# Dealing with Zero Density Using Piecewise Phase-Type Approximation*

L'uboš Korenčiak[1], Jan Krčál[2], and Vojtěch Řehák[1]

[1] Faculty of Informatics, Masaryk University, Brno, Czech Republic
{korenciak,rehak}@fi.muni.cz

[2] Saarland University – Computer Science, Saarbrücken, Germany
krcal@cs.uni-saarland.de

**Abstract.** Every probability distribution can be approximated up to a given precision by a phase-type distribution, i.e. a distribution encoded by a continuous time Markov chain (CTMC). However, an excessive number of states in the corresponding CTMC is needed for some standard distributions, in particular most distributions with regions of zero density such as uniform or shifted distributions. Addressing this class of distributions, we suggest an alternative representation by CTMC extended with *discrete-time* transitions. Using discrete-time transitions we split the density function into multiple intervals. Within each interval, we then approximate the density with standard phase-type fitting. We provide an experimental evidence that our method requires only a moderate number of states to approximate such distributions with regions of zero density. Furthermore, the usage of CTMC with discrete-time transitions is supported by a number of techniques for their analysis. Thus, our results promise an efficient approach to the transient analysis of a class of non-Markovian models.

## 1 Introduction

In the area of performance evaluation and probabilistic verification, discrete-event systems (DES) are a prominent modelling formalism. It includes models such as continuous-time Markov chains, stochastic Petri nets, or generalized semi-Markov processes. A DES is a random process that is initialized in some state and then moves from state to state in continuous-time whenever an event occurs. Every time a state is entered, some of the events get *initiated*. An initiated event then occurs after a delay chosen randomly according to its distribution function. When no restrictions on the distribution functions are imposed, analysis of these models is complicated [8,21], one often resorts to simulation [17]. When all the distributions $F_e$ are exponential, the DES is then called

* This work is supported by the EU 7th Framework Programme under grant agreements 295261 (MEALS) and 318490 (SENSATION), Czech Science Foundation grant No. P202/12/P612, the DFG Transregional Collaborative Research Centre SFB/TR 14 AVACS, and by the CAS/SAFEA International Partnership Program for Creative Research Teams. Access to computing and storage facilities owned by parties and projects contributing to the National Grid Infrastructure MetaCentrum, provided under the programme "Projects of Large Infrastructure for Research, Development, and Innovations" (LM2010005), is greatly appreciated.

A. Horváth and K. Wolter (Eds.): EPEW 2014, LNCS 8721, pp. 119–134, 2014.

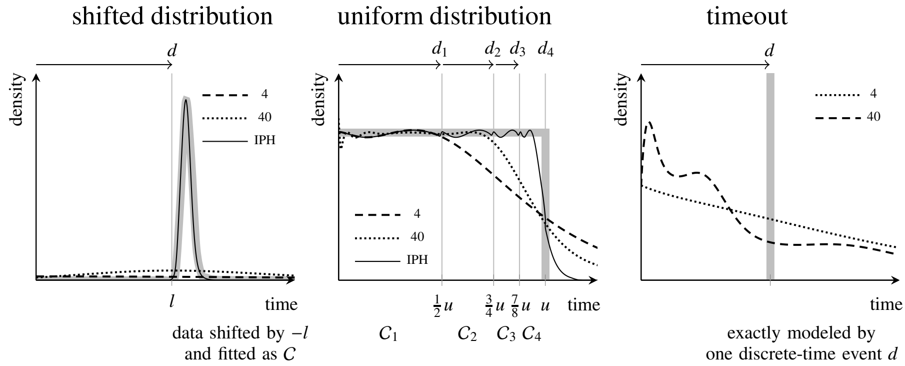


**Fig. 1.** Three usages of the discrete-time events for PH approximation. In the figures there are the densities (with thick grey lines), their standard PH approximations with 4 and 40 phases, and their IPH approximation with 30 phases. On the left, the discrete-time event $d$ postpones the start of the CTMC $C$ fitted to the area of positive density. On the right, the discrete-time event can be used directly, instead of its continuous approximation. In the middle, 3 discrete-time events split the support into 4 intervals with different approximations $C_1$, $C_2$, $C_3$, and $C_4$. Note that for a distribution with a steep change in density at its upper bound (such as the uniform distribution), PH fitting performs well on the first *half* of the support; logarithmic partitioning into intervals works better than equidistant.

a continuous-time Markov chain (CTMC) for which many efficient analysis methods exist [26,4] thanks to the memoryless property of the exponential distribution. Hence, an important method for analysing DES is to *approximate* it by a CTMC using *phase-type* (PH) approximation and to solve resulting CTMC analytically. Roughly speaking, each event $e$ such that its distribution function is not exponential is replaced by a small CTMC $C_e$. This CTMC has a designated absorbing state such that the time it takes to reach the absorbing state is distributed as closely as possible to the given distribution function. A well known result [34] states that any continuous probability distribution can be fitted up to a given precision by the PH approximation. Nevertheless, the closer the approximation, the more states it requires in the CTMC. For some lower bounds on the number of required states see, e.g., [1,35,13,12].

In this paper we propose another approach for approximating probability distributions where phase-type requires extreme amount of states to be fitted precisely [35,12]. In particular, we deal with distributions often encountered in practice that we call *interval distributions* and that are supported on a proper subinterval of $[0,\infty)$. For example distributions of events that cannot occur before time $l > 0$ such as due to physical limits when *sending a packet*; or that cannot occur after time $u < \infty$ such as *waiting for a random amount of time* in a collision avoidance protocol; or that occur exactly after time $l = u$ such as *timeouts*. We address these interval distributions by an approach that we call *Interval phase-type (IPH)* approximation. The crucial point is that it allows to separate the discrete and the continuous nature of these distributions by enriching the output formalism. Along with the exponential distribution of the CTMC we allow discrete-time events (also called fixed-delay, deterministic, or timeout events) and denote it as d-CTMC.[1] As illustrated in Figure 1, the usage of discrete-time events for approximating a non-exponential distribution is threefold:

1. For an event $e$ with occurrence time bounded from below by $l > 0$, an occurrence of a discrete-time event $d$ splits the waiting into two parts – an initial part of length $l$ where the event $e$ cannot occur and the rest that can be more efficiently approximated by a CTMC $C$ using standard PH methods.
2. For an event $e$ with occurrence time bounded from above by $u < \infty$, a series of discrete-time events partition the support of its distribution into $n$ subintervals. The system starts in the chain $C_1$ which is the standard PH approximation of the whole density. In parallel to movement in $C_1$ a discrete-time event $d_1$ is awaited with its occurrence set to the beginning of the second interval. If the absorbing state in $C_1$ is not reached before $d_1$ occurs, the system moves to $C_2$. The chain $C_2$ is fitted to the whole remaining density conditioned by the fact that the event does not occur before the beginning of the second interval. Similarly, another discrete-time event $d_2$ is awaited in $C_2$ with its occurrence set to the beginning of the third interval, etc. The last interval is not ended by any discrete-time event; occurrence of the event $e$ thus corresponds to reaching any absorbing state in any of $C_1, \dots, C_n$.
3. An event with constant occurrence time ($\ell = u$) is directly a discrete-time event.

*Example.* As our running example, we consider the Alternating bit protocol. Via a lossy FIFO channel, a transmitter attempts to send a sequence of messages, each endowed with a one-bit sequence number – alternating between 0 and 1. The transmitter keeps resending each message until it is acknowledged by its sequence number (the receiver sends back the sequence number of each incoming message). As resending of messages is triggered by a timeout, setting an appropriate value for the timeout is essential in balancing the performance of the protocol and the network congestion. For a given timeout, one may ask, e.g., *what is the probability that* 10 *messages will be successfully sent in* 100*ms?* In the next section we show a simple DES model of this protocol. Subsequently, we show the CTMC model yielded by a PH approximation of individual events, and the d-CTMC model obtained by our proposed IPH approximation.

*Our Contribution.* We propose an alternative approach to PH approximation, resulting in a CTMC enriched with fixed-delay events. Our approach is tailored to interval probability distributions that are often found in reality and for which the standard continuous PH approximation requires a substantial amount of states. We performed an experimental evaluation of our approach. In the evaluation, we represent (1) the lower-bounded distributions by the distribution of the *transport time in network communication* and (2) the upper-bounded distributions by the *uniform* distribution. For both cases, we show that our approach requires only a moderate number of states to approximate these distributions up to a given error. Thus, for DES models with interval distributions our approach promises a viable method for transient analysis as also indicated by our experiments.

[1] Note that the formalism of d-CTMC is inspired by the previously studied similar formalisms of *deterministic and stochastic Petri nets* [32] and *delayed CTMC* [16].

*Related Work.* Already in the original paper of Neuts [34], the fixed-delay and shifted exponential distributions have been found difficult to fit with a phase-type approximation. This fact was explicitly quantified by Aldous and Shepp [1] showing that the Erlang distribution is the best PH fitting for the fixed-delay distributions. A notoriously difficult example of a shifted distribution is the data set measuring the length of eruptions of a geyser in the Yellowstone National Park [38] whose PH approximation has been discussed in, e.g., [3,13]. Also heavy tailed distributions often found in telecommunication systems are hard to fit; similarly to our method, separate fitting of the body and the tail of such distributions is used [14,23].

Apart from continuous PH fitting, there are several other methods applicable to analysis of DES with interval distributions. First, there are several symbolical solution methods [2,5,20,21] for direct analysis of DES with non-exponential events. Usually, *expolynomial* distributions are allowed; non-expolynomial distributions need to be fitted by expolynomials – a problem far less studied than standard PH fitting. Our approach can be understood as a specific fitting technique that uses a limited subclass of expolynomial distributions (resulting in models with a wider range of analysis techniques). Second, interval distributions can be efficiently fitted by discrete phase-type approximation [6]. Instead of a CTMC, this method yields a *discrete-time* Markov chain (DTMC) where each discrete step corresponds to elapsing some fixed $\delta$ time units. Note however that this method usually requires to discretize *all the events* of a DES into a DTMC. To analyse faithfully a DES with many parallel events one either needs to use a very small $\delta$ [40] or to allow occurrence of multiple events within each $\delta$-time step [33,19], exponentially increasing the amount of states or transitions in the DTMC, respectively. Third, similarly to our approach, ideas for combining discrete PH approximation with continuous PH approximation have already appeared [27,18]. To the best of our knowledge, no previous work considers combining these two approaches on *one* distribution having both discrete and continuous "nature". Expressing the continuous part of such a distribution using continuous PH again decreases the coincidence of parallel discrete events discussed above. Note that with d-CTMC, one can freely combine continuous PH, discrete PH, and interval PH for approximation of different events of a DES.

*Organization of the Paper.* In Section 2, we define the necessary preliminaries. In Section 3, we describe the IPH approximation method and briefly review the analysis techniques for d-CTMC. The paper is concluded by an experimental evaluation in Section 4.

## 2 Preliminaries

We denote by $\mathbb{N}$, $\mathbb{Q}$, and $\mathbb{R}$ the sets of natural, rational, and real numbers, respectively. For a finite set $X$, $\mathcal{D}(X)$ denotes the set of all discrete probability distributions over $X$.

**Modelling Formalisms.** There are several equivalent formalisations of DES. Here we define *generalized-semi Markov processes* that contain both CTMC and d-CTMC as subclasses. Let $\mathcal{E}$ be a finite set of *events* where each event is either a *discrete-time* event or a *continuous-time event*. To each discrete-time event $e$ we assign its delay $delay(e) \in \mathbb{Q}$. To each continuous-time event $e$ we assign a *probability density function*

$f_e : \mathbb{R} \to \mathbb{R}$ such that $\int_0^\infty f_e(x)\,dx = 1$. An event is called *exponential* if it is a continuous-time event with density function $f(x) = \lambda \cdot e^{-x\lambda}$ where $\lambda > 0$ is its rate.

**Definition 1.** *A* generalized semi-Markov process *(GSMP) is a tuple* $(S, \mathcal{E}, \mathbf{E}, \mathrm{Succ}, \alpha_0)$ *where*

- $S$ *is a finite set of* states*,*
- $\mathcal{E}$ *is a finite set of* events*,*
- $\mathbf{E} : S \to 2^{\mathcal{E}}$ *assigns to each state* $s$ *a set of events* active *in* $s$*,*
- $\mathrm{Succ} : S \times \mathcal{E} \to \mathcal{D}(S)$ *is the* successor *function, i.e. it assigns a probability distribution specifying the successor state to each state and event that occurs there,*
- $\alpha_0 \in \mathcal{D}(S)$ *is the* initial distribution.

*We say that a GSMP is a* continuous-time Markov chain *(CTMC) if every events of* $\mathcal{E}$ *is exponential. We say that a GSMP is a* continuous-time Markov chain with discrete-time events *(d-CTMC) if every event of* $\mathcal{E}$ *is either exponential or discrete-time.*

The run of a GSMP starts in a state $s$ chosen randomly according to $\alpha_0$. At start, each event $e \in \mathbf{E}(s)$ is *initialized*, i.e. the amount of time *remain*$(e)$ remaining until it occurs is (1) set to *delay*$(e)$ if $e$ is a discrete-time event, or (2) chosen randomly according to the density function $f_e$ if $e$ is a continuous-time event. Let the process be in a state $s$ and let the event $e$ have the minimal remaining time $t = \mathit{remain}(e)$ among all events active in $s$. The process waits in $s$ for time $t$ until the event $e$ occurs, then the next state $s'$ is chosen according to the distribution $\mathrm{Succ}(s, e)$[2]. Upon this transition, the remaining time of each event of $\mathbf{E}(s) \setminus \mathbf{E}(s')$ which is not active any more is discarded, and each event of $\mathbf{E}(s') \setminus \mathbf{E}(s)$ is initialized as explained above. Furthermore if the just occurred event $e$ belongs to $\mathbf{E}(s')$, it is also initialized. For a formal definition we refer to [8].

*Example (continued)* To illustrate the definition, Figure 2 shows on the left a simplified GSMP model of the Alternating bit protocol. The transmitter sending a message corresponds to the exponential event send. The whole remaining process of the message being transported to the receiver, the receiver sending an acknowledgement message and the acknowledgement message being transported back to the transmitter is modelled using one continuous-time event ack. In parallel with the event ack, there is a discrete-time event timeout and an exponential event err representing a packet loss.

To exemplify the semantics, assume the process is in the state sent with *remain*(timeout) = 10, *remain*(ack) is chosen randomly to 12.6 and *remain*(err) is chosen randomly to 7.2. Hence, after 7.2 time units the event err occurs and the process moves to the state lost with *remain*(timeout) = 2.8. After further 2.8 time units, the timeout elapses and the process moves to the state init where *remain*(send) is chosen randomly to 0.8. After this time, the process moves to send where *remain*(timeout) is again set to 10 and *remain*(ack) and *remain*(err) are again sampled according to their densities and so on. In the next section, we show the PH approximation of this model.

[2] For the sake of simplicity, when multiple events $X = \{e_1, \ldots, e_n\}$ occur simultaneously, the successor is determined by the minimal element of $X$ according to some fixed total order on $\mathcal{E}$. A more general definition [8] allows to specify different behaviour for simultaneous occurrence of any subset of events.

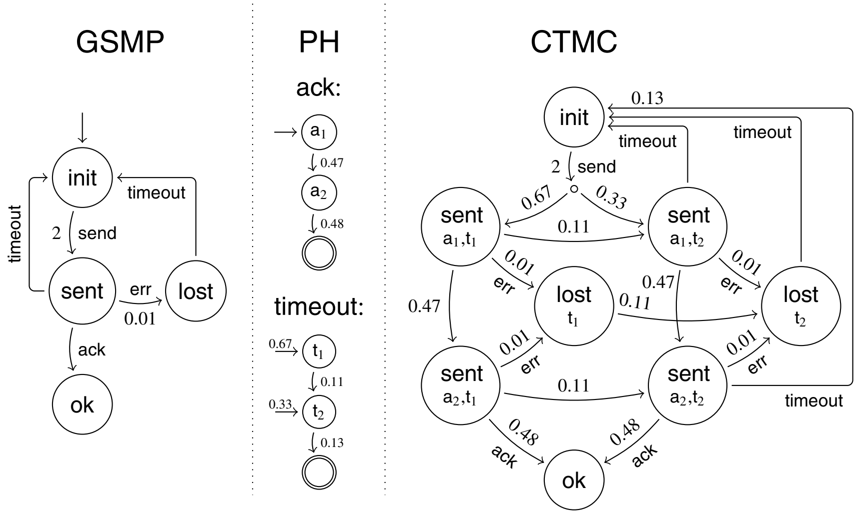


**Fig. 2.** On the left, there is a GSMP model of sending *a single* message using the the Alternating bit protocol. The set of events active in a state corresponds to the edges outgoing from that state. The event timeout is discrete-time with delay 10 ms, send is exponential with rate 2 meaning that it takes 0.5 ms on average to send a message, err is exponential with rate 0.01 corresponding to a packet being lost each 100 ms of network traffic on average, and ack is continuous-time with density displayed in Figure 4 on the left. In the middle, there are 2-phase PH approximations of events ack and timeout. On the right, there is a PH approximation of the GSMP model obtained roughly speaking as a product of the GSMP and the two PH components.

**Continuous PH Approximation.** Continuous PH can be viewed as a class of algorithms

- which take as input the number of phases $n \in \mathbb{N}$ and a probability density function $f$ of a positive random variable, and
- output a CTMC $C$ with states $\{0, 1, \ldots, n\}$ where 0 is an absorbing[3] state.

Any such CTMC $C$ defines a positive random variable $X$ expressing the time when the absorbing state 0 is reached in $C$. Let $\hat{f}$ denote the probability density function of $X$. A possible goal of a PH algorithm is to minimize the *absolute density difference* [7][4]

$$\int_0^\infty |f(x) - \hat{f}(x)|\, dx. \tag{Err}$$

*Example (continued)* When building a CTMC model of the Alternating bit protocol from the GSMP model, we need to approximate the non-exponential events ack and timeout. Their simple approximation and the whole CTMC model of the system is depicted in Figure 2 on the right. Observe that each state of the whole model needs to be enriched with the phase-number of every non-exponential event scheduled in this state. The events are then defined in a natural way on this product state space.

[3] We say that a state $s$ is *absorbing* if there are no outgoing transitions, i.e. $\mathbf{E}(s) = \emptyset$.

[4] Note that there are PH methods that do not allow specifying the number of phases. For further metrics for evaluating quality of PH approximation, see, e.g., [7].

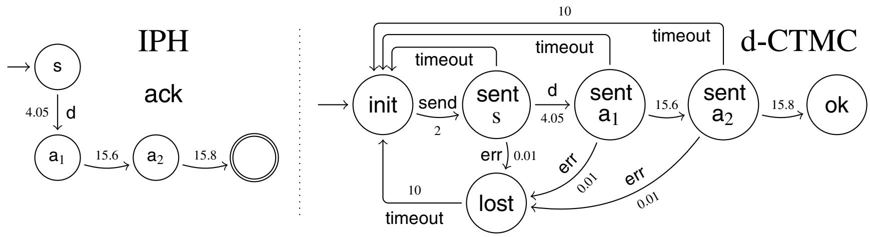


**Fig. 3.** On the left, there is an IPH approximation of the event ack using the algorithm `IPH-shift[PhFit]` with 3 phases. The discrete event $d$ has delay 4.05. On the right, there is the whole d-CTMC with discrete-time events timeout and d. The model is obtained similarly as the CTMC in Figure 2.

In the next section we describe our extension of PH fitting with discrete-time events.

## 3 Interval Phase-Type Approximation

The Interval phase-type (IPH) approximation addresses the interval probability distributions which are supported on a proper subinterval of $[0, \infty)$. Similarly as above,

- it takes as input the number of phases $n \in \mathbb{N}$ and a probability density function $f$ of a positive random variable, and
- outputs a d-CTMC $\mathcal{D}$ with states $\{0, 1, \ldots, n\}$ where 0 is absorbing.

The goal is again to minimize (Err) for $\hat{f}$ being the probability density function[5] of the random variable $X$ expressing the time when the absorbing state 0 is reached in $\mathcal{D}$.

### 3.1 Constructing d-CTMC

As the first step in this alternative direction, we provide two basic techniques that significantly decrease the error for interval distributions (compared to standard PH algorithms that are by definition IPH algorithms as well). The first technique deals with interval distributions bounded from below.

**Delay Bounded from Below.** For an event that cannot occur before some $l > 0$ and for a given number of phases $n > 1$, our algorithm works as follows. Let $C = (S, \mathcal{E}, \mathbf{E}, \mathrm{Succ}, \alpha_0)$ be a chain with $n-1$ phases fitted by some other tool `FIT` to the density on the interval $[l, \infty)$. We output a d-CTMC $(S \uplus \{s_0\}, \mathcal{E} \uplus \{d\}, \mathbf{E}', \mathrm{Succ}', \alpha'_0)$ with $n$ states that starts with probability $\alpha'_0(s_0) = 1$ in the newly added state $s_0$ in which only the newly added event $d$ is scheduled, i.e. $\mathbf{E}'(s_0) = \{d\}$; the event $d$ has delay $\mathit{delay}(d) = l$ and after it occurs, the chain moves according to the initial distribution of $C$, i.e. $\mathrm{Succ}'(s_0, d) = \alpha_0$; $\mathbf{E}'$ and $\mathrm{Succ}'$ coincide with $\mathbf{E}$ and Succ elsewhere. A pseudo-code for this algorithm `IPH-shift[FIT]` is given in [28].

[5] For the error metrics (Err) we assume that the algorithm outputs a d-CTMC such that $X$ has a density (which holds for our algorithms presented later).

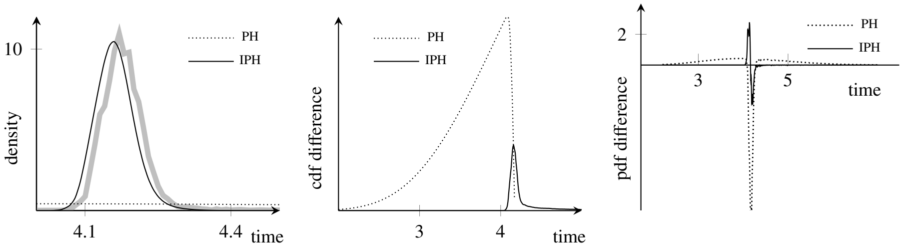


**Fig. 4.** The comparison of the approximations of the event ack using the algorithms PhFit and `IPH-shift[PhFit]` with 30 phases. On the left, there is the density of the original distribution as well as the both approximated densities. In the centre, there is for both approximations the difference of the original and the approximate cumulative distribution function. Notice that in point $x$ the plot displays for each algorithm the error we obtain when measuring the probability that the event occurs within time $x$ (rising as high as 0.5). On the right, there is for both approximations the difference of the original and the approximate density. The integral of this curve is the absolute density difference (Err) that we study.

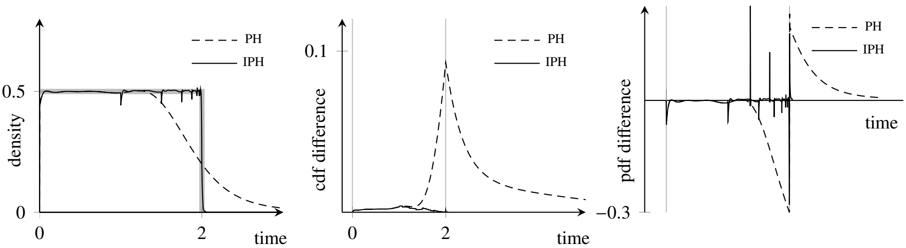


**Fig. 5.** The comparison of the approximations of the distribution uniform on $[0,2]$ using the algorithms PhFit and `IPH-slice[PhFit]`. It goes along the same lines as in Figure 4.

*Example (continued).* To obtain the d-CTMC approximation of the GSMP model of the Alternating bit protocol, we only need to approximate the event ack since timeout is a discrete-time event. To show an example of the technique, the approximation of the event ack using the algorithm `IPH-shift[PhFit]` as well as the whole resulting d-CTMC is depicted in Figure 3. Since `IPH-shift` is using the phase-type approximation only on the "simple" part of the density function, it gets much better results. For instance for 30 phases it yields approx. 4x smaller error compared to the best results of PH algorithms. In Figure 4 we provide a more detailed comparison.

**Delay Bounded from Above.** For an event that cannot occur after some $u < \infty$, our algorithm `IPH-slice[FIT,p]` slices the interval $[0,u]$ using discrete-time events into $p$ subintervals $[0,\frac{1}{2}u]$, $[\frac{1}{2}u,\frac{3}{4}u]$, $[\frac{3}{4}u,\frac{7}{8}u],\ldots,[(1-\frac{1}{2}^{p-2})u,(1-\frac{1}{2}^{p-1})u]$, $[(1-\frac{1}{2}^{p-1})u,u]$. Their length decreases exponentially with the last two subintervals having the same length. Corresponding to these intervals, we build a sequence of components $C_1,\ldots,C_p$ that is traversed by a sequence of discrete-time events $d_1,\ldots,d_{p-1}$ as the time flows. The component of each subinterval $[a,b]$ has $n/p$ phases and is fitted by `FIT` to the *conditional* density of the *remaining* delay given the event has not occurred on $[0,a)$.

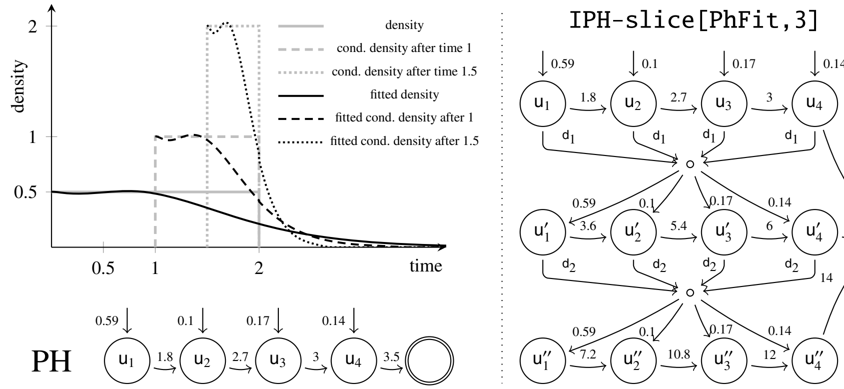


**Fig. 6.** On the left, the uniform distribution on $[0,2]$ is sliced into three subintervals. With the solid line, there is the whole density and its PH approximation corresponding to the CTMC below. With the dashed and dotted line, there are the conditional densities given the event does not occur before 1 and 1.5, and their PH approximations. Their corresponding CTMC are the same as the CTMC below, only with rates $2x$ and $4x$ larger, respectively. This is clarified on the right, in the complete d-CTMC approximation with all 3 components (sharing the absorbing state).

Consider the example from Figure 6. The uniform distribution on $[0,2]$ has density 0.5 in this interval and 0 elsewhere. When already 1.5 time units pass, the conditional density of the remaining delay equals 2 on $[0,0.5]$ and 0 elsewhere.

This algorithm `IPH-slice[FIT,p]` is formally described in [28]. Example output of `IPH-slice[PhFit,3]` on the above mentioned uniform distribution is depicted in Figure 6. Similarly to the previous technique, it provides approximately 8x better results than the standard PH fitting as demonstrated in Figure 5. Note that we can easily combine the two techniques for distributions bounded both from below and above such as uniform on $[5,6]$. It suffices to apply `IPH-shift[IPH-slice[FIT,slices]]`.

Let us provide two remarks on this technique. First, notice that a standard fitting tool is applied on the conditional densities. However, a standard fitting tool tries to minimize the error also *beyond* the subinterval we are dealing with which may lead to suboptimal approximation *on* the subinterval. Modification of a PH algorithm addressing this issue might decrease the error of `IPH-slice` even more. Second, dividing the support of the distribution into subintervals of exponentially decreasing length is a heuristic that works well for distributions where the density does not vary much. For substantial discontinuities in the density, one should consider dividing the support in the points of discontinuity. Next, we briefly review the analysis methods for d-CTMC.

### 3.2 Analysing d-CTMC

The existing theory and algorithms applicable to analysis of d-CTMC are a crucial part of our alternative IPH approximation method. Extending the knowledge in this direction is out of scope of this paper, here we only summarize the state-of-the-art of transient and stationary analysis.

**Table 1.** The (Err) errors and CPU time for different PH tools fitting by 30 phases

| PH fitting tool | (Err) for event ack | (Err) for uniform distribution | CPU time |
|---|---|---|---|
| EMpht | 1.7957 | 1.8980 | over one day |
| G-FIT | 1.6100 | 0.1603 | 4 min 49 s |
| momfit | 1.8980 | 0.5820 | 1 day |
| PhFit | 1.6518 | 0.1868 | 4.33 s |

The method of *supplementary variables* [11,15,31] analyses the continuous state-space $S \times (\mathbb{R}_{\geq 0})^{\mathcal{E}}$ extended by the remaining times *remain*($e$) until each currently active discrete-time event $e$ occurs. The system is described by partial differential equations and solved by discretization in the tool DSPNExpress 2.0 [30]. A more elaborate method of *stochastic state classes* [37,2,22,21] implemented in the tool Oris [9] studies the continuous state-space model at moments when events occur (defining an embedded Markov chain). In each such moment, multidimensional densities over *remain*($e$) are symbolically derived. The embedded chain is finite iff the system is regenerative, approximation is applied otherwise.

If the d-CTMC has at most one discrete-time event active at a time (e.g. when only one event is approximated by IPH), one can apply the efficient method of *subordinated Markov chains* [32]. It builds the embedded Markov chain using transient analysis of CTMC, similarly to the analysis of CTMC observed by a one-clock timed automaton [10]. In the tool Sabre [16], this method is extended to parallel discrete-time events by approximating them using one discrete-time event $\Delta$ [18] that is active in all states and emulates other discrete-time events. An event $e$ occurs with the $\lfloor delay(e)/delay(\Delta)\rfloor$-th occurrence of $\Delta$ after initialization of $e$. Note that this corresponds to discretizing time for the discrete-time events while leaving the exponential events intact.

As some of the methods are recent, no good comparison of these methods exists. Based on our preliminary experiments, we apply in Section 4 the tool Sabre.

## 4 Experimental Evaluation

In this section we evaluate the reduction of the state space and hence the reduction of the time needed for the analysis when using IPH compared to PH. Precisely, (1) we inspect the growth of the state space of both IPH and PH approximations when decreasing the tolerated error; (2) for a fixed tolerated error, we examine the growth of the state space of the PH approximation when increasing the shift of a shifted distribution; and (3) for a fixed model and a fixed PCTL property we compare the running time of the analysis of d-CTMC yielded by IPH and the running time of the analysis of CTMC yielded by PH when increasing the number of phases.

We consider the distributions from the previous sections, namely the shifted distribution of the event ack addressed by the `IPH-shift` algorithm and the distribution uniform on $[0,2]$ addressed by the `IPH-slice` algorithm. The uniform distribution is specified simply by its formula whereas the density of the event ack is based on real data. Using the Unix `ping` command, we collected 10000 successful ICMP response

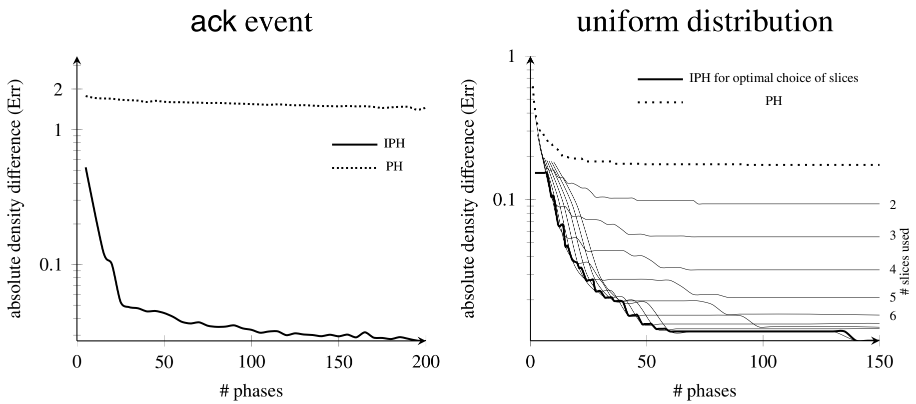


**Fig. 7.** The (logarithmically scaled) relationship between the size of the state space and the error obtained. For the uniform distribution, we show the results for different numbers of slices, each with the same number of phases. The plotted number of phases is the sum of the phases within all used slices. The error for the optimal number of slices is plotted in bold.

times of a web server (www.seznam.cz, the most visited web portal in the Czech Republic). The data set has mean 4.19 ms, standard deviation 0.314 ms, variance 0.0986, coefficient of variation 0.075 ms, and the shortest time is 4.06 ms (see Figure 4).

To get reliable results, we need to compare IPH with state-of-the-art tools for continuous PH fitting. For our experiments, we considered the tools *EMpht* [3], *G-FIT* [39], *momfit* [25], and *PhFit* [24] (an extended comparison including the tool *HyperStar* [36] is in [28]). We ran the tools to produce PH approximations of the two events with 30 phases (we chose such a small number of phases because for some tools it already took a substantial amount of time). Based on the results shown in Table 1, we have selected PhFit as the baseline tool. Most of the tools achieve similar precision, however PhFit significantly outperforms all others regarding the CPU time[6].

## 4.1 Growth of the State Space When Decreasing Error

In the first experiment, we focus on the size of the state space necessary to fit the distributions up to a decreasing error. The decreasing errors (Err) when increasing the number of phases, i.e. the state space, are shown in Figure 7. Both our IPH algorithms exhibit a fast decrease of the error (note that the scales are logarithmic). Observe that the continuous PH method does not perform particularly well on the event ack obtained as a real-world example since the absolute density difference of two densities can never exceed 2. For the uniform distribution, we show the results for different numbers of slices used in the `IPH-slice` algorithm. According to our experiments on the uniform distribution, a finer slicing with less phases in each slice is better than a coarser one with more phases in each slice, whenever each slice is fitted by at least 4 phases.

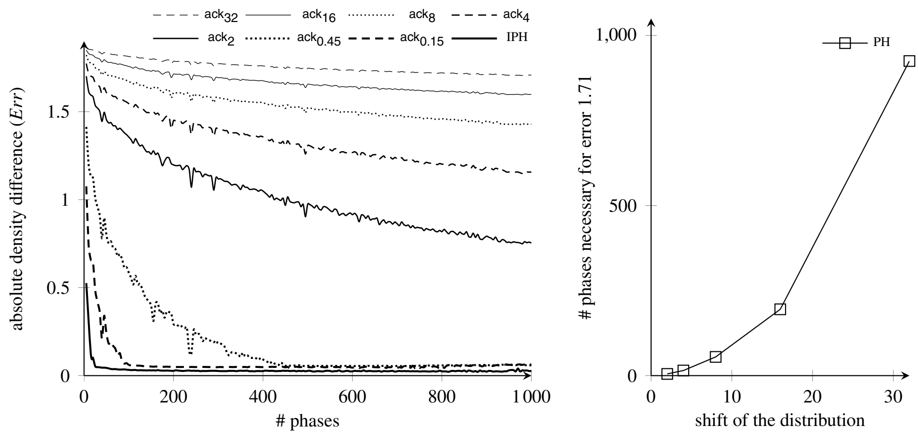


**Fig. 8.** The growth of the state space when increasing the shift. On the left, there is the dependence of the error on the size of the state space for distributions with different shifts. Note that the IPH fitting does not depend on the shift. On the right, there is the growth of the state space when increasing the shift and fixing the PH fitting error to 1.71.

### 4.2 Growth of the State Space When Increasing the Shift

In the second experiment, we analyse the growth of the state space when increasing the shift of a shifted distribution. In other words, *how much larger model we get when we try to fit with a fixed error an event with lower coefficient of variation?* We took the distribution of the ack event and shifted the data to obtain a sequence of events $\mathsf{ack}_{0.15}, \cdots, \mathsf{ack}_4, \cdots \mathsf{ack}_{32}$ where $\mathsf{ack}_i$ has zero density on the interval $[0,i]$. Note that compared to ack we shifted the data in both directions as $\mathsf{ack} \approx \mathsf{ack}_4$. The results in Figure 8 confirm a quadratic relationship between the shift and the necessary number of phases for the PH approximation [16].

The quadratic relationship can be supported by the following explanation. Assume we want to approximate a discrete distribution with shift $s$ by a PH distribution. Due to [1], the best PH distribution for this purpose is the Erlang distribution, the chain of $k$ phases with exit rates $k/s$. Since (Err) does not work in this setting (density is not defined for discrete distributions), we use another common metric - matching moments. Here the goal is to exactly match the mean and minimize the difference of variance. Since the variance of the discrete distribution is zero, the error for $k$ phases is the variance of the Erlang distribution, i.e. $s^2/k$. To get the same error for a discrete distribution with $n$-times increased shift $n \cdot s$, we need $n^2 \cdot k$ phases as $(n \cdot s)^2/(n^2 \cdot k) = s^2/k$.

### 4.3 Time Requirements and Error Convergence When Increasing State Space

So far, we studied how succinct the IPH approximations are compared to PH. One can naturally dispute the impact of IPH approximation by saying that the complexity of d-CTMC analysis is higher that the complexity of CTMC analysis. Here, we show an example where IPH in fact leads to a *lower* overall analysis time.

[6] The analysis has been performed on Red Hat Enterprise Linux 6.5 running on a server with 8 processors Intel Xeon X7560 2.26GHz (each with 8 cores) and shared 448 GiB DDR3 RAM.

**Table 2.** Probability of collision computed by unbounded reachability in CTMC derived using PH and in d-CTMC derived using IPH. The exact probability of collision is 0.7753.

| PRISM on CTMC | | | Sabre on d-CTMC | | |
|---|---|---|---|---|---|
| phases | result | CPU time | phases | result | CPU time |
| 100 | 0.527 | 6.37 s | 5 | 0.695 | 41s |
| 200 | 0.541 | 14.92 s | 10 | 0.730 | 1 min 24 s |
| 500 | 0.562 | 47.74 s | 20 | 0.745 | 3 min 37 s |
| 1000 | 0.585 | 2 min 55 s | 30 | 0.758 | 10 min 30 s |
| 2000 | 0.629 | 9 min 20s | | | |
| 3000 | 0.680 | 50 min 49s | | | |
| 5030 | 0.705 | 3h 14 min | | | |
| 10030 | 0.731 | 32 h 2 min | | | |

We model two workstations competing for a shared channel. Each workstation wants to transmit its data for which it needs 1.2 seconds of an exclusive use of the channel. Each workstation starts the transmission at a random time. If one workstations starts its transmission when the other is transmitting, a collision occurs. Our goal is to compute the probability of collision. For the transmission initiations, we again used the `ping` command (for two different servers) and obtained two distributions with zero density in the first 4.1 seconds and the first 5.51 seconds, respectively.

We approximated the model using both PH and IPH and subsequently run analysis in the tools PRISM [29] and Sabre [16] that are according to our knowledge the best tools for analysing large CTMC and d-CTMC models, respectively [7] we used IPH approximation for both transmission initiating distributions and a discrete-time event for the 1.2 seconds of transmission. The probability of collision was computed by reachability analysis. In the CTMC model for PRISM, we used PH approximations for both transmission initiating distributions. Furthermore, as PRISM does not support nesting of time bounded until operator into until operator, we again needed to transform the problem into (unbounded) reachability analysis by incorporating the 1.2 seconds of transmission time in the model. We approximated the time by Erlang distribution with 1000 phases (using different number of phases causes at most 1% error in the result).

The results of our experiments are shown in Table 2. The exact probability of collision is 0.7753 as computed directly from the data sampled by `ping` using a LibreOffice spreadsheet. Due to some numerical errors in the version of Sabre that we used, we were not able to get a lower error than 2% even when using more than 30 phases. Note that the results we were able to obtain from PRISM have a more then twice as high error[8]. Moreover, the immense analysis times shown in Table 2 do not include the durations of PH approximations. The largest approximation we were able to obtain using PhFit was for 3000 phases as for 4000 phases it did not finish within 5 days. For 5030 and 10030 phases we thus constructed the approximations by concatenating an Erlang

[7] To eliminate the effects of the implementation, we also run the CTMC analysis in Sabre. However, it is much slower than PRISM. Full details are in [28].

[8] We did our best to make CTMC analysis as quick as possible, we used parameters `-s -gs -maxiters 1000000 -cuddmaxmem 18000000` and set `PRISM_JAVAMAXMEM` to `200000m`.

approximation of the shift with the 30 phases PH approximation of the remaining part (as it was obtained during IPH). Overall, the results indicate that for models that are sensitive to precise approximation of the distributions, the IPH approximation can lead to a significantly faster analysis compared to PH approximation.

## 5 Future Work

There are several directions for future work. First, a comparison of the existing algorithms [16,31,21] that can be applied to the transient analysis of d-CTMC would be highly welcome. Furthermore, for the best algorithm for d-CTMC, one can perform a more detailed comparison of its running times on the d-CTMC obtained by the IPH fitting with the analysis times of other available methods (such as the standard PH fitting). Second, we believe that further heuristics can increase the efficiency of IPH or its applicability to a wider class of distributions. Finally, our method justifies the importance of research on further analysis algorithms for d-CTMC.

**Acknowledgement.** We would like to thank Vojtěch Forejt, András Horváth, David Parker, and Enrico Vicario for inspiring discussions.

## References

1. Aldous, D., Shepp, L.: The least variable phase type distribution is Erlang. Communications in Statistics. Stochastic Models 3(3), 467–473 (1987)
2. Alur, R., Bernadsky, M.: Bounded model checking for GSMP models of stochastic real-time systems. In: Hespanha, J.P., Tiwari, A. (eds.) HSCC 2006. LNCS, vol. 3927, pp. 19–33. Springer, Heidelberg (2006)
3. Asmussen, S., Nerman, O., Olsson, M.: Fitting phase-type distributions via the EM algorithm. Scandinavian Journal of Statistics 23(4), 419–441 (1996)
4. Baier, C., Haverkort, B., Hermanns, H., Katoen, J.-P.: Model-checking algorithms for continuous-time Markov chains. IEEE Trans. on Software Engineering 29(6), 524–541 (2003)
5. Bernadsky, M., Alur, R.: Symbolic analysis for GSMP models with one stateful clock. In: Bemporad, A., Bicchi, A., Buttazzo, G. (eds.) HSCC 2007. LNCS, vol. 4416, pp. 90–103. Springer, Heidelberg (2007)
6. Bobbio, A., Horváth, A., Scarpa, M., Telek, M.: Acyclic discrete phase type distributions: Properties and a parameter estimation algorithm. Performance Evaluation 54(1), 1–32 (2003)
7. Bobbio, A., Telek, M.: A Benchmark for PH Estimation Algorithms: Results for Acyclic-PH. Communications in Statistics. Stochastic Models 10(3), 661–677 (1994)
8. Brázdil, T., Krčál, J., Křetínský, J., Řehák, V.: Fixed-delay events in generalized semi-Markov processes revisited. In: Katoen, J.-P., König, B. (eds.) CONCUR 2011. LNCS, vol. 6901, pp. 140–155. Springer, Heidelberg (2011)
9. Bucci, G., Carnevali, L., Ridi, L., Vicario, E.: Oris: A tool for modeling, verification and evaluation of real-time systems. International Journal on Software Tools for Technology Transfer 12(5), 391–403 (2010)
10. Chen, T., Han, T., Katoen, J.-P., Mereacre, A.: Quantitative model checking of continuous-time Markov chains against timed automata specifications. In: LICS, pp. 309–318. IEEE (2009)
11. Cox, D.R.: The analysis of non-Markovian stochastic processes by the inclusion of supplementary variables. Math. Proceedings of the Cambridge Phil. Society 51, 433–441, 7 (1955)

12. Fackrell, M.: Fitting with matrix-exponential distributions. Stoch. Models 21(2-3), 377–400 (2005)
13. Faddy, M.J.: On inferring the number of phases in a Coxian phase-type distribution. Communications in Statistics. Stochastic Models 14(1-2), 407–417 (1998)
14. Feldmann, A., Whitt, W.: Fitting Mixtures of Exponentials to Long-Tail Distributions to Analyze Network. Perform. Eval. 31(3-4), 245–279 (1998)
15. German, R., Lindemann, C.: Analysis of stochastic Petri nets by the method of supplementary variables. Performance Evaluation 20(1-3), 317–335 (1994)
16. Guet, C.C., Gupta, A., Henzinger, T.A., Mateescu, M., Sezgin, A.: Delayed continuous-time Markov chains for genetic regulatory circuits. In: Madhusudan, P., Seshia, S.A. (eds.) CAV 2012. LNCS, vol. 7358, pp. 294–309. Springer, Heidelberg (2012)
17. Haas, P.J.: Stochastic petri nets. Springer (2002)
18. Haddad, S., Mokdad, L., Moreaux, P.: Performance evaluation of non Markovian stochastic discrete event systems – a new approach. In: WODES 2004, p. 243. Elsevier (2005)
19. Hatefi, H., Hermanns, H.: Improving time bounded reachability computations in interactive Markov chains. In: Fundamentals of Soft. Engineering, pp. 250–266. Springer (2013)
20. Horváth, A., Paolieri, M., Ridi, L., Vicario, E.: Probabilistic model checking of non-Markovian models with concurrent generally distributed timers. In: QEST, pp. 131–140. IEEE (2011)
21. Horváth, A., Paolieri, M., Ridi, L., Vicario, E.: Transient analysis of non-markovian models using stochastic state classes. Performance Evaluation 69(7), 315–335 (2012)
22. Horváth, A., Ridi, L., Vicario, E.: Transient analysis of generalised semi-Markov processes using transient stochastic state classes. In: QEST, pp. 231–240. IEEE (2010)
23. Horváth, A., Telek, M.: Markovian modeling of real data traffic: Heuristic phase type and MAP fitting of heavy tailed and fractal like samples. In: Calzarossa, M.C., Tucci, S. (eds.) Performance 2002. LNCS, vol. 2459, pp. 405–434. Springer, Heidelberg (2002)
24. Horváth, A., Telek, M.: Phfit: A general phase-type fitting tool. In: Field, T., Harrison, P.G., Bradley, J., Harder, U. (eds.) TOOLS 2002. LNCS, vol. 2324, pp. 82–91. Springer, Heidelberg (2002)
25. Horváth, A., Telek, M.: Matching more than three moments with acyclic phase type distributions. Stochastic Models 23(2), 167–194 (2007)
26. Jensen, A.: Markoff chains as an aid in the study of Markoff processes. Scandinavian Actuarial Journal 1953(supp. 1), 87–91 (1953)
27. Jones, R., Ciardo, G.: On phased delay stochastic Petri nets: Definition and an application. In: Petri Nets and Performance Models, pp. 165–174. IEEE (2001)
28. Korenčiak, Ľ., Krčál, J., Řehák, V.: Dealing with zero density using piecewise phase-type approximation. CoRR, abs/1406.7527 (2014)
29. Kwiatkowska, M., Norman, G., Parker, D.: PRISM 4.0: Verification of probabilistic real-time systems. In: Gopalakrishnan, G., Qadeer, S. (eds.) CAV 2011. LNCS, vol. 6806, pp. 585–591. Springer, Heidelberg (2011)
30. Lindemann, C., Reuys, A., Thummler, A.: The DSPNexpress 2.000 performance and dependability modeling environment. In: Fault-Tolerant Comp., pp. 228–231. IEEE (1999)
31. Lindemann, C., Thümmler, A.: Transient analysis of deterministic and stochastic Petri nets with concurrent deterministic transitions. Performance Evaluation 36, 35–54 (1999)
32. Marsan, M.A., Chiola, G.: On Petri nets with deterministic and exponentially distributed firing times. In: Rozenberg, G. (ed.) APN 1987. LNCS, vol. 266, pp. 132–145. Springer, Heidelberg (1987)
33. Molloy, M.K.: Discrete time stochastic Petri nets. Software Eng. SE-11(4), 417–423 (1985)
34. Neuts, M.F.: Matrix-geometric solutions in stochastic models: An algorithmic approach. The Johns Hopkins University Press, Baltimore (1981)

35. O'Cinneide, C.A.: Phase type distributions: Open problems and a few properties. Communications in Statistics. Stochastic Models 15(4), 731–757 (1999)
36. Reinecke, P., Krauss, T., Wolter, K.: Hyperstar: Phase-type fitting made easy. In: QEST, pp. 201–202. IEEE (2012)
37. Sassoli, L., Vicario, E.: Close form derivation of state-density functions over DBM domains in the analysis of non-Markovian models. In: QEST, pp. 59–68. IEEE (2007)
38. Silverman, B.W.: Density estimation for statistics and data analysis. Monographs on Statistics and Applied Probability. Chapman and Hall (1986)
39. Thummler, A., Buchholz, P., Telek, M.: A novel approach for phase-type fitting with the EM algorithm. Dependable and Secure Computing 3(3), 245–258 (2006)
40. Zhang, L., Neuhäußer, M.R.: Model checking interactive Markov chains. In: Esparza, J., Majumdar, R. (eds.) TACAS 2010. LNCS, vol. 6015, pp. 53–68. Springer, Heidelberg (2010)