Phase-type Approximations for Message Transmission Times in Web Services Reliable Messaging

Philipp Reinecke and Katinka Wolter

Humboldt-Universität zu Berlin
Unter den Linden 6
10099 Berlin
{preineck|wolter}@informatik.hu-berlin.de

Abstract. Web-Services based Service-Oriented Architectures (SOAs) become ever more important. The Web Services Reliable Messaging (WSRM) standard provides a reliable messaging layer to these systems. In this work we present parameters for acyclic continuous phase-type (ACPH) approximations for message transmission times in a WSRM implementation confronted with several different levels of IP packet loss. These parameters illustrate how large data sets may be represented by just a few parameters. The ACPH approximations presented here can be used for the stochastic modelling of SOA systems. We demonstrate application of the models using an M/PH/1 queue.

1 Introduction

Web-Services-based Service-Oriented Architectures (SOAs) play an increasing role in private and commercial activities on the Internet. These systems require reliable transmissions of SOAP messages over possibly unreliable links.

The Web Services Reliable Messaging (WSRM) standard provides an interface for reliable message transmissions [BIMT05]. From the application's point of view, a WSRM implementation guarantees several properties (i.e. INORDER, EXACTLY-ONCE) for message transmissions.

WSRM ensures message transmission by retransmitting messages for which no acknowledgement arrives within a certain timeout. The timeout after which the message transmission task is restarted strongly influences the timing characteristics of the WSRM as perceived by the application. In previous work we studied the effect of different restart strategies [RvMW06], and the adaptivity of restart strategies in the WSRM area [RW08b].

In this work we provide phase-type approximations for the effective transmission times encountered in a WSRM implementation where messages are transmitted over a link with IP packet loss. We consider three scenarios with different loss levels and derive acyclic continuous phase-type (ACPH) approximations for the transmission time distribution. We present three different approximations for each data set: First, we use a simple ACPH(2) model fitted by matching the first three moments [TH02]. The second model is a Hyper-Erlang distribution (HErD), fitted using the G-FIT tool [TBT06,PT07]. Third, we approximate the trace distributions with general ACPH distributions using the PhFit tool [HT02].

We provide several contributions. First, we show how a large data set can be represented in just a few model parameters and can hence be stored in a very efficient and compact way. Second, we provide a model of a lossy communication network using WSRM that can be inserted into a larger model of e.g. a web services scenario. We present a simple M/PH/1 queue as an example of how to use our fitted models. In that sense, we provide benchmarks to evaluate web services performance and reliability. Third, we compare different models as well as different fitting tools on a large data set of measurements in an intricate scenario. The shape of the empirical distribution is unlike any known probability distribution, therefore a compromise between well matching the heavy tail, that

is still based on extremely few observations and a good fit of the bulk of the data must be found.

The remainder of this paper is structured as follows: The next section briefly introduces the basic formalisms employed throughout the paper. Section 3 presents the experiments. In Section 4 we discuss important properties of the data sets and present the ACPH models. We evaluate our approximations in Section 5. The paper concludes with an application of the models in an M/PH/1 queue.

2 Acyclic Phase-Type Distributions (ACPH)

Phase-type distributions represent the time to absorption in a Markov chain with one absorbing state. Acyclic Phase-type distributions (APH) form an important subclass of PH distributions. In this paper we focus on continuous phase-type distributions.

An ACPH model with N transient states is described by the tuple $(\mathbf{Q}, \alpha)$, where

$$\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_N)$$

is the vector of initial probabilities, and $\mathbf{Q}$ is the transition matrix for the transient states. The underlying CTMC for this model is [Hav98]:

$$\hat{\mathbf{Q}} := \begin{bmatrix} \mathbf{Q} & \mathbf{q} \\ \mathbf{0} & 0 \end{bmatrix},$$

where the column vector $\mathbf{q}$ can be easily derived from $\mathbf{Q}$: $\mathbf{q} = -Q\mathbf{1}$. The probability density function (PDF) and the cumulative density function (CDF) of an ACPH distribution are given by, respectively [HT02],

$$f(x) = \alpha \mathrm{e}^{\mathbf{Q}x}\mathbf{q}$$
$$F(x) = 1 - \alpha \mathrm{e}^{\mathbf{Q}x}\mathbf{1}.$$

The ith non-central moment of an ACPH can be computed as [TH02]

$$E[X^i] = i!\alpha(-\mathbf{Q})^{-i}\mathbf{1}.$$

In this work we present ACPH approximations of WSRM message transmission times. We consider models from the general ACPH class and models from two subclasses, viz. second-order ACPH models (ACPH(2)) and Hyper-Erlang distributions (HErD). The next two paragraphs discuss properties of these special cases.

Second-order ACPH. ACPH(2) models consist of only two transient states, i.e.

$$\alpha = (\alpha, 1-\alpha)$$
$$\mathbf{Q} = \begin{bmatrix} -\lambda_1 & \lambda_1 \\ 0 & -\lambda_2 \end{bmatrix}.$$

ACPH(2) models are attractive due to their low number of states, which allows for efficient models. Furthermore, using moment-matching [TH02], parameters for these models can often be obtained directly from the first three moments.

However, precise matching of the first three moments with ACPH(2) is limited by tight bounds on the second and third moment of the data. For data sets whose moments are outside of these bounds, one must either employ a higher-order ACPH or settle on a model that only approximates the second and/or third moment [TH02].

Hyper-Erlang Distributions. Hyper-Erlang distributions (HErDs) consist of a mixture of M Erlang distributions with parameters $(\lambda_r, k_r), r = 1, \ldots, M$. The

transition matrix for the underlying CTMC has the following general structure ($r = 1, \ldots, M$):

$$\mathbf{Q} = \begin{bmatrix} \ddots & & & & & & & & \\ 0 & -\lambda_{r-1} & 0 & 0 & & & & & \\ & 0 & -\lambda_r & \lambda_r & 0 & & & & \\ & & 0 & -\lambda_r & \lambda_r & 0 & & & \\ & & & \ddots & \ddots & \ddots & \ddots & & \\ & & & & 0 & -\lambda_r & 0 & 0 & \\ & & & & & 0 & -\lambda_{r+1} & \lambda_{r+1} & 0 \\ & & & & & & & & \ddots \end{bmatrix}.$$

Accordingly, the vector of initial probabilities is

$$\alpha = (\alpha_1, 0, \ldots, 0, \alpha_2, 0, \ldots, 0, \alpha_M, 0, \ldots, 0).$$

Hyper-Erlang distributions allow us to approximate the distribution of the data more closely than is possible with moment matching using second-order acyclic PH distributions. Moreover, any ACPH can be approximated by a HErD of sufficiently high order, although this may require a HErD with an infinite number of states. On the other hand, typical HErD models have an order higher than two and thus may become computationally expensive. However, HErD matching can be performed more efficiently than general ACPH matching. See [TBT06,PT07] for more detail on Hyper-Erlang distributions.

3 Experiments

Web Services Reliable Messaging provides reliable SOAP message transmissions to Web Services by resending messages for which no acknowledgement has been received before a timeout elapsed. Every message can thus be transmitted several times. In particular, a retransmission may arrive earlier than the original transmission. Our data sets present the time between the first attempt at sending the message and the first time the message arrived at the destination. This Effective Transmission Time (ETT) determines the effect of the WSRM on the application [RvMW06,RW08b]. The current section describes the experiment setup and measurement preparation procedures.

3.1 Experiment Setup

In our setup a Web Services client transmits one-way messages asynchronously to a server, using WSRM. An enhanced version of Sandesha1/Axis1 [ASFb,ASFa] provides the WSRM implementation for the experiments. The modifications add support for asynchronous message transmissions (implemented similarly to [ZVK04]) and improve Sandesha's stability and performance.[1] The operation environment consists of a 10 Mbit LAN connection, emulated on top of the physical 100 Mbit Ethernet, using the Linux traffic control facilities [Net]. Fault injection occurs in the 10 Mbit connection on a dedicated host. We inject packet loss at the IP layer.

In each experiment run the client transmits 20000 messages with a payload of 256 bytes and a message interarrival time of 100 ms. Experiment runs that take longer than one hour to complete are discarded. In order to avoid the effects of software aging, client and server are restarted between runs.

[1] The current branch of Sandesha development, Sandesha2/Axis2, supports asynchronous invocations natively, but was not stable enough for the experiments. However, the basic operation of different WSRM implementations can be considered comparable.

Packet loss model. IP Packet loss is generated according to a simplified continuous-time Gilbert loss model with one loss-free and one lossy state. Gilbert loss models generate sequences of alternating loss episodes and loss-free periods of exponentially-distributed length, which capture characteristics of packet-loss on Internet links quite well [ZPS00,ZDPS01,VMG06,SCK00,JS00]. We consider three scenarios with different mean loss episode and loss-free period lengths, presented in Tab. 1.

Table 1. Loss model scenarios.

	$\mathcal{S}_1$	$\mathcal{S}_2$	$\mathcal{S}_3$
Loss episode length	0.05 s	1 s	1 s
Loss-free period length	120 s	30 s	8 s

WSRM restart algorithms. We provide results for three different algorithms to compute the restart timeout: The *Fixed Intervals* algorithm uses a constant timeout of 4 s. The well-known Jacobson/Karn algorithm adjusts the timeout based on the mean and variance of round-trip time observations [KP91,KR01]. We set the parameters for the Jacobson/Karn algorithm as follows: $k = 4, \alpha = 1/8, \beta = 1/4$, and initial timeout $\mathrm{RTO}_{\mathrm{initial}} = 4\,\mathrm{s}$. Third, we use the QEST algorithm presented in [vMW04], which observes the distribution of completion times and computes a timeout that minimises the expected completion time. The parameters for QEST are: Number of buckets $H = 1000$, maximum timeout $t_{max} = 60\,\mathrm{s}$ and initial timeout $\mathrm{RTO}_{\mathrm{initial}} = 4\,\mathrm{s}$.

In order to reduce load on the medium, both Jacobson/Karn and QEST perform exponential backoff upon timeout, i.e. they double the restart timeout for the next transmission.

3.2 Measurement Preparation

Measurements were obtained by off-line analysis of message send/receive events recorded during the experiments.

Accuracy. Since measurements were computed from timestamps recorded on two different machines, both system clocks needed to be synchronised. System clocks were synchronised using NTP. Clock synchronisation in the testbed was assessed based on NTP log files. System clocks stayed within +/- 2 ms of each other during the experiments, which we consider sufficiently accurate so as to not necessitate the use of skew removal procedures such as presented in e.g. [Pax97,MST98,KG06].

Artifacts. Many runs exhibited a transient increase of ETTs shortly after the start of the experiment, whose root cause could not be identified. Since this phenomenon affected experiments irrespective of the scenario or the restart strategy used, we consider this an artifact introduced by the testbed itself. We only include message numbers $1000, \ldots, 20000$ in our data sets.

Furthermore, measurements for Fixed Intervals have minimum values of 8–9 ms (median 9–13 ms), whereas the other measurements have minima of 12–13 ms (median 10–18 ms). All experiments whose measurements have lower minima were performed some time after the experiments with higher minima. The decrease indicates a change in network characteristics in the time in between. The nature and cause of this change is unknown. We consider this difference negligible for the approximations presented in the next section.

4 Phase-type Approximations

We approximate the data using different classes of acyclic phase-type distributions. For each scenario/algorithm combination we aggregate observations from four runs into one data set (76004 samples) for the approximation, and keep one run (19001 samples) for cross-evaluation of the models. We employ the R statistics package [R D06] in the statistical evaluation of the data.

4.1 Data Set Characteristics

Table 2 presents some statistical properties of the data sets. $\mathcal{S}_{iFI}$, $\mathcal{S}_{iJK}$ and $\mathcal{S}_{iQ}$ denote data sets for Fixed Intervals, Jacobson/Karn and QEST algorithm, respectively, obtained in the ith scenario. Note that all data sets exhibit a coefficient of variation (CoV) above one, which points at possible heavy-tailed behaviour of the distribution underlying the data.

Table 2. Statistical properties of the data sets.

	$\mathcal{S}_{1FI}$	$\mathcal{S}_{2FI}$	$\mathcal{S}_{3FI}$	$\mathcal{S}_{1JK}$	$\mathcal{S}_{2JK}$	$\mathcal{S}_{3JK}$	$\mathcal{S}_{1Q}$	$\mathcal{S}_{2Q}$	$\mathcal{S}_{3Q}$
Mean	17.24	21.32	22.44	124.71	106.51	104.60	386.29	334.91	313.76
Std. Dev.	59.54	46.57	26.98	542.27	441.05	426.03	995.20	903.30	955.51
Minimum	8	12	12	8	12	12	8	9	7
Median	11	16	18	11	17	17	11	11	11
95% quantile	19	24	27	213	179	449	3011	2404	1986
99% quantile	152	162	160	3014	2883	2674	4051	3031	3140
Maximum	3017	2226	562	6280	6562	9260	9900	12499	21056
CoV	11.91	4.79	1.45	18.91	17.15	16.59	6.64	7.23	9.27

Visual inspection of the empirical complementary cumulative density function (CCDF) for $\mathcal{S}_{1FI}$, $\mathcal{S}_{2FI}$ and $\mathcal{S}_{3FI}$ (Figures 1–3) reveals that the bulk of the samples is small, with large values centered around 3 s and 6 s and few observations between the bulk and the extreme values. The 'steps' in the CCDF become more pronounced with higher loss levels, e.g. in $\mathcal{S}_{3FI}$. The accumulation of extreme values at 3 s and 6 s as well as the gaps in between are due to the way TCP detects packet loss during the three-way handshake on connection setup: The TCP starts with a fixed retransmission timeout (RTO) of 3 s, which is doubled on every expiration [KR01]. This means that any SOAP message transmission for which the TCP connection setup experiences packet loss will be delayed by at least 3 s. Since the Fixed Intervals algorithm restarts the transmission only if the message has not been acknowledged after 4 s, it will not resend messages for which only one packet loss happened during connection setup. These message will have effective transmission times between 3 and 4 s. Messages whose connection setup suffers from more than one packet loss are delayed by more than 4 s and will thus be resent. If the retransmission does not experience packet loss, it will succeed earlier than the first one. In this case, instead of a large ETT of 3 s we observe a small ETT from the bulk of the distribution. On the other hand, some retransmissions will also be delayed by packet loss. Since these messages where restarted after at least 4 s and experience a minimum delay of 3 s, their ETT is above 7 s. Then, the first transmission, which was delayed by only 6 s, will finish earlier, which explains the accumulation of samples around 6 s.

Due to space constraints we omit the empirical CCDFs of the data sets from the experiments with Jacobson/Karn and QEST. These differ from the observations for the Fixed Intervals algorithm in that they gather more samples in the bulk. In particular, in the $\mathcal{S}_1$ data set, Jacobson/Karn has fewer observations around 3 s, while for QEST the tail of the observations breaks down even below 1 s. With higher loss levels, however, both start to show peaks around 3 s and 6 s similar to observations from Fixed Intervals. This can be explained by the way in which these algorithms adjust the restart timeout: Message transmissions that

are not delayed by packet loss finish very fast. Based on obervations of these low completion times, both algorithms compute low timeout values (with Jacobson/Karn being more conservative), typically much lower than the 4 s of the Fixed Intervals algorithm. These lower timeouts allow the algorithms to detect (and restart) delayed transmissions early, which results in low completion times. On the other hand, Jacobson/Karn and QEST perform exponential backoff when the timeout elapses. With higher loss levels, it is likely that several timeouts elapse successively. Then, the timeout grows quickly, eventually allowing even very slow transmissions to finish without restart. When the first transmission completes without restart, the timeout is recomputed from the observations, i.e. then slow transmissions will be restarted again.

4.2 ACPH(2)-Approximation

Table 3 presents the ACPH(2) models obtained by moment-matching [TH02] for the first three moments. Note that the third moment could not be matched exactly by an ACPH(2) in data sets $\mathcal{S}_{2FI}$, $\mathcal{S}_{2JK}$ and $\mathcal{S}_{3FI}$. In these cases we approximate the third moment as suggested in [TH02].

Table 3. Parameters for the ACPH(2) models, obtained by moment matching.

	$\mathcal{S}_{1FI}$	$\mathcal{S}_{2FI}$	$\mathcal{S}_{3FI}$	$\mathcal{S}_{1JK}$	$\mathcal{S}_{2JK}$	$\mathcal{S}_{3JK}$	$\mathcal{S}_{1Q}$	$\mathcal{S}_{2Q}$	$\mathcal{S}_{3Q}$
α_1	2.05e-03	2.46e-03	9.07e-03	1.00e-01	1.10e-01	7.86e-02	2.62e-01	2.29e-01	5.70e-02
λ_1	1.11e-03	1.66e-03	8.13e-03	8.06e-04	1.04e-03	9.13e-04	6.78e-04	7.05e-04	3.53e-04
λ_2	6.49e-02	5.04e-02	4.69e-02	7.21e+01	8.36e+01	5.40e-02	1.91e+01	8.80e-02	6.57e-03

4.3 Hyper-Erlang Approximation

We employ the G-FIT tool [TBT06], which implements an EM-algorithm, for fitting the parameters of a HErD to a data set. In order to improve the quality of the fitting, we initialise the parameters λ_r and α using the logarithmic aggregation method presented in [PT07]. We then fit the parameters using the EM algorithm and the whole (i.e. non-aggregated) data set.

We found that using a Hyper-Erlang distribution with 15 branches and shape parameters $k_r = r$ for the rth branch provided good approximation of the data. Table 4 shows the parameters for the data set $\mathcal{S}_{2FI}$. Parameters for the other data sets can be downloaded from [RW08a].

Table 4. HErD parameters for $\mathcal{S}_{2FI}$. The shape parameter k_r of the rth Erlang branch is $k_r = r$.

r	α_r	λ_r	α_{r+1}	λ_{r+1}	α_{r+2}	λ_{r+2}	α_{r+3}	λ_{r+3}
1	1.11e-09	2.20e-03	2.35e-05	4.77e-03	1.03e-04	3.631e-02	4.05e-03	9.47e-03
5	1.87e-03	9.22e-02	1.26e-03	7.10e-03	8.05e-14	2.71e-03	4.98e-03	3.36e-02
9	6.26e-03	8.430e-03	6.69e-03	6.92e-02	1.46e-02	7.70e-02	7.64e-03	4.47e-01
13	3.44e-02	4.48e-03	3.42e-02	1.14e+00	8.84e-01	1.24e+00		

4.4 ACPH Parameters

To fit general ACPH distributions to the data we use the PhFit tool [HT02]. In order to reduce the time needed for fitting the data, we used logarithmic aggregation [PT07] to reduce the size of the data sets before applying PhFit. We compared approximations using the full data sets to those obtained using

the aggregated representation and found that the aggregation procedure had no detrimental effect on the quality of the models.

We present here approxmiations with 30 phases in the body and no special treatment for the tail, with an upper limit of body fitting at the 0.001 quantile. It should be noted that one of the strengths of PhFit lies in fitting the body and the tail of the data separately, which provides for better approximations. However, with our data sets we were not able to obtain feasible parameters for the tail fitting. For this reason, we expect our ACPH(30) models to not represent the tail behaviour correctly. Table 5 presents the ACPH(30) model for the $\mathcal{S}_{2FI}$ data set. General ACPH(30) models for the other data sets may be downloaded from [RW08a].

Table 5. General ACPH(30) parameters for $\mathcal{S}_{2FI}$.

i	α_i	λ_i	α_{i+1}	λ_{i+1}	α_{i+2}	λ_{i+2}	α_{i+3}	λ_{i+3}
1	4.36e-02	1.38e-04	6.51e-03	1.82e-02	7.36e-04	2.28e-02	1.80e-02	6.43e-02
5	1.80e-04	6.56e-02	8.85e-06	6.94e-02	4.09e-05	7.10e-02	1.54e-03	8.28e-02
9	4.83e-04	9.55e-02	1.76e-05	1.16e-01	5.85e-07	1.44e-01	9.30e-05	1.55e-01
13	2.91e-03	1.93e-01	9.45e-05	1.94e-01	2.83e-06	1.98e-01	1.33e-05	2.91e-01
17	1.34e-05	3.89e-01	1.41e-05	5.60e-01	5.18e-06	5.82e-01	4.90e-02	8.03e-01
21	7.08e-01	8.14e-01	1.66e-01	8.14e-01	2.15e-03	8.15e-01	2.44e-04	8.15e-01
25	6.06e-05	8.15e-01	1.59e-05	8.15e-01	2.11e-06	8.15e-01	9.31e-06	8.15e-01
29	1.86e-05	8.15e-01	4.42e-06	8.15e-01				

5 Evaluation

In the previous section we presented three models for each data set. In order to facilitate appropriate application of the models, we will now evaluate the quality of these approximations.

Visual inspection of the ACPH models for e.g. the Fixed Intervals data sets (Figures 1–3) shows that the models approximate the CCDF of the data quite differently. The ACPH(2) models tend to follow the general shape of the empirical CCDF only roughly. In Figure 1 the CCDF of the ACPH(2) fluctuates around the empirical CCDF, while in $\mathcal{S}_{2FI}$ and $\mathcal{S}_{3FI}$ the ACPH(2) overestimates the portion of samples below 10 ms. In contrast, the more complex HErD and the ACPH(30) models provide good approximations of the CCDF for all three scenarios. Both follow the general shape of the CCDF closely.

Note that the ACPH(30) model underestimates the length of the tail in the model for $\mathcal{S}_{1FI}$, and overestimates it in $\mathcal{S}_{2FI}$ and $\mathcal{S}_{3FI}$. This behaviour may be avoided by appending a special tail to the distribution. However, one cannot easily derive the actual shape of the tail from the data. In fact, one may argue that extreme values are rare events, and that thus the breakdown observed in the data sets is simply an artifact of the limited observation period. On the other hand, restart aims to reduce effective transmission times by replacing slow transmissions with fast ones (cf. (4.1)), which makes an abrupt tail breakdown appear likely.

While the similarity of the shape of the CCDFs offers some measure of the goodness of the approximation, other quality measures may be of more interest in particular applications. In [BT94,HT00], several quality measures were proposed. Our quality measures are summarised in Tab. 5. Note that area distances have been computed up to the maximum of the observations for each data set.

Figure 4 presents an overview of the relative error in the first three moments $(e_1, \ldots, e_3)$, absolute PDF area distance (PDFAD) and absolute CDF area distance (CDFAD). The figure shows values for the measures for the nine models we obtained using each approach, e.g. the first nine bars (dark grey) of every

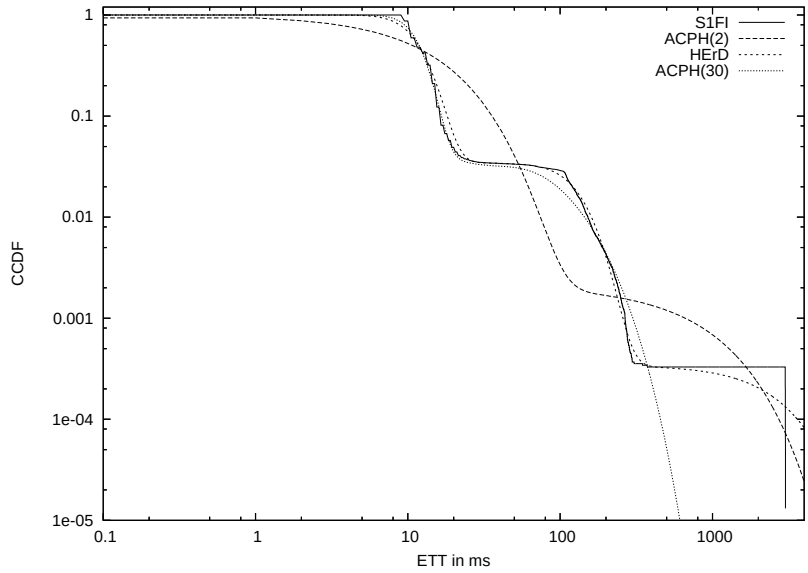

Fig. 1. Empirical CCDF and CCDFs of the approximations for $\mathcal{S}_{1FI}$.

Table 6. Quality measures employed in the evaluation.

Measure	Definition
Rel. Err. in the first moment (c_1 is the mean)	$e_1 = \frac{\lvert c_1(\hat{F}) - c_1(F)\rvert}{c_1(F)}$
Rel. Err. in the second moment (c_2 is the variance)	$e_2 = \frac{\lvert c_2(\hat{F}) - c_2(F)\rvert}{c_2(F)}$
Rel. Err. in the third moment (c_3 is the centered third moment)	$e_3 = \frac{\lvert c_3(\hat{F}) - c_3(F)\rvert}{c_3(F)}$
Absolute PDF area distance	$\text{PDFAD} = \int_0^\infty \lvert\hat{f}(t) - f(t)\rvert dt$
Absolute CDF area distance	$\text{CDFAD} = \int_0^\infty \lvert\hat{F}(t) - F(t)\rvert dt$

measure represent the quality of the ACPH(2) models for $\mathcal{S}_{1FI}$, $\mathcal{S}_{1JK}$, $\mathcal{S}_{1Q}$ and so on.

We note that the ACPH(2) models (dark grey) provide the best approximations of the first three moments. Only in in $\mathcal{S}_{2FI}$, $\mathcal{S}_{2JK}$ and $\mathcal{S}_{3FI}$ do we observe a significant relative error in the third moment. Recall that for these data sets, no precise matching of the third moment was possible, and thus this error is to be expected. The Hyper-Erlang models match the first moment precisely, but exhibit much larger relative errors in the second and third moments. Finally, our ACPH(30) models have large relative errors in the first three moments. According to the area distance measures, all models approximate the data similarly well.

Cross-Evaluation Recall that we used only four runs from each scenario for fitting the models. Table 7 lists statistical properties of the fifth run for each scenario and restart algorithm. We observe subtle differences between the data sets used for the fitting procedure and the evaluation data sets, however, the data sets obviously exhibit the same general characteristics.

Using these data sets, we can assess how well the models capture the typical characteristics of the data. Figure 5 presents the goodness measures for this case. As expected, the goodness of the fit decreases. In particular, the first moments are not matched exactly by any of the models. However, the models still fit the data quite well.

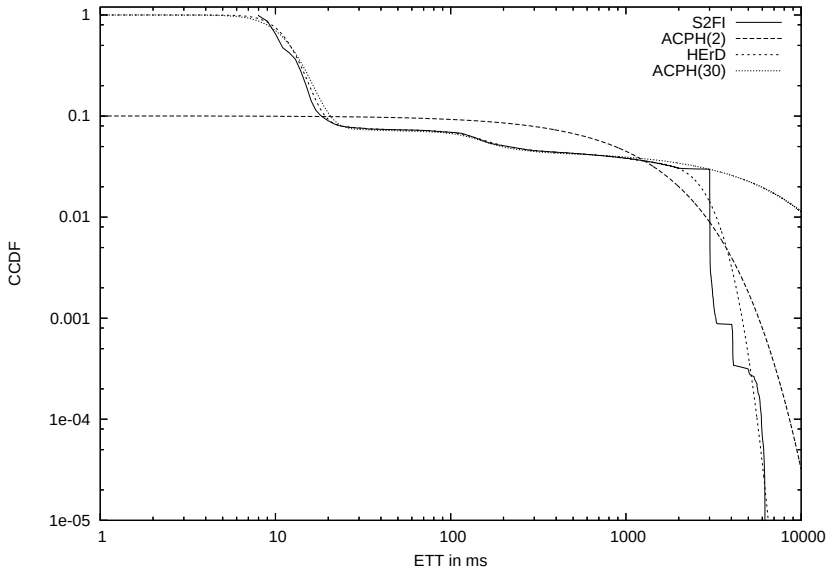

Fig. 2. Empirical CCDF and CCDFs of the approximations for $\mathcal{S}_{2FI}$.

Table 7. Statistical properties of the data sets used for cross-evaluation.

	$\mathcal{S}_{1FI}$	$\mathcal{S}_{2FI}$	$\mathcal{S}_{3FI}$	$\mathcal{S}_{1JK}$	$\mathcal{S}_{2JK}$	$\mathcal{S}_{3JK}$	$\mathcal{S}_{1Q}$	$\mathcal{S}_{2Q}$	$\mathcal{S}_{3Q}$
Mean	19.38	20.32	19.22	120.68	117.10	105.37	403.33	275.83	260.86
Std. Dev.	63.41	49.96	22.86	538.25	466.86	399.14	1021.36	692.76	654.60
Minimum	9	12	12	9	12	12	8	9	9
Median	13	16	15	10	16	15	10	13	13
95% quantile	25	23	23	161	267	495	3009	1879	1803
99% quantile	149	160	162	3013	2551	2516	4059	3017	3013
Maximum	3017	2106	328	4465	3863	3366	9018	5467	6011
CoV	10.71	6.04	1.41	19.89	15.90	14.35	6.41	6.31	6.30

6 Application

To illustrate the potential use of the fitted models we set up a simple M/PH/1 queueing model [Hav98] using the ACPH models for the $\mathcal{S}_{2FI}$ data set. The phase-type distributed service process could represent transmission of SOAP messages over the WSRM, while the stream of SOAP messages is modeled by a Markovian arrival process. We are interested in the mean queue length versus the utilisation of the system, from which the reader may easily compute other standard measures in queueing systems such as response time, waiting time, etc.

The mean service time, that is the mean of the models fitted to our data, equals $E[S] = 124.71\,\text{ms}$ for both the ACPH(2) and the HErD model, and $E[S] = 341.85\,\text{ms}$ for the ACPH(30) model. Note that the ACPH(30) model overestimates the mean service time.

We vary the arrival rate λ to obtain different values of the utilisation ρ of the queue.

The M/PH/1 queueing system has the following matrix-geometric solution [Hav98]

$$E[N] = \mathbf{z}_1(\mathbf{I} - \mathbf{R})^{-2}\mathbf{1}, \tag{1}$$

where $\mathbf{I}$ is the identity matrix. For the M/PH/1 queue the matrix $\mathbf{R}$ evaluates to

$$\mathbf{R} = \lambda(\lambda\mathbf{I} - \lambda\tilde{\mathbf{B}} - \mathbf{Q})^{-1},$$

where the matrix $\tilde{\mathbf{B}}$ is the cross-product of the unit vector $\mathbf{1}$ and the vector of initial probabilities α, i.e. $\tilde{\mathbf{B}} = \mathbf{1}.\alpha$. The steady-state boundary probability vector can then be computed as (cf. [Hav98] Eqn. (8.37))

$$\mathbf{z}_1 = (1 - \rho)\alpha\mathbf{R}.$$

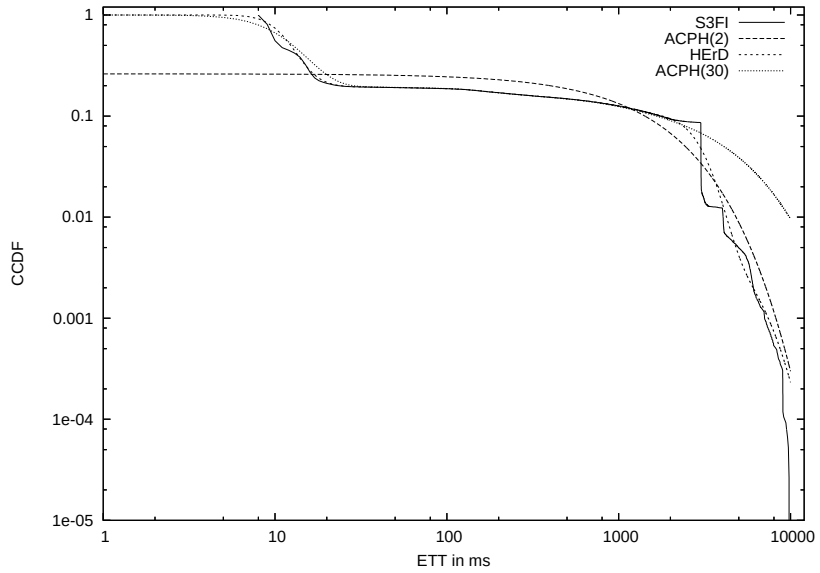

Fig. 3. Empirical CCDF and CCDFs of the approximations for $\mathcal{S}_{3FI}$.

Similar in structure to a DTMC the steady-state probability vectors can be computed as $\mathbf{z}_i = \mathbf{z}_1.\mathbf{R}^{i-1}, i = 1, 2, \ldots$.

Figure 6 shows the mean queue length of an M/PH/1 on a logscale where the three curves differ in the service process represented by the three models we fit to our data, the ACPH(2), the HErD and the ACPH(30) model. Interestingly, both the ACPH(2) and the HErD service distributions not only have the same mean value but also a fairly similar development of the mean queue length. When looking at those two curves one may decide that there is not much gain in the huge HErD model, as compared to the conveniently small two-state ACPH(2) model. We will see that this conclusion is in most cases justified.

The caudal curve, shown in Figure 7, represents the tail behaviour of a matrix-geometric queue [HvMD92]. The caudal curve is constructed using the largest real eigenvalue of the matrix $\mathbf{R}$ versus the utilisation ρ of the queue. In [HvMD92] Eqn. (17) defines the blocking probability in a matrix-geometric queue, showing that if the caudal curve is above the bisector the queue length distribution has a heavy tail, while if the caudal curve is below the bisector there is little probability mass in the tail.

As were the expected queue lengths, also the caudal curves using the ACPH(2) and the HErd service time distribution are very similar. More precisely, for low load the ACPH(2) service time distribution has the heavier tail, while for high load the curves cross and the HErD model leads to the heavier tail.

Since the ACPH(30) model overestimates the tail of the data set, the tail of the queue length distribution is overestimated as well. Furthermore, the large mean value of the ACPH(30) model can be traced back to the overestimated tail.

One may summarise, that for low load the ACPH(2) model gives a conveniently small model that leads to reasonably good results. For high load one might rather resort to the large HErd model, while the ACPH(30) model should be applied with care.

7 Conclusion and Future Work

In this work we presented phase-type models for the distributions of the effective message transmission times in a WSRM implementation under various levels of packet loss. We evaluated the goodness of fit of these models and demonstrated the use of ACPH models in an M/PH/1 queuing model. We conclude that the convenient ACPH(2) class may be sufficient to model the observed transmission

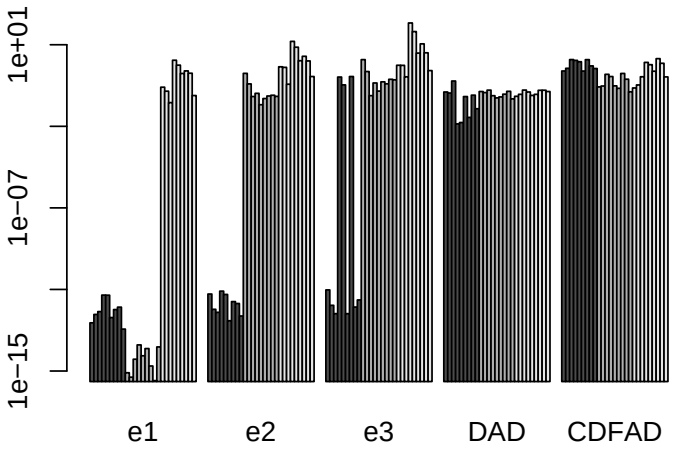

Fig. 4. Goodness measures for the ACPH(2) (dark grey), HErD (medium grey) and ACPH(30) (light grey) models.

times, while there is little gain with large HErD models. Furthermore, general ACPH(30) models perform rather poorly without special treatment for the tail of the distributions.

In this paper we limited ourselves to ACPH distributions. According to [BHT04], DPH models may be preferable for fitting distributions with abrupt changes in the CDF. Since our data exhibits such changes, future work will include trying to fit DPH distributions with appropriate scale factors to these data sets.

References

[ASFa] The Apache Software Foundation: Apache Axis. http://ws.apache.org/axis/.

[ASFb] The Apache Software Foundation: Apache Sandesha. http://ws.apache.org/sandesha/.

[BHT04] A. Bobbio, A. Horváth, and M. Telek. The scale factor: a new degree of freedom in phase-type approximation. *Perform. Eval.*, 56(1-4):121–144, 2004.

[BIMT05] BEA Systems, IBM, Microsoft Corporation Inc, and TIBCO Software Inc. Web Services Reliable Messaging Protocol (WS-ReliableMessaging), February 2005.

[BT94] A. Bobbio and M. Telek. A benchmark for ph estimation algorithm: Results for acyclic-ph, 1994.

[Hav98] B. R. Haverkort. *Performance of Computer Communication Systems: A Model-Based Approach*. John Wiley & Sons, Chichester, UK, 1998.

[HT00] A. Horváth and M. Telek. Approximating heavy tailed behaviour with phase type distributions. In *3d. International Conference on Matrix-Analytic Methods in Stochastic Models, MAM3*, pages 191–214, Leuven, Belgium, 2000. Notable Publications Inc.

[HT02] A. Horváth and M. Telek. Phfit: A general phase-type fitting tool. In *TOOLS '02: Proceedings of the 12th International Conference on Computer Performance Evaluation, Modelling Techniques and Tools*, pages 82–91, London, UK, 2002. Springer-Verlag.

[HvMD92] B. R. Haverkort, A. P. van Moorsel, and A. Dijkstra. Mgmtool: A performance modelling tool based on matrix geometric techniques. In R. Pooley

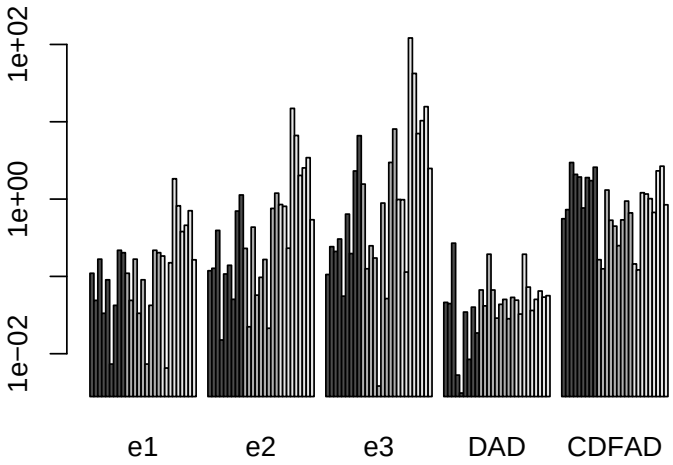

Fig. 5. Cross-evaluation of the models.

and J. Hillston, editors, *Computer Performance Evaluation '92, Modelling Techniques and Tools*, pages 397 – 401. Antony Rowe Ltd., 1992.

[JS00] W. Jiang and H. Schulzrinne. Modeling of Packet Loss and Delay and Their Effect on Real-Time Multimedia Service Quality. In *Proc. NOSSDAV*, 2000.

[KG06] H. Khlifi and J.-C. Grégoire. Low-complexity offline and online clock skew estimation and removal. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 50(11):1872–1884, 2006.

[KP91] P. Karn and C. Partridge. Improving Round-Trip Time Estimates in Reliable Transport Protocols. *ACM Transactions on Computer Systems*, 9(4):364–373, November 1991.

[KR01] B. Krishnamurthy and J. Rexford. *Web Protocols and Practice*. Addison Wesley, 2001.

[MST98] S. B. Moon, P. Skelly, and D. Towsley. Estimation and Removal of Clock Skew from Network Delay Measurements. Technical report, University of Massachusetts, Amherst, MA, USA, 1998.

[Net] Various authors: Netem – linuxnet. http://linux-net.osdl.org/index.php/Netem. Last visited October 8th, 2007.

[Pax97] V. Paxson. End-to-End Internet Packet Dynamics. In *Proceedings of the ACM SIGCOMM '97 conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, volume 27,4 of *Computer Communication Review*, pages 139–154, Cannes, France, September 1997. ACM Press.

[PT07] A. Panchenko and A. Thümmler. Efficient phase-type fitting with aggregated traffic traces. *Perform. Eval.*, 64(7-8):629–645, 2007.

[R D06] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2006. ISBN 3-900051-07-0.

[RvMW06] P. Reinecke, A. P. A. van Moorsel, and K. Wolter. The Fast and the Fair: A Fault-Injection-Driven Comparison of Restart Oracles for Reliable Web Services. In *QEST '06: Proceedings of the 3rd International Conference on the Quantitative Evaluation of Systems*, pages 375–384, Washington, DC, USA, 2006. IEEE Computer Society.

[RW08a] P. Reinecke and K. Wolter. ACPH models for WSRM, `http://www.informatik.hu-berlin.de/~preineck/acphmodels/`, 2008.

[RW08b] P. Reinecke and K. Wolter. Adaptivity Metric and Performance for Restart Strategies in Web Services Reliable Messaging. *WOSP 2008*, 2008. (Accepted for publication).

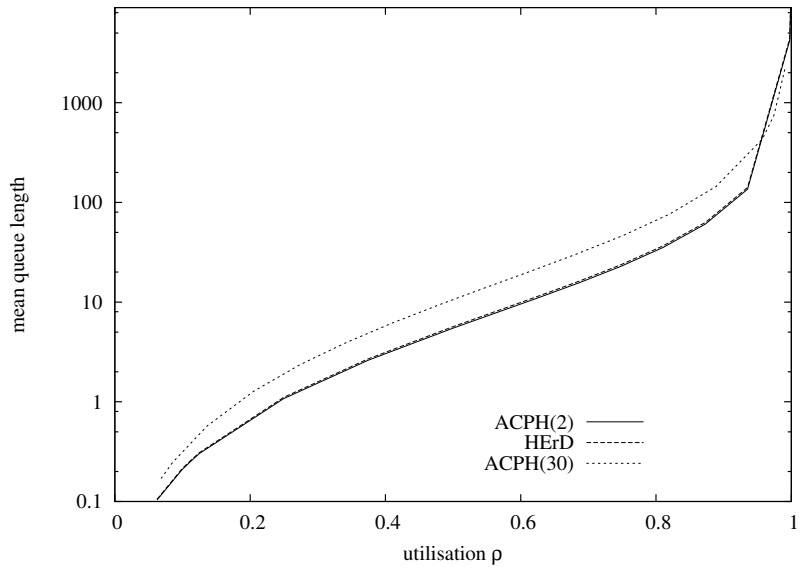

Fig. 6. Mean queue length for all three models

[SCK00] H. Sanneck, G. Carle, and R. Koodli. A framework model for packet loss metrics based on loss runlengths. In *Proceedings of the SPIE/ACM SIGMM Multimedia Computing and Netoworking Conference 2000*, San Jose, CA, January 2000. SPIE/ACM SIGMM.

[TBT06] A. Thümmler, P. Buchholz, and M. Telek. A novel approach for phase-type fitting with the em algorithm. *IEEE Trans. Dependable Secur. Comput.*, 3(3):245–258, 2006.

[TH02] M. Telek and A. Heindl. Matching moments for acyclic discrete and continous phase-type distributions of second order. *International Journal of Simulation Systems, Science & Technology*, 3(3–4):47–57, December 2002.

[VMG06] M. Varela, I. Marsh, and B. Grönvall. A systematic study of PESQ's behavior (from a networking perspective). In *Proceedings of the 5th International Conference MESAQIN 2006: Measurement of Audio and Video Quality in Networks*, 2006.

[vMW04] A. van Moorsel and K. Wolter. Analysis and Algorithms for Restart. In *Proc. 1st International Conference on the Quantitative Evaluation of Systems (QEST)*, pages 195–204, Twente, The Netherlands, September 2004.

[ZDPS01] Y. Zhang, N. Du, V. Paxson, and S. Shenker. On the Constancy of Internet Path Properties. In *Proceedings of the ACM SIGCOMM Internet Measurement Workshop*, 2001.

[ZPS00] Y. Zhang, V. Paxson, and S. Shenker. The Stationarity of Internet Path Properties: Routing, Loss, and Throughput. *ACIRI Technical Report*, 2000.

[ZVK04] U. Zdun, M. Völter, and M. Kircher. Pattern-Based Design of an Asynchronous Invocation Framework for Web Services. *Int. J. Web Service Res.*, 1(3):42–62, 2004.

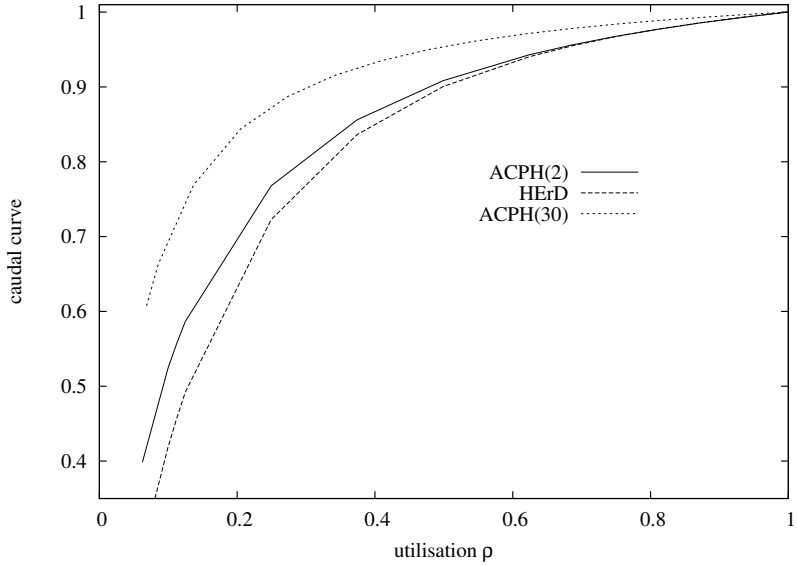

Fig. 7. Caudal curves for all three models