the UB(4)\UB(3) part.

We can interpret this as a qualitative difference between the order 3 and the higher order UBs. Namely, fixing one eigenvalue of an UB(4) additional to the dominant one makes its feasible region bounded, while feasible region of UB(3) is unbounded.

Figures 7 - 11 demonstrate the change of the feasible region of an UB(5) block, when the dominant eigenvalue is -1 the real part of the fixed complex conjugate eigenvalue is -1.8 and its imaginary part changes between 0.25 and 0.6. The numerical errors appearing when the imaginary part is close to zero cause the white zones inside the continuous regions. Similar to the UB(4) case fixing the dominant eigenvalue and a complex conjugate pair limits the remaining eigenvalue pair to a bounded region.

The results on Figures 7 - 11 can not be related with the eigenvalues of the order 5 FE blocks, because none of the considered fixed eigenvalue arrangements is according to the angle of the regular pentagon, which is the case with order 5 FE blocks.
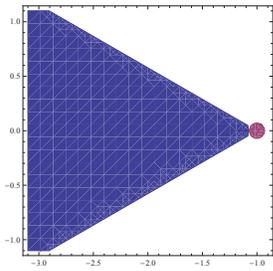
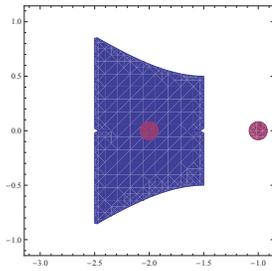**Figure 4: UB(3) region with real eigenvalue −1.**
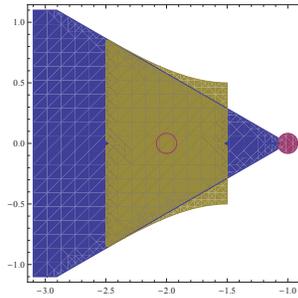

Figure 5: UB(4) region with −1, −2.
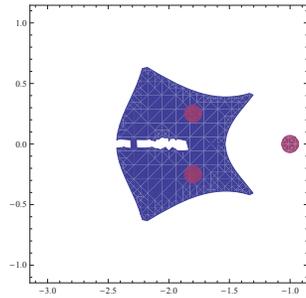

Figure 6: Intersection of UB(3) and UB(4)

Figure 7: UB(5) region with −1, −1.8 ± 0.25i.

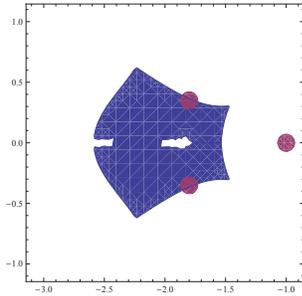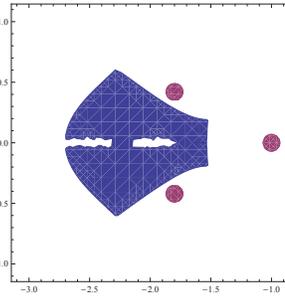**Figure 8: UB(5) region with −1, −1.8 ± 0.35i.**

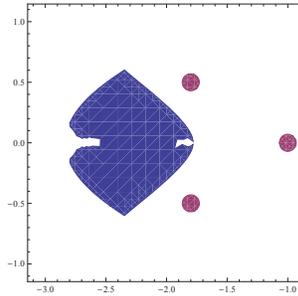

Figure 9: UB(5) region with −1, −1.8 ± 0.42i.
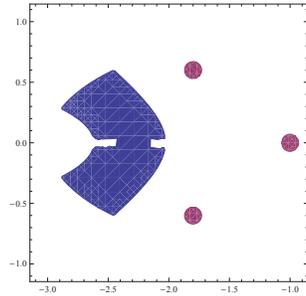

Figure 10: UB(5) region with −1, −1.8 ± 0.5i.

Figure 11: UB(5) region with −1, −1.8 ± 0.6i.

The consideration about the unbounded UB(3) and bounded UB(4) blocks anticipate one of the main points of the present paper that unicyclic blocks of size 3 are all that are necessary in the majority of cases. In the next section, we test the effectiveness of the FE block structure and the UB structure in different settings.

3. EXPERIMENTAL RESULTS

There are a number of questions to be examined. The most important one is, of course, how much is to be gained by introducing the unicyclic blocks compared to pure FE blocks. This can be measured in several ways.

In subsection 3.1, we will focus on obtaining minimal representations with unicyclic blocks. To obtain a large sample of random PH distributions, the function `randomPH` from the package BuTools is used [4] under Wolfram Mathematica.

We realize that there is no single best way to generate random samples; for a more thorough experimental analysis, we decided to include a second round of tests. Both FE and unicyclic block representations are based on the eigenvalues of matrix **A**, and the number of complex eigenvalues is a major factor in the size of the representation obtained. To that end, in Subsection 3.2 we tested both types of representations in a setting where only the set of eigenvalues is generated, with the ratio of complex eigenvalues from among all eigenvalues as a scalable parameter. While this only tests the matrix part of the representation, it still gives a good idea in how does the size of a unicyclic block structure representation compare to the FE block structure representation,

depending on the number of complex eigenvalues.

The two main statistics to be examined is the average size of representations found and the percentage of the cases where a minimal representation was found. In Section 3.1, the average is omitted due to the omission of a tail procedure (see Step 4 in Section 2.3); without a tail procedure, in some cases no representation will be found, and thus any form of average would be inherently distorted. Instead, we rely on the percentage of minimal representations. This issue is partly remedied by Section 3.2.

All representations were calculated numerically. Working precision is 10 decimal digits. We note that the numerical solver works fine up to block size 5 (which means solving an equation system of degree 4), but gets unreliable above that.

3.1 Experiments with PH samples

In the following experiments, a large set of random PH samples were generated. `randomPH` is capable of generating PH distributions along with minimal Markovian representations. While starting from a minimal Markovian representation may seem counterintuitive at first, this has the advantage that the size of the minimal Markovian representation is known upfront, making the results easier to evaluate. Also, the random matrices generated by `randomPH` are dense, but the canonical representations obtained by either form of block structure are sparse, making them more efficient for actual Markovian modeling applications.

Then each sample is tested by four algorithms. In all algorithms, the blocks are determined first. `algo1` uses the FE block structure: it represents each real eigenvalue by a block of size 1 (referred to as a *real block* from now on) and each complex eigenvalue by an FE block whose size is determined by (5). `algo2` also searches for UB(3), prioritizing it over FE and real blocks. When a block is found, the corresponding eigenvalues are removed from the set of all eigenvalues, and the search continues for the rest of the eigenvalues until all eigenvalues are represented. `algo3` works similarly, but searches for UB(3) and also UB(4), with UB(3) getting higher priority. The reason behind this choice of priority is due to the fact that a block of size 4 may be feasible even when a block of size 3 is not, but a block of size 4 represents 2 real eigenvalues instead of 1 (in other words, real eigenvalues may be regarded as "resources" to construct unicyclic blocks, and UB(3) uses less of those resources). `algo4` searches for UB(5), UB(3), UB(4), FE and real blocks in this order (UB(5) represents 2 complex and 1 real eigenvalues, thus being the most resource-efficient in the above sense).

For each algorithm, once the blocks are determined, they still need to be arranged in some way to obtain a full representation. By this we refer to the order of the blocks. To limit the complexity of the experiments we assume that absorption is only possible from the last phase of the last block.

The initial vector corresponding to the obtained FE or unicyclic representation is uniquely determined by the ordered blocks. The initial vector needs to be checked for nonnegativity; if it is nonnegative, then the obtained representation is Markovian. If the initial vector contains at least one negative element, the obtained representation is not Markovian. Methods are available to remedy this situation (e.g. by the addition of an Erlang-tail, see [10, 16]), but they typically involve the addition of extra phases. We do not pursue this direction, and as a result, the algorithms may fail to find a Markovian representation due to the negativity of the initial vector. As we shall see, this is not a significant issue in our experiments.

The ordering of the blocks is important in that it greatly affects whether the initial vector will be Markovian or not. For acyclic PH distributions (this corresponds to the subclass where all blocks are real and of size 1), the right ordering is when blocks are in arranged in increasing order of absolute value [6]. Based on this, we expect that for unicyclic representations, increasing order of the absolute value of the diagonal element will work well. Nevertheless, we test the reverse ordering as well.

The random generator `randomPH` has one more interesting feature: the number of *restrictions* (# restr.) is also a scalable parameter. Restrictions in this case are the number of prescribed 0 elements in either the matrix or the vector. Several samples with a different number of restrictions will be tested.

The first test focuses on testing the right ordering; each of the 4 algorithms with both orderings will be tested on the same sample set. `algo1`, `algo2`, `algo3` and `algo4` denote the versions of the algorithms (as defined above) with the

blocks in increasing order and `algo1b`, `algo2b`, `algo3b` and `algo4b` denote the same algorithms but with the blocks in decreasing order.

In the first experiment 1000 random PH distributions were used; the minimal size is 10 and the number of restrictions is 0. The results show the number of minimal Markovian representations (min. size) found by each of the algorithms out of 1000.

size=10	# min. repr.
`algo1b`	0
`algo2b`	0
`algo3b`	0
`algo4b`	0
`algo1`	4
`algo2`	683
`algo3`	700
`algo4`	710

As can be seen from the results, the algorithms using decreasing ordering are absolutely inefficient, failing to produce even one single minimal Markovian representation (in fact, decreasing ordering versions did not produce *any* Markovian representations at all). The increasing ordering, on the other hand, works quite nicely. `algo1` finds representations that are typically larger than minimal (although it should be noted that it does find a Markovian representation of some size most of the time). `algo2`, `algo3` and `algo4` find a minimal unicyclic representation at around 70% of the time. The huge jump is between `algo1` and `algo2`, i.e. most of the improvement is due to the introduction of UB(3). The most likely explanation is that UB(3) is capable of representing a complex eigenvalue and a real eigenvalue with the minimal number of phases possible, while FE blocks need at least one extra phase for each complex eigenvalue.

From this point on, all tests are done solely for the increasing order versions of the algorithms.

In the next experiment, 5000 random PH distributions were generated with size 10 and the number of restrictions set to 0, 20 and 60 (low/medium/high) respectively. The results show the number of minimal representations (monocyclic for `algo1`, unicyclic for `algo2`–`algo4`) found by each of the algorithms out of 5000.

size=10	# of restrictions		
	0	20	60
`algo1`	16	186	450
`algo2`	3485	4419	3863
`algo3`	3556	4430	3884
`algo4`	3640	4426	3875

As can be seen, all of the algorithms improve when the number of restrictions is set from 0 to 20. This is possibly due to the number of complex eigenvalues typically being lower in this case (again, we remark that it is necessary that *all* eigenvalues are real in order for `algo1` to find a minimal representation). Also, the difference between between

`algo2`, `algo3` and `algo4` is very small. `algo2`–`algo4` performed slightly worse when the number of restrictions was increased to 60. We do not have an intuitive explanation for this phenomenon. There is one more interesting issue: in some cases, `algo4` performed slightly worse than `algo3`. This might be due to the preference settings of `algo4`; while UB(5) contains two pairs of complex eigenvalues, and thus seems generally more desirable, it is possible that a block of size 5 found in actual examples represents two pairs of complex eigenvalues with real parts far from the dominant eigenvalue, plus a real eigenvalue that is either the dominant or close to it. Instead, two UB(3) or UB(4) blocks with real eigenvalues farer from the dominant eigenvalue might be a more optimal choice (we found examples of this case).

In the next experiment, 2000 random PH distributions were generated with size 20 and the number of restrictions set to 0 and 100 respectively. The results show the number of minimal representations (monocyclic for `algo1`, unicyclic for `algo2`–`algo3`) found by each of the algorithms out of 2000.

size=20	# of restrictions	
	0	100
`algo1`	0	0
`algo2`	471	1821
`algo3`	474	1821

A minimal representation is much less likely for these larger-sized PH distributions, but, while `algo1` is unable to find a minimal representation at all, `algo2` and `algo3` still work for a portion of the samples. The number of restrictions is again very important, making `algo2` and `algo3` much better in finding a minimal representation when the number of restrictions is set to 100.

3.2 Experiments with eigenvalue samples

In this section, a list of eigenvalues is generated randomly. Initial vectors were not tested; the main point is to introduce the number of complex eigenvalues as a scalable parameter. We also tested not just for the number of complex eigenvalues, but also for "how" complex they are, i.e. how large are the imaginary parts of complex eigenvalues.

The samples were generated according to the following:

- the minimal size of each representation is 10 (thus 10 eigenvalues are generated);

- the dominant eigenvalue is -1;

- the rest of the eigenvalues have real part $-3 - 5|N_r|$, where N_r is standard normal distributed;

- the number of complex pairs of eigenvalues is 1 to 4; with imaginary parts $|N_i|$ (smaller) or $5|N_i|$ (larger), where N_i is standard normal distributed.

This is a total of 8 experiments, each containing 5000 randomly generated set of 10 eigenvalues. We tested not only for minimal representation, but also for the average size of a representation found (which is more informative in certain cases).

The first table contains the experiments where the imaginary parts are "small", and the number of complex eigenvalues goes from 1 to 4.

size=10	# of complex *smaller* pairs			
	1		2	
	# min.	avg.	# min.	avg.
`algo1`	0	11.22	0	12.47
`algo2`	4101	10.40	866	11.34
`algo3`	4589	10.21	1074	11.22
`algo4`	4589	10.21	2934	10.80
size=10	# of complex *smaller* pairs			
	3		4	
	# min.	avg.	# min.	avg.
`algo1`	0	13.69	0	14.91
`algo2`	43	12.49	0	13.82
`algo3`	46	12.46	0	13.82
`algo4`	554	11.61	25	12.74

The first case is where the unicyclic representation really shines. It reduces the size of the representation considerably compared to the FE representation, and finds a minimal representation matrix most of the time. Also, that is case closest to the actual distribution of eigenvalues resulting from `randomPH`. Also, the difference between `algo3` and `algo4` vanishes when the number of complex eigenvalues is low but becomes more important when the number of complex eigenvalues is higher. In the case of 4 complex pairs of eigenvalues, practically none of the algorithms were able to give a minimal representation matrix (except `algo4` in very few cases), but the average size of the representation is still relevant. While these case of high number of complex pairs of eigenvalues are less typical for `randomPH` generated PH distributions, the unicyclic algorithms still outperform the FE one.

The next table contains the experiments with larger imaginary parts.

size=10	# of complex *larger* pairs			
	1		2	
	# min.	avg.	# min.	avg.
`algo1`	0	12.69	0	15.28
`algo2`	2300	12.23	831	14.40
`algo3`	2789	12.04	1103	14.12
`algo4`	2789	12.04	1192	14.10
size=10	# of complex *larger* pairs			
	3		4	
	# min.	avg.	# min.	avg.
`algo1`	0	18.11	0	20.66
`algo2`	163	16.93	0	19.41
`algo3`	186	16.73	0	19.37
`algo4`	310	16.61	12	19.02

Our findings are somewhat similar to the previous case; the main conclusion here is that all algorithms perform worse

in the presence of complex eigenvalues with typically larger imaginary parts.

Ackonwledgement

4. CONCLUSION

Due to the higher flexibility of the unicyclic blocks with respect to the FE blocks the Markovian representation of PH distribution based on unicyclic blocks is commonly smaller than the one based on FE blocks. The benefit depends on the particular case. Most of the improvement comes with the introduction of order 3 unicyclic blocks, while the improvement coming from the introduction of larger unicyclic blocks is marginal. Order 3 unicyclic blocks are thus an efficient and computationally tractable alternative to Feedback-Erlang blocks when searching for a small Markovian representation of PH distributions. The minimality of the representation is not tackled in this paper. The examples staring from sampled Markovian PH representations indicate that our procedure does not find a minimal representation in some of the cases.

5. REFERENCES

[1] Soren Asmussen and Mogens Bladt. Point processes with finite-dimensional conditional probabilities. *Stochastic Processes and their Applications*, 82(1):127 − 142, 1999.

[2] L. Benvenuti. Minimal positive realizations of transfer functions with real poles. *Automatic Control, IEEE Transactions on*, 58(4):1013–1017, April 2013.

[3] L. Benvenuti and L. Farina. A tutorial on the positive realization problem. *Automatic Control, IEEE Transactions on*, 49(5):651–664, May 2004.

[4] Levente Bodrog, Peter Buchholz, Armin Heindl, András Horváth, Gábor Horváth, István Kolossváry, Zoltán Németh, Philipp Reinecke, Miklós Telek, and Miklós Vécsei. Butools: Program packages for computations with PH, ME distributions and MAP, RAP processes. http://webspn.hit.bme.hu/~butools, October 2011.

[5] Peter Buchholz and Miklós Telek. On minimal representation of rational arrival processes. *Annals of Operations Research*, 2011. DOI 10.1007/s10479-011-1001-5.

[6] A. Cumani. On the canonical representation of homogeneous Markov processes modelling failure-time distributions. *Microelectronics and Reliability*, 22:583–602, 1982.

[7] Lorenzo Farina. Minimal order realizations for a class of positive linear systems. *Journal of the Franklin Institute*, 333(6):893 − 900, 1996.

[8] Qi-Ming He and Hanqin Zhang. A note on unicyclic representations of phase type distributions. *Stochastic Models*, 21(2-3):465–483, 2005.

[9] Gábor Horváth and Miklós Telek. On the canonical representation of phase type distributions. *Performance Evaluation*, 66(8):396 − 409, 2009.

[10] S. Mocanu and C. Commault. Sparse representations of phase-type distributions. *Stochastic Models*, 15:759–778, 1999.

[11] B. Nagy and M. Matolcsi. Minimal positive realizations of transfer functions with nonnegative multiple poles. *Automatic Control, IEEE Transactions on*, 50(9):1447–1450, Sept 2005.

[12] M. Neuts. Probability distributions of phase type. In *Liber Amicorum Prof. Emeritus H. Florin*, pages 173–206. University of Louvain, 1975.

[13] M.F. Neuts. *Matrix Geometric Solutions in Stochastic Models*. Johns Hopkins University Press, Baltimore, 1981.

[14] Colm A. O'Cinneide. Characterization of phase-type distributions. *Communications in Statistics - Stochastic Models*, 6(1):1–57, 1990.

[15] Colm A. O'Cinneide. Phase-type distributions: open problems and a few properties. *Communications in Statistics - Stochastic Models*, 15(4):731–758, 1999.

[16] P. Reinecke and M. Telek. Does a given vector-matrix pair correspond to a ph distribution? *Performance Evaluation*, 80(0):40 − 51, 2014.

[17] Zhizhen Wang, Long Wang, Wensheng Yu, and Guoping Liu. Minimal positive realizations of a class of third-order systems. In *American Control Conference, 2004. Proceedings of the 2004*, volume 5, pages 4151–4152 vol.5, June 2004.