Hybrid analysis of large scale PEPA models

Anton Stefanek Richard A. Hayden Jeremy T. Bradley
Department of Computing, Imperial College London
as1005,rh,jb@doc.ic.ac.uk

We introduce a new hybrid analysis technique for large scale PEPA models that combines stochastic simulation with ODEs, providing a finer trade-off between accuracy of the solutions and the associated computational costs. We describe a version that employs ODEs for the mean component counts and also a version using ODEs for second order moments. This enables the calculation of covariances of component counts and also improves the accuracy of the mean approximations . We look at various examples demonstrating the advantages and limitations of this approach.

1 Introduction

Quantitative analyses of stochastic systems via ordinary differential equations (ODEs) or fluid techniques provide fast approximations of transient measures for models with state spaces beyond the limits of explicit analysis of the underlying stochastic process. The accuracy of the approximations depends on the scale of the component populations and for some approximations the relative error converges to zero as the scale goes to infinity [1].

However, in real applications the populations are finite and the approximations may not be accurate enough to faithfully capture the dynamics of the system. In such cases, the expensive stochastic simulation remains the only alternative. The computational cost is even more apparent if the analysis has to be repeated numerous times, such as when optimising system parameters.

In this paper we try to address this problem by combining the ODE based analysis with stochastic simulation in a way that gives better control over the accuracy/computational cost trade-off. We will work in the context of the Markovian stochastic process algebra PEPA, where the recently developed techniques [2, 3] show how to derive ODE approximations of *moments* (means, variances, etc.) of counts of the individual components in the system. We use the observations from [4] about the error of the ODEs that identify the time intervals where the ODEs are not accurate. During these time intervals, stochastic simulation replaces the ODEs, giving rise to a *hybrid analysis*. We note that this hybrid analysis is different from the hybrid modelling techniques such [5] where a combination of stochastic and deterministic components is used to represent the systems.

We implement the hybrid analyses in an extension to the *GPA* tool [4, 6] that provides the ODE approximations of moments from [3]. We then look at various examples that demonstrate the advantages of the techniques and also highlight several limitations.

1.1 Grouped PEPA

Grouped PEPA is a version of the PEPA stochastic process algebra developed in [3] to conveniently express the ODE approximations to moments of component counts. Consider a simple client/server system where the clients can request some data from the servers. The servers provide the data and the

Submitted to:
PASTA 2010

clients perform some independent action with the data. Additionally, the servers can break and need to be repaired. The components of such system can be described by the GPEPA component definitions:

$$\mathit{Client} \stackrel{def}{=} (\mathit{request}, r_{\mathit{request}}).\mathit{Client\_waiting} \qquad \mathit{Server} \stackrel{def}{=} (\mathit{request}, r_{\mathit{request}}).\mathit{Server\_get} + (\mathit{break}, r_{\mathit{break}}).\mathit{Server\_broken}$$

$$\mathit{Client\_waiting} \stackrel{def}{=} (\mathit{data}, r_{\mathit{data}}).\mathit{Client\_think} \qquad \mathit{Server\_get} \stackrel{def}{=} (\mathit{data}, r_{\mathit{data}}).\mathit{Server}$$

$$\mathit{Client\_think} \stackrel{def}{=} (\mathit{think}, r_{\mathit{think}}).\mathit{Client} \qquad \mathit{Server\_broken} \stackrel{def}{=} (\mathit{reset}, r_{\mathit{reset}}).\mathit{Server}$$

Each client has three states: *Client*, *Client_waiting*, *Client_think*, where it is requesting data, waiting for the data and performing the independent action respectively. Similarly, each server has three states: *Server*, *Server_get*, *Server_broken*, where it is listening for requests, serving the data and waiting to be repaired.

The following *system equation* composes the components to represent the whole system:

$$\mathbf{Clients}\{\mathit{Client}[c]\} \underset{\{\mathit{request},\mathit{data}\}}{\bowtie} \mathbf{Servers}\{\mathit{Server}[s]\}$$

The names inside the $\underset{\mathit{request},\mathit{data}}{\bowtie}$ operator specify that the *request* and *data* actions are shared and the client and server components can only perform them simultaneously. Duration of all the actions in the system are exponential and the model gives rise to a continuous time Markov chain (CTMC). The *group names* **Clients** and **Servers** fix the level at which the ODE approximation is performed by causing the c copies of the *Client* component ($\mathit{Client}[c]$) and s copies of the *Server* component ($\mathit{Server}[s]$) to be undistinguishable. As a result, each state of the system (and of the CTMC) can be uniquely represented by counting the number of instances of the *Client*, *Client_waiting*, *Client_think* state in the group **Clients** and number of instances of *Server*, *Server_get*, *Server_broken* in the group **Servers**. We will use shorthands $C(t)$, $C_w(t)$, $C_t(t)$, $S(t)$, $S_g(t)$, $S_b(t)$ for these counts at time t respectively.

1.2 ODE approximations

In [3] (and originally for PEPA in [2]) it has been shown how to derive ordinary differential equations (ODEs) that approximate the *mean* component counts at each time t, such as the mean number of *Client* states in the group **Clients**, $\mathbb{E}[C(t)]$. The structure of the GPEPA model is used to derive ODEs with solutions $v_P(t)$. These can be shown to approximate the mean count of P components at time t, in the sense that the two quantities are equal when the scale of the model (the initial population size, e.g. the number of client and server components c and s respectively) tends to infinity. Therefore we will use the notation $\widetilde{\mathbb{E}}[P(t)]$ for $v_P(t)$.

Taking ODEs defining $\frac{\mathrm{d}}{\mathrm{d}t}\widetilde{\mathbb{E}}[P(t)]$ for all the components P in the model, we get a closed system of ODEs with initial values given by the component counts in the system equation (counts c and s in the client/server model). Numerically solving this system of ODEs is much cheaper than running the sufficient number of replicated stochastic simulations that would give accurate estimates of the means.

Additionally, [3] shows how to derive ODEs approximating *higher order moments* of the component counts, giving access to further useful measures such as the variance of the component counts, for example $\widetilde{\mathrm{Var}}[C(t)]$ approximating $\mathrm{Var}[C(t)]$.

1.3 Error of the ODEs

The increased efficiency of the numerically solved ODEs is paid for by the decreased accuracy of the method. This results from an approximation on the right hand sides of the ODEs that has to be used in

order to obtain a closed system. This approximation is of the form

$$\mathbb{E}[\min(C(t), S(t))] \approx \min(\mathbb{E}[C(t)], \mathbb{E}[S(t)]). \tag{1}$$

Although it may seem rather radical, the resulting ODEs for means tend to be quite accurate, especially for larger populations. The accuracy decreases for higher moments, where the error for variance can be quite severe even at low scales.

In [4], the accuracy of the ODEs approximating moments of component counts is investigated. It is observed that the error is significant only within certain time intervals.

Intuitively, the approximation (1) is accurate when the processes in the two arguments, i.e. $C(t)$ and $S(t)$, of the min function are far apart. In that case the behaviour of the "mixed process", $\min(C(t), S(t))$ is mostly given by the behaviour of one of the processes $C(t)$ or $S(t)$. However, when the two processes are close (relative to their individual variances), the behaviour of the mixed process is given as a combination of both of the processes $C(t)$ and $S(t)$, as the mixed process is very likely to "switch" between the two modes of behaviour. This reflects changes in cooperation in the modelled system, when the total performance is given by the number of free servers, or by the number of clients if there are sufficiently many servers. The mean of the mixed process is not captured by the approximation (1) that only chooses the mean of one of the processes.

Investigations in [4] propose that a useful indicator of this accuracy is to look at the distance between the means of the two arguments in (1), where the error is the highest roughly when the two are equal. Such points of time are named *switch points* and can be seen from *switch point distance plots*, such as on Figure 1.

The work in [4] doesn't provide any exact quantitative relationship, but observes that the error of the approximations is proportional to the switch point distance. Therefore the times where the error is high concentrate in intervals around the switch points. These are thus good candidates for time intervals where the ODE analysis gets replaced by more accurate stochastic simulation in the below described *hybrid analyses*.

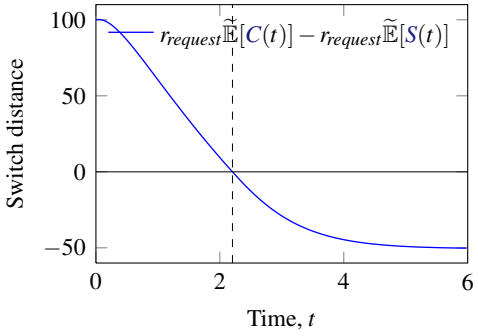

Figure 1: Switch point distance plot. In this example, the system goes through a switch point at $t = 2.1$.

2 Hybrid analyses

In this section we describe two versions of a hybrid analysis where for certain time intervals the numerical solutions to the moment ODEs get replaced by sample moments from a suitably chosen stochastic simulation, providing hybrid approximations $\widehat{\mathbb{E}}[\cdot]$ to the moments. Figure 2 shows an overview.

Each such hybrid analysis must address several issues. It has to choose which ODEs are included in the system. The analysis also must facilitate the change from ODEs to the simulations at each desired time t_s and the change from simulation to ODEs at each time t_o.

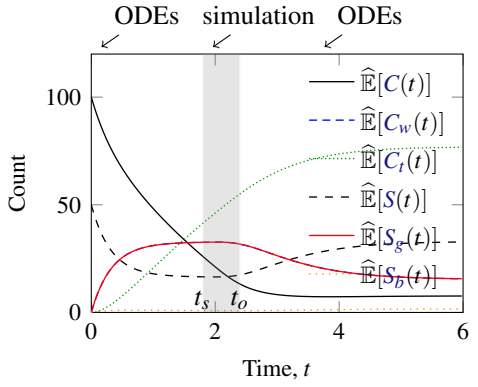

Figure 2: Overview of the hybrid analyses.

The efficiency of the hybrid analysis depends on the length of the simulation interval. Usually, the cost of numerically solving the ODEs is negligible compared to running sufficiently many simulations. The length of the simulation intervals depends on the switch point behaviour. For many of the examples below this interval is only one tenth of the total time considered, thus giving a ten fold improvement over the stochastic simulation.

2.1 First order hybrid analysis

The *first order hybrid analysis* is perhaps the most straightforward instance of the hybrid analysis outlined above. It combines the ODEs for mean component counts with stochastic simulation to produce more accurate approximations of the mean component counts $\widehat{\mathbb{E}}^1[\cdot]$.

At each time t_s where simulation replaces the ODEs, each replication of the simulation is restarted with component counts set to the values given by the solution to the ODEs at time t_s. These can be real valued and so the simulated stochastic process is extended accordingly.

At each time t_o where the analysis returns back to the ODEs after simulation, the initial values of the mean ODEs are set to the means of the simulation at time t_o. The left hand side plot on Figure 3 shows an example of the first order hybrid analysis applied to the client/server model with the switching behaviour on Figure 1.

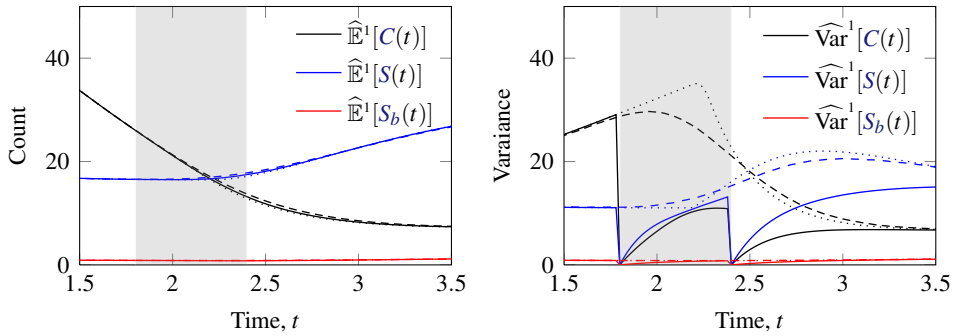

Figure 3: The first order hybrid analysis used to approximate the mean and the variance of component counts. The simulation replaces ODEs in the interval $[1.8, 2.4]$. For comparison, the dashed lines are the values from simulation and the dotted lines from the ODEs. Due to restarting of the simulation, the variance resets to 0.

It can be seen that the hybrid analysis improves the accuracy of the ODE approximations to the means. However, since the simulations always get started at times t_s with the same initial counts, the variances at times t_s reset to 0. The right hand side plot on Figure 3 demonstrates this problem, which the hybrid analysis bellow tries to correct.

2.2 Second order hybrid analysis

The *second order hybrid analysis* uses ODEs for means and covariances of component counts to restart the simulation with the correct covariances to produce more accurate estimates of the mean component counts and covariances $\widehat{\mathbb{E}}^2[\cdot]$.

At each time t_s where simulation replaces the ODEs, each replication of the simulation is started with component counts drawn from a multivariate normal distribution with mean and covariance matrix given by the respective values from the solution to the ODEs at time t_s. There are several technical issues that have to be considered.

First, in order to simulate a multivariate normal distribution, the covariance matrix has to be positive definite. However, the approximations from the ODEs at times t_s may not necessarily be positive definite. Therefore, we have implemented the method from [7] which transforms the covariance matrices to have positive real eigenvalues (a sufficient and necessary condition for the matrix to be positive definite).

Second, the support of the multivariate normal distribution is the whole of $\mathbb{R}$, whereas the component counts can only be positive or zero. To tackle this, we have implemented a simple greedy method that transforms a set of multivariate normal samples so that they are all positive while trying to maintain the same mean and variances. We will demonstrate that this introduces errors for components with small means.

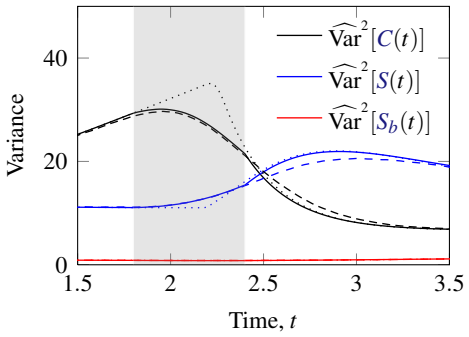

Figure 4: The second order hybrid analysis improves the accuracy of the variance approximations by restarting the simulations with correct covariances.

The Figure 4 shows an example of the variance approximation for the client/server model.

In the ODEs for moments of component counts from [3], only the ODEs for the second order moments depend on the means and not vice-versa. Therefore, the approximations of the means do not take the covariances into account. One way of looking at the second order hybrid analysis is that it feeds back the second order moments into the mean approximations through the re-sampling at the times t_s. This should result in an improved accuracy of means from the second order hybrid analysis over the approximations from the first order analysis. The next section demonstrates this.

3 Examples

3.1 Effects of interval lengths

Intuitively, increasing the length of the time interval where simulations replace the ODEs should increase the accuracy of the hybrid approximations to mean component counts. In Figure 5 we vary the length of the time interval around the switch point in the client/server model and look at the error of the first and second order hybrid analyses.

It can be seen that for components with large populations such as the *Client* component, both the first and second order hybrid analyses improve the accuracy. In the first order analysis, the simulation causes smooth decrease in the error, as both the mean ODEs and the restarted simulations ignore the covariances of component counts at time t_s. In case of the second order hybrid analysis, the decrease in error is radical, since the mean estimate from the restarted simulation considers the covariances at time t_s calculated by the ODEs.

However, the situation is different for components with low populations, such as the *Server_broken* component. In case of those, the restarted simulations of the real valued process do not faithfully capture the behaviour of the original integer valued process. This is due to the fact that neither the restart from a constant real value nor the restart from the manipulated normal samples correctly approximate the distribution of the process. In some cases, the error is even worse than the error from the ODEs.

3.2 Scaling

It can be shown that the relative error of the ODEs approximating mean component counts converges to zero as the scale of the system increases. More precisely, if instead of starting with c client and s server

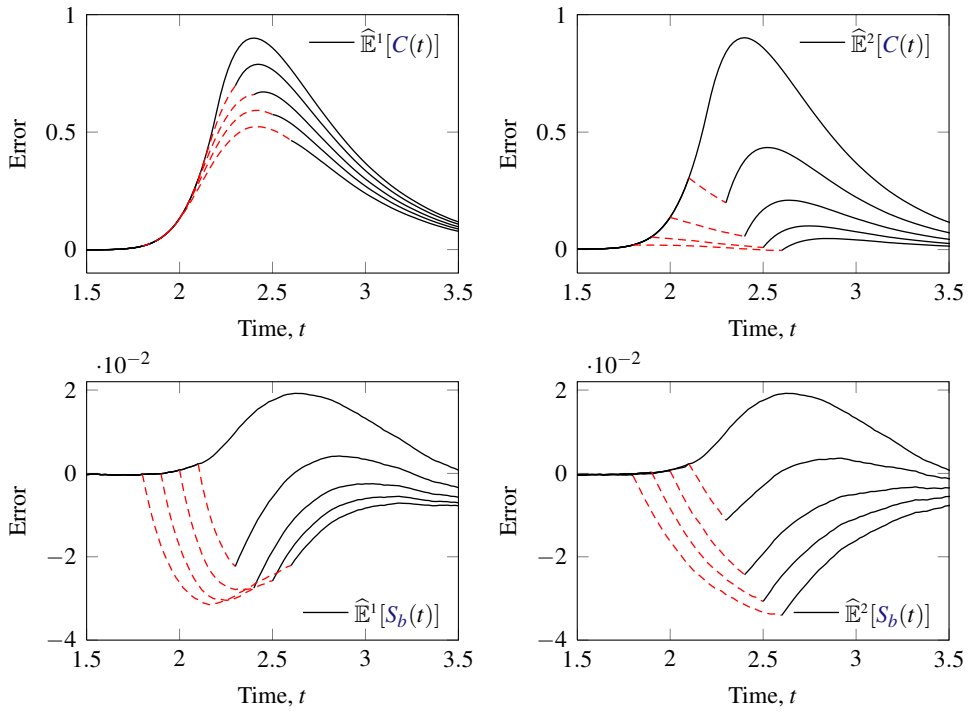

Figure 5: Effect of the size of the interval where ODEs are replaced by simulation on the error of mean component counts. The dashed segments of the plots show the intervals used in the hybrid analyses. Intervals $[2.1,2.3],[2.0,2.4],[1.9,2.5],[1.8,2.6]$ were used. The solid line shows the error of the pure ODE analysis.

components, the system starts with $c \cdot n$ and $s \cdot n$ components respectively, the relative error $\frac{1}{n}(\mathbb{E}[\cdot] - \widetilde{\mathbb{E}}[\cdot])$ converges to 0 as n goes to infinity. We examine the effects of increasing the scale n on the error of the hybrid analyses. Figure 6 takes $n = 10$ and produces the same plots as in Figure 5, also showing the actual error (not the relatively scaled) $\widehat{\mathbb{E}}[\cdot] - \mathbb{E}[\cdot]$.

Here, the error for the *Client* component with large populations decreases as the simulation interval size increases. The decrease is similar to the case $n = 1$ for the first order hybrid analyses. However, for the second order analysis, the improvement is more significant. This is because at larger scales, due to the Central Limit Theorem, the component counts become normally distributed and so restarting the simulation from multivariate normal distribution becomes more accurate.

In case of the *Server_broken* component, the small populations are no longer near zero and so both the first and second order hybrid analyses improve the accuracy of the mean approximations.

3.3 Variance

For the ODEs for higher moments of component counts, it is not formally known whether the approximations converge in the same regime as the means. The authors of [4] provide a heuristic justification for the convergence of the second order ODEs. Figure 7 shows how increasing the simulation interval length and changing the scale to $n = 10$ influences the error in the variance approximation from the second order hybrid analysis.

Similar to the mean, the approximations of the variance of the *Client* component with large populations is quite accurate and improves well with scaling. The error is present in case of the *Server_broken* component with small populations, but disappears with the increased scale.

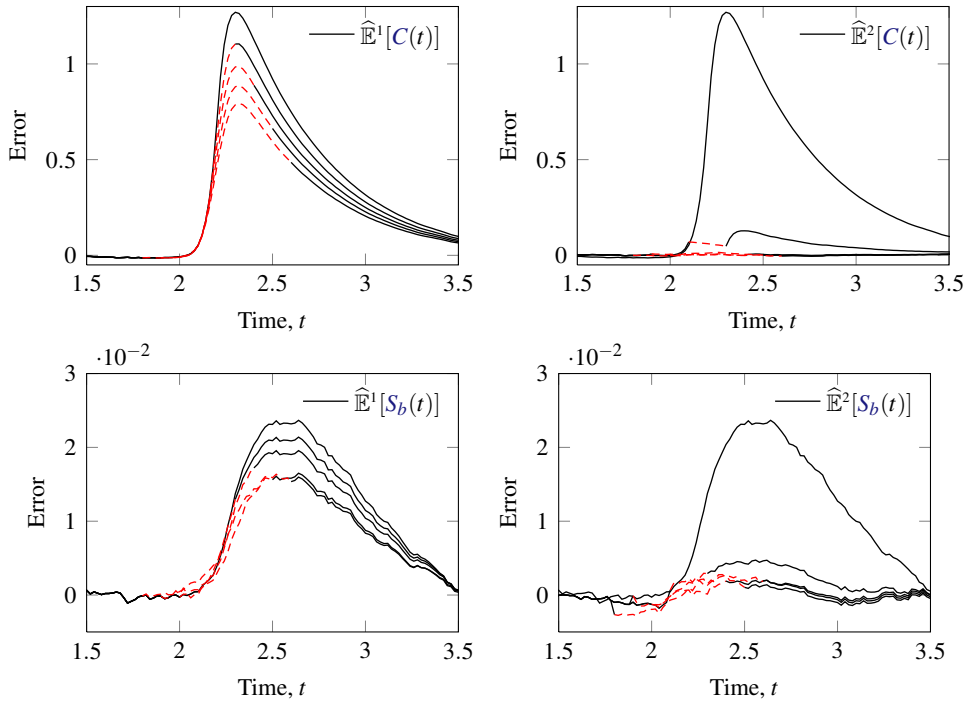

Figure 6: Effect of scaling on the error of the two hybrid analyses – plots from Figure 5 are reproduced with $n = 10$.

It is important to note that in order to produce all the above graphs, sufficiently many simulations had to be run. In case of the scaled model and the variance of the *Server_broken* component with low populations, even 10^8 replications taking several hours were not enough to obtain smooth estimates of the variance.

4 Conclusion

We have introduced two hybrid analyses of large scale PEPA models. From one point of view, these improve the accuracy of the ODE analysis by using stochastic simulation for chosen time intervals. From another point of view, they improve the efficiency of stochastic simulation by using numerical solution to ODEs in chosen time intervals. We have demonstrated that the improvements can be significant in some cases.

The main limitation is that there are no guaranteed bounds on the accuracy of the resulting approximations and the error behaviour is complicated due to the components with low populations. However, the same already applies to the ODE analysis approximations. We believe that the hybrid analyses can contribute to the investigations into the relationship between the switch point distance and the error of the approximations, possibly leading to a quantitative formulation of the relationship. We plan to investigate techniques that would automatically derive the simulation time intervals, given the total computational cost that can be spent on simulation.

Another area for future work is in generating the samples with correct means and covariances for starting values of the simulations. We plan to improve the current ad-hoc correction by finding a better distribution, for example the *truncated* multivariate normal distribution. Additionally, higher moments such as skewness and kurtosis could be employed with a suitable distribution to increase the resulting accuracy.

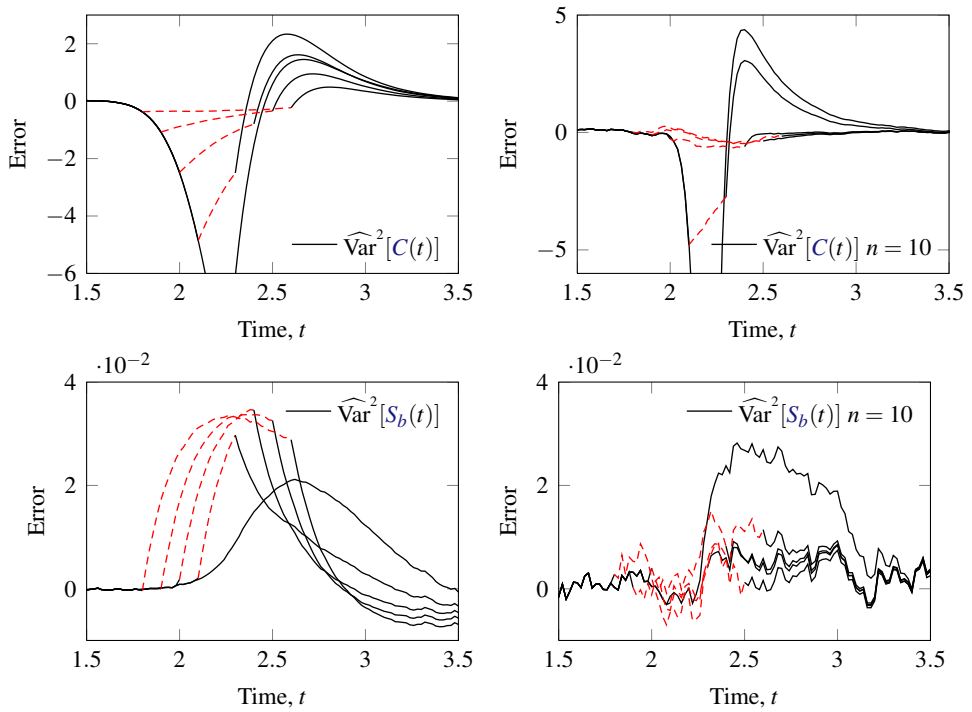

Figure 7: Effects of increasing the simulation interval length and population scaling on the error of the variance approximations from the second order hybrid analysis.

Finally, it is straightforward to use the work in [8] to extend the hybrid analyses to support means and covariances of accumulated rewards.

References

[1] T. Kurtz, "Solutions of Ordinary Differential Equations as Limits of Pure Jump Markov Processes," vol. 7, pp. 49–58, Apr. 1970.

[2] J. Hillston, "Fluid flow approximation of PEPA models," in *Second International Conference on the Quantitative Evaluation of Systems (QEST'05)*, pp. 33–42, IEEE, 2005.

[3] R. A. Hayden and J. T. Bradley, "A fluid analysis framework for a Markovian process algebra," *Theoretical Computer Science*, vol. 411, pp. 2260–2297, May 2010.

[4] A. Stefanek, R. A. Hayden, and J. T. Bradley, "A new tool for the performance analysis of massively parallel computer systems," in *Eighth Workshop on Quantitative Aspects of Programming Languages (QAPL 2010), March 27-28, 2010, Paphos, Cyprus*, Electronic Proceedings in Theoretical Computer Science, Mar. 2010.

[5] L. Bortolussi and A. Policriti, "Hybrid dynamics of stochastic programs," *Theoretical Computer Science*, vol. 411, pp. 2052–2077, Apr. 2010.

[6] A. Stefanek, R. Hayden, and J. T. Bradley, "GPA - Tool for rapid analysis of very large scale PEPA models," in *UKPEW'10, 7-8th July, University of Warwick*, pp. 91–101, July 2010.

[7] J. Wang and C. Liu, "Generating multivariate mixture of normal distributions using a modified Cholesky decomposition," *Winter Simulation Conference*, p. 342, 2006.

[8] A. Stefanek, R. A. Hayden, and J. T. Bradley, "Rapid analysis of Rewards and Completion times in massively parallel Markov models." In preparation, 2010.