

# Fluid Semantics for Passive Stochastic Process Algebra Cooperation

Richard A. Hayden

Dept. of Computing, Imperial College London  
180 Queen's Gate, London SW7 2BZ, UK  
rh@doc.ic.ac.uk

Jeremy T. Bradley

Dept. of Computing, Imperial College London  
180 Queen's Gate, London SW7 2BZ, UK  
jb@doc.ic.ac.uk

## ABSTRACT

Fluid modelling is a next-generation technique for analysing massive performance models. Passive cooperation is a popular cooperation mechanism frequently used by performance engineers. Therefore having an accurate translation of passive cooperation into a fluid model is of direct practical application. We compare different existing styles of fluid model translation of passive cooperation in a stochastic process algebra. We explain why the development of a fluid semantics for passive cooperation is not straightforward and we present an alternative definition which more closely matches the underlying discrete model. Finally, we present quantitative comparisons with a previous version of the fluid semantics in which numerical discrepancies can be observed.

## 1. INTRODUCTION

Fluid analysis of performance models offers the exciting potential for the analysis of massive state-spaces at small computational cost. In the case of stochastic process algebra models, fluid analysis involves approximating the underlying discrete state space with continuous real-valued variables and describing the time-evolution of those variables with ordinary differential equations (ODEs). This approach was first applied to a subset of the stochastic process algebra PEPA by Hillston [9].

Since then, we have been looking at reasonable extensions of this approach which can translate bigger subsets of PEPA to a fluid model. This would therefore give the modeller more access to the full expressiveness of the PEPA language when it comes to using fluid analysis techniques.

One such language feature, for which it has been particularly tricky to provide a satisfactory fluid semantics, is passive cooperation. Passive cooperation, in process algebra terms, describes a popular style of synchronisation between cooperating components where one component passively waits for the other component to perform its action. This can be seen simply in a client-server scenario, where initially the server waits passively for a client to issue a re-

quest. When constructing a continuous variable to represent the number of replications of a passive component in a system, a key issue was how to disable the passive cooperation in the fluid model as the number of copies of the passive component approached zero.

A semantics for passive cooperation was suggested in [3] but, as we will discuss, this involved the use of indicator functions in the fluid model. In this case, reasonable solutions to the system of ODEs are not always even guaranteed to exist. Indeed, as investigated in this paper, the approach taken in the literature to overcome this introduced significant quantitative and qualitative errors in the resulting analysis.

In this paper, we will explore the existing fluid semantics for passive cooperation and suggest a replacement model that is more closely related to the underlying continuous-time Markov chain (CTMC) of the PEPA model. We will show empirically how the new fluid model generates more realistic results and we will apply this to an example client-server model with two-stage fetch and server breakdowns.

In the following section, Section 1.1, we introduce the stochastic process algebra PEPA and in Section 1.2, we introduce a new way of expressing the fluid model originally given in [9]. In Section 1.3, we present the existing passive fluid semantics, while in Section 2 we show how a modified active cooperation can be used to represent passive cooperation. This leads naturally to a new fluid semantics for passive cooperation. Finally, in Sections 2.4 and 3, we look at some empirical comparisons between the old and new techniques.

### 1.1 PEPA

PEPA [8] as a performance modelling formalism has been used to study a wide variety of systems: multimedia applications [1], mobile phone usage [6], GRID scheduling [11], production cell efficiency [10] and web-server clusters [2] amongst others. The definitive reference for the language is [8].

As in all process algebras, systems are represented in PEPA as the composition of *components* which undertake *actions*. In PEPA the actions are assumed to have a duration, or delay. Thus the expression  $(\alpha, r).P$  denotes a component which can undertake an  $\alpha$  action at rate  $r$  to evolve into a component  $P$ . Here  $\alpha \in \mathcal{A}$  where  $\mathcal{A}$  is the set of action types. The rate  $r$  is interpreted as a random delay which samples from an exponential random variable with parameter,  $r$ .

PEPA has a small set of combinators, allowing system descriptions to be built up as the concurrent execution and interaction of simple sequential components. The syntax of

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ValueTools 2008, October 21 - 23, 2008, Athens, GREECE  
Copyright 2008 ICST 978-963-9799-31-8.

the type of PEPA model considered in this paper may be formally specified using the following grammar:

$$\begin{aligned} S &::= (\alpha, r).S \mid S + S \mid C_S \\ P &::= P \underset{L}{\bowtie} P \mid P/L \mid C \end{aligned} \quad (1)$$

where  $S$  denotes a *sequential component* and  $P$  denotes a *model component* which executes in parallel.  $C$  stands for a constant which denotes either a sequential component or a model component as introduced by a definition.  $C_S$  stands for constants which denote sequential components. The effect of the syntactic separation between these types of constants is to constrain legal PEPA components to be cooperations of sequential processes.

More information and structured operational semantics on PEPA can be found in [8]. A brief discussion of the basic PEPA operators is given below:

**Prefix** The basic mechanism for describing the behaviour of a system with a PEPA model is to give a component a designated first action using the prefix combinator, denoted by a full stop, which was introduced above. As explained,  $(\alpha, r).P$  carries out an  $\alpha$  action with rate  $r$ , and it subsequently behaves as  $P$ .

**Choice** The component  $P + Q$  represents a system which may behave either as  $P$  or as  $Q$ . The activities of both  $P$  and  $Q$  are enabled. The first activity to complete distinguishes one of them: the other is discarded. The system will behave as the derivative resulting from the evolution of the chosen component.

**Constant** It is convenient to be able to assign names to patterns of behaviour associated with components. Constants are components whose meaning is given by a defining equation. The notation for this is  $X \stackrel{\text{def}}{=} E$ . The name  $X$  is in scope in the expression on the right hand side meaning that, for example,  $X \stackrel{\text{def}}{=} (\alpha, r).X$  performs  $\alpha$  at rate  $r$  forever.

**Hiding** The possibility to abstract away some aspects of a component's behaviour is provided by the hiding operator, denoted  $P/L$ . Here, the set  $L$  identifies those activities which are to be considered internal or private to the component and which will appear as the unknown type  $\tau$ .

**Cooperation** We write  $P \underset{L}{\bowtie} Q$  to denote cooperation between  $P$  and  $Q$  over  $L$ . The set which is used as the subscript to the cooperation symbol, the *cooperation set*  $L$ , determines those activities on which the components are forced to synchronise. For action types not in  $L$ , the components proceed independently and concurrently with their enabled activities. We write  $P \parallel Q$  as an abbreviation for  $P \underset{L}{\bowtie} Q$  when  $L$  is empty. Further, particularly useful in fluid analysis is,  $P[n]$  which is shorthand for the parallel cooperation of  $n$   $P$ -components,  $\underbrace{P \parallel \dots \parallel P}_n$ .

In process cooperation, if a component enables an activity whose action type is in the cooperation set it will not be able to proceed with that activity until the other component also enables an activity of that type. The two components then proceed together to complete the *shared activity*. Once

enabled, the rate of a shared activity has to be altered to reflect the slower component in a cooperation.

In some cases, when a shared activity is known to be completely dependent only on one component in the cooperation, then the other component will be made *passive* with respect to that activity. This means that the rate of the activity is left unspecified (denoted  $\top$ ) and is determined upon cooperation, by the rate of the activity in the other component. All passive actions must be synchronised in the final model.

Within the cooperation framework, PEPA respects the definition of *bounded capacity*: that is, a component cannot be made to perform an activity faster by cooperation, so the rate of a shared activity is the minimum of the apparent rates of the activity in the cooperating components.

The definition of the derivative set of a component will be needed later in the paper. The derivative set,  $ds(C)$ , is the set of states that can be reached from the state  $C$ . If  $C$  is a state in a strongly connected sequential component,  $ds(C)$  represents the state space of that component.

The total capacity of a component  $P$  to carry out activities of type  $\alpha$  is termed the *apparent rate* of  $\alpha$  in  $P$ , denoted  $r_\alpha(P)$ . It is used heavily when calculating the pairwise cooperation rate: when cooperating with another component, the bounded capacity principle ensures that the overall rate of cooperation does not exceed either of the constituent apparent rates.

To summarise the original ruleset from [8], the apparent rate function can be defined as:

$$r_\alpha(P) = \sum_{P \xrightarrow{(\alpha, \lambda_i)}} \lambda_i \quad (2)$$

This is the sum of the rates of all  $\alpha$ -activities enabled by  $P$ , i.e.  $P \xrightarrow{(\alpha, \lambda_i)}$  means that  $P$  enables an  $\alpha$  activity at rate  $\lambda_i \in \mathbb{R}^+ \cup \{n\top \mid n \in \mathbb{Q}, n > 0\}$ .  $n\top$  is shorthand for  $n \times \top$  and  $\top$  represents the passive action rate that inherits the rate of the coaction from the cooperating component.  $\top$  requires the following arithmetic rules:

$$\begin{aligned} m\top < n\top &: \text{for } m < n \text{ and } m, n \in \mathbb{Q} \\ r < n\top &: \text{for all } r \in \mathbb{R}, n \in \mathbb{Q} \\ m\top + n\top = (m+n)\top &: m, n \in \mathbb{Q} \\ \frac{m\top}{n\top} = \frac{m}{n} &: m, n \in \mathbb{Q} \end{aligned}$$

Note that  $(r + n\top)$  is undefined for all  $r \in \mathbb{R}$ ,  $r \neq 0$ ,  $n \in \mathbb{Q}$ ,  $n \neq 0$  in PEPA therefore implicitly disallowing components which enable both active and passive actions in the same action type at the same time, e.g.  $(a, \lambda).P + (a, \top).P'$ . Also, we define the multiset  $Act(P)$  as the activities enabled by  $P$  (this is a multiset because  $P$  may enable an activity  $(\alpha, \lambda)$  through more than one transition):

$$Act(P) = \{(\alpha, \lambda) : P \xrightarrow{(\alpha, \lambda)} \}$$

## 1.2 Fluid semantics for active cooperation

Fluid semantics for PEPA have been presented in a number of different ways in the literature [9, 3]. In each case, the class of models for which fluid semantics are presented differs, but the semantics are equivalent in each case where the class of models considered overlaps. A fluid semantics for passive cooperation is given in only one case [3], and we discuss the limitations of this approach in Section 1.3. For now,

we present the existing fluid semantics for PEPA not including passive cooperation. To do this in a manner which will aid in explaining the contributions of this paper, we adopt a new presentational approach, extending the standard PEPA grammar of Equation (1) to support explicit identification of *component groups* using *component group labels*. A component group  $D$  is simply a parallel cooperation (involving no synchronisation) of standard PEPA components  $P$ :

$$D ::= D \parallel D \mid P \quad (3)$$

For the moment, we avoid having to provide a fluid semantics for passive cooperation by explicitly asserting that the standard PEPA components  $P$  in Equation (3) are such that all  $(\alpha, r) \in \mathcal{A}ct(P')$  for any  $P' \in ds(P)$  have  $r \in \mathbb{R}^+$ ; that is, neither they nor any of their derivative states enable any action passively. A *grouped PEPA model*  $M$  is then an arbitrary combination of labelled component groups:

$$M ::= M \boxtimes_L M \mid M/L \mid Y\{D\} \quad (4)$$

where  $L$  is a set of action types. The term  $Y\{D\}$  is a *labelled component group* and extends the original PEPA syntax.  $Y$  is a unique component group label drawn from some sufficiently large label set  $\mathcal{T}$ . The operational semantics for this augmented version of PEPA are the natural extension of the standard PEPA operational semantics [8]. The only difference is that the explicit identification of component groups is maintained as the model evolves. An example grouped PEPA model might therefore be:

$$\begin{aligned} P_0 &\stackrel{def}{=} (a, r_1).P_1 & R_0 &\stackrel{def}{=} (a, r_2).R_1 \\ P_1 &\stackrel{def}{=} (b, q_1).P_0 & R_1 &\stackrel{def}{=} (b, q_2).R_0 \end{aligned} \quad (5)$$

$$Sys \stackrel{def}{=} \text{Processors}\{P_0[N]\}_{\{\alpha\}} \boxtimes \text{Resources}\{R_0[M]\}$$

This might represent  $N$  processors (grouped together explicitly in the **Processors** component group) which first undertake a shared  $a$ -action with one of  $M$  resources (grouped together explicitly in the **Resources** component group) before the processor and resource in question each complete independent  $b$ -actions. The quantities which will be subject to the fluid approximation are exposed through an aggregation of the model's state space. For example, in this case, there are  $N \times M$  different ways the initial shared  $a$ -action can be performed because it involves exactly one  $P_0$  and exactly one  $R_0$  component. As defined by the standard PEPA operational semantics [8], each of these transitions occurs at rate:

$$\frac{1}{N} \frac{1}{M} \min(Nr_1, Mr_2)$$

The aggregation collects states together based on the number of each type of component in each component group. In the case of the *Sys* component, we would represent the initial aggregate state informally as " $N \times P_0, 0 \times P_1, M \times R_0$  and  $0 \times R_1$  components". All of the  $N \times M$  transitions above would thus become one transition from the aggregate state " $N \times P_0, 0 \times P_1, M \times R_0$  and  $0 \times R_1$  components" to the aggregate state " $(N-1) \times P_0, 1 \times P_1, (M-1) \times R_0$  and  $1 \times R_1$  components" at aggregate rate  $\min(Nr_1, Mr_2)$ .

More formally, each state of the underlying aggregated CTMC of a grouped PEPA model can be uniquely determined by the model's initial derivative state  $G$  and a function  $E \in \mathcal{G}(G) \times \mathcal{P} \rightarrow \mathbb{Z}_{\geq 0}$  which counts the number of standard PEPA components of each type currently active in

a given component group; that is,  $\mathcal{G}(G)$  is the set of component group labels in the model  $G$  and  $\mathcal{P}$  is the set of standard PEPA component derivative states in the model. Conversely, note that not all such functions specify valid states in the underlying aggregated CTMC (for example, if it specifies a total number of components in a component group that exceeds the component group's size, or specifies an otherwise unreachable CTMC state).

We may then define the *component rate function* for a grouped PEPA model  $G$ , which calculates the aggregate rate at which a standard PEPA component  $P$  within a component group  $H$  completes an action  $\alpha$ , in the aggregate state specified by  $E$ . This is needed to describe the rate of evolution of a component group from one derivative state to the next when constructing the fluid model.

DEFINITION 1 (COMPONENT RATE FUNCTION).

$$\mathcal{R}_\alpha(M_1 \boxtimes_L M_2, E, H, P) := \begin{cases} \frac{\mathcal{R}_\alpha(M_1, E, H, P)}{r_\alpha(M_1, E)} \min(r_\alpha(M_1, E), r_\alpha(M_2, E)) & \text{if } H \in \mathcal{G}(M_1) \text{ and } \alpha \in L \\ \mathcal{R}_\alpha(M_1, E, H, P) & \text{if } H \in \mathcal{G}(M_1) \text{ and } \alpha \notin L \\ \frac{\mathcal{R}_\alpha(M_2, E, H, P)}{r_\alpha(M_2, E)} \min(r_\alpha(M_1, E), r_\alpha(M_2, E)) & \text{if } H \in \mathcal{G}(M_2) \text{ and } \alpha \in L \\ \mathcal{R}_\alpha(M_2, E, H, P) & \text{if } H \in \mathcal{G}(M_2) \text{ and } \alpha \notin L \end{cases}$$

$$\mathcal{R}_\alpha(M/L, E, H, P) := \begin{cases} \mathcal{R}_\alpha(M, E, H, P) & \text{if } \alpha \notin L \text{ and } \alpha \neq \tau \\ 0 & \text{if } \alpha \in L \text{ (and } \alpha \neq \tau) \\ \sum_{\beta_i \in L \cup \{\tau\}} \mathcal{R}_{\beta_i}(M, E, H, P) & \text{if } \alpha = \tau \end{cases}$$

$$\mathcal{R}_\alpha(Y\{D\}, E, H, P) := \begin{cases} E(H, P) r_\alpha(P) & \text{if } H = Y \\ 0 & \text{otherwise} \end{cases}$$

Rational functions with zero-valued denominators in the above are defined to be zero. We similarly require the definition of apparent rate to be extended to the component group model,  $r_\alpha(Y\{D\}, E)$ . As with normal PEPA models, it captures the overall rate of an action within a structure (in this case the ability of the component group,  $Y\{D\}$ , to carry out an action  $\alpha$ ).  $r_\alpha(Y\{D\}, E)$  is defined similarly to that of standard PEPA (Equation (2)), apart from the explicit specification of component counts by  $E$ .

DEFINITION 2 (COUNT-ORIENTED APPARENT RATE).

$$r_\alpha(M_1 \boxtimes_L M_2, E) := \begin{cases} \min(r_\alpha(M_1, E), r_\alpha(M_2, E)) & \text{if } \alpha \in L \\ r_\alpha(M_1, E) + r_\alpha(M_2, E) & \text{otherwise} \end{cases}$$

$$r_\alpha(M/L, E) := \begin{cases} r_\alpha(M, E) & \text{if } \alpha \notin L \\ 0 & \text{otherwise} \end{cases}$$

$$r_\alpha(Y\{D\}, E) := \sum_{\substack{P_i \xrightarrow{(\alpha, \lambda_j)} \\ P_i \in \mathcal{P}}} E(Y, P_i) \lambda_j$$

For a given component group label  $H$  and standard PEPA component  $P$ , we introduce the integer-valued stochastic process  $n_{H,P}(t)$  which counts the number of  $P$ -components active at a given time  $t \geq 0$  within the component group,  $H$ . We intend to define real-valued deterministic functions  $v_{H,P}(t)$  as fluid approximations to the  $n_{H,P}(t)$ , in some sense.

For some time  $t \geq 0$ , define  $E_t \in \mathcal{G}(G) \times \mathcal{P} \rightarrow \mathbb{Z}_{\geq 0}$  such that  $E_t(H, P) = n_{H,P}(t)$  for all  $H \in \mathcal{G}(G)$  and  $P \in \mathcal{P}$ . It is clear that  $E_t$  represents the aggregate CTMC state at time  $t$ . Then it can be shown that  $\mathcal{R}_\alpha(G, E_t, H, P)$  is simply the sum of the rates of all outgoing  $\alpha$ -transitions from the current aggregate CTMC state to any other where the number of  $P$  components in group  $H$  is decreased. In order to consider outgoing transitions which increase a component count, we need to make one further definition. We define the *derivative weighting function* which calculates the probability that given that a standard PEPA component  $P$  does an  $\alpha$ -action, when it does so, it transits to another specified standard PEPA component  $Q$ .

DEFINITION 3 (DERIVATIVE WEIGHTING FUNCTION).

$$p_\alpha(P, Q) := \frac{1}{r_\alpha(P)} \sum_{P \xrightarrow{(\alpha, \lambda_i)} Q} \lambda_i$$

Then it is also the case that the sum of all outgoing  $\alpha$ -transitions from CTMC states where the number of  $P$  components in group  $H$  is increased is:

$$\sum_{\substack{Q \in \mathcal{P} \\ Q \neq P}} p_\alpha(Q, P) \mathcal{R}_\alpha(G, E_t, H, Q)$$

Furthermore, it is easy to see that all increases and decreases of component counts are one component at a time. It then makes at least heuristic sense to define the fluid approximations  $v_{H,P}(t)$  as the solution to the system of ordinary differential equations obtained by balancing these rates of *increasing* and *decreasing* CTMC transitions for each component. In particular, we assume a known initial CTMC configuration  $n_{H,P}(0)$  and define  $v_{H,P}(0) := n_{H,P}(0)$ . The functions  $v_{H,P}(t)$  for  $t > 0$  are then defined as the solutions to the initial value problem obtained from the following system of ODEs (where  $V(H, P, t) := v_{H,P}(t)$  and the component and apparent rate function definitions are naturally extended to the real numbers):

$$\left\{ \begin{aligned} \frac{dv_{H,P}(t)}{dt} = & \\ & \sum_{\alpha \in \mathcal{A}} \sum_{\substack{Q \in \mathcal{P} \\ Q \neq P}} \left[ p_\alpha(Q, P) \mathcal{R}_\alpha(G, V(\cdot, \cdot, t), H, Q) \right] \\ & - \mathcal{R}_\alpha(G, V(\cdot, \cdot, t), H, P) : P \in \mathcal{P}, H \in \mathcal{G}(G) \end{aligned} \right\} \quad (6)$$

We now illustrate this by considering again the grouped PEPA model of Equation (5). In this context and writing  $P_0(t)$  for  $v_{P,P_0}(t)$  and so on, Equation (6) yields the system of ODEs:

$$\begin{aligned} \dot{P}_0(t) &= -\min(P_0(t) \cdot r_1, R_0(t) \cdot r_2) + P_1(t) \cdot q_1 \\ \dot{P}_1(t) &= -P_1(t) \cdot q_1 + \min(P_0(t) \cdot r_1, R_0(t) \cdot r_2) \\ \dot{R}_0(t) &= -\min(P_0(t) \cdot r_1, R_0(t) \cdot r_2) + R_1(t) \cdot q_2 \\ \dot{R}_1(t) &= -R_1(t) \cdot q_2 + \min(P_0(t) \cdot r_1, R_0(t) \cdot r_2) \end{aligned}$$

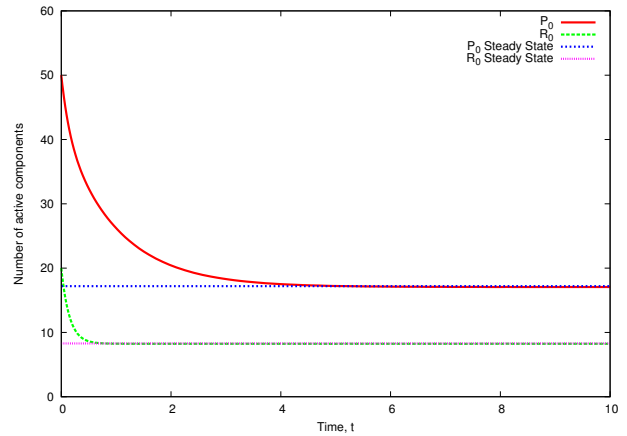


Figure 1: Comparison of ODE solutions with steady-state solutions of the underlying CTMC for the model of Equation (5). Rates used are  $r_1 = 4.0$ ,  $r_2 = 4.0$ ,  $q_1 = 1.0$  and  $q_2 = 2.8$ . Initial conditions are  $N = 50$   $P_0$  and  $M = 20$   $R_0$  components.

Figure 1 shows the result of integrating these ODEs compared with the steady-state populations computed by traditional methods.

The general success of such fluid analysis techniques for models involving only active cooperation is well supported by the literature [5, 9]. However, in the case of models involving passive cooperation, fluid analysis techniques are less well developed.

### 1.3 Existing passive fluid semantics

One fluid semantics for passive cooperation has already been proposed in the literature [3] and has seen some use in modelling real systems [7, 4]. It is a literal extension of the existing fluid semantics for active cooperation (Section 1.2). That is, ODEs are constructed using the aggregate CTMC rate, extended as a function of the component populations naturally to the real numbers. However, in the case of passive cooperation, this results in ODEs where the derivative is specified by a discontinuous function. Consider again the example of Equation (5), modified now to exhibit passive as opposed to active cooperation (also, for the sake of brevity, we have renamed the component groups to  $\mathbf{P}$  and  $\mathbf{R}$  respectively):

$$\begin{aligned} P_0 &\stackrel{\text{def}}{=} (a, r).P_1 & R_0 &\stackrel{\text{def}}{=} (a, \top).R_1 \\ P_1 &\stackrel{\text{def}}{=} (b, q_1).P_0 & R_1 &\stackrel{\text{def}}{=} (b, q_2).R_0 \\ Sys &\stackrel{\text{def}}{=} P\{P_0[N]\} \boxtimes_{\{a\}} R\{R_0[M]\} \end{aligned} \quad (7)$$

If ODEs are constructed from this model by using the aggregate CTMC rate, we obtain the following system:

$$\begin{aligned} \dot{P}_0(t) &= -\mathbf{I}(R_0(t)) \cdot P_0(t) \cdot r + P_1(t) \cdot q_1 \\ \dot{P}_1(t) &= -P_1(t) \cdot q_1 + \mathbf{I}(R_0(t)) \cdot P_0(t) \cdot r \\ \dot{R}_0(t) &= -\mathbf{I}(R_0(t)) \cdot P_0(t) \cdot r + R_1(t) \cdot q_2 \\ \dot{R}_1(t) &= -R_1(t) \cdot q_2 + \mathbf{I}(R_0(t)) \cdot P_0(t) \cdot r \end{aligned} \quad (8)$$

where  $\mathbf{I}(\cdot)$  is a discontinuous indicator function defined on



non-negative real numbers:

$$\mathbf{I}(x) := \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0 \end{cases}$$

Given the discontinuous nature of these ODEs, it is not guaranteed by the standard theoretical results that they even have a solution. Indeed, there exist many possible interpretations of the differential equations themselves. When we talk of a solution to these equations, we usually mean continuous functions which are differentiable everywhere that the right-hand side of the ODEs are continuous, with derivatives satisfying the ODEs at these points. Otherwise, the right-hand derivatives should exist and should agree with the right hand side of the ODEs. For some parameter combinations,  $R_0(t)$  never reaches zero and the ODEs have a unique solution for all times  $t \geq 0$ . Figure 2a shows one such case.<sup>1</sup> However, Figure 2b shows, for a different parameter combination, the unique solution up to the point at which  $R_0(t)$  does reach zero. In fact, there does not exist a meaningful continuation of the ODE solution past this point (it is shown simply to terminate in Figure 2b). To see this, assume there does exist such a continuation and define:

$$t_1 := \inf\{t : R_0(t) = 0\}$$

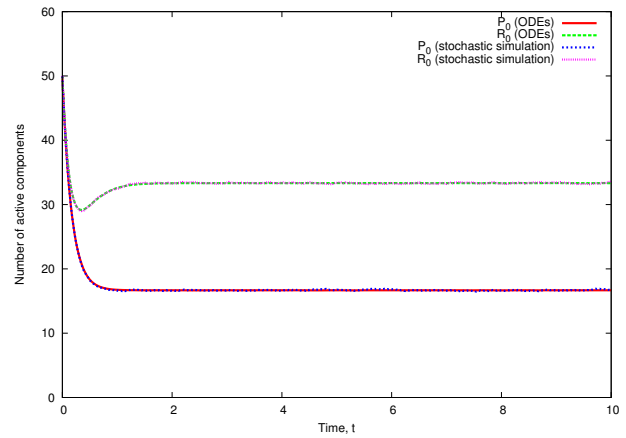
If  $R_0(t_1) \neq 0$  then by continuity of  $R_0(t)$ , we can choose a better infimum  $t_2 > t_1$ , a contradiction, so  $R_0(t_1) = 0$ . Note that in some interval  $t \in [t_1 - \epsilon, t_1)$  for  $\epsilon > 0$ ,  $\dot{R}_0(t) < 0$  ( $[0, t_1)$  works here). Now it cannot be that  $R_0(t) = 0$  in any interval  $t \in [t_1, t_1 + \delta)$ ,  $\delta > 0$  because in some non-empty sub-interval of it containing  $t_1$ , according to the ODEs,  $R_0(t) > 0$ , a contradiction. Now if  $R_0(t) < 0$  in some interval  $t \in (t_1, t_1 + \delta)$ ,  $\delta > 0$ , we would have a contradiction if we demand that the right-hand derivative at  $t_1$  must be that given by the ODEs, since the ODEs assert  $\dot{R}_0(t_1) > 0$ .<sup>2</sup> Therefore the only possibility for a meaningful solution is that in some interval  $t \in (t_1, t_1 + \delta)$ ,  $\delta > 0$ , we have  $R_0(t) > 0$ . This is a contradiction since the ODEs give  $\dot{R}_0(t) < 0$  in said interval (or some sub-interval of it  $t \in (t_1, t_1 + \delta')$ ), because we recall that we had  $\dot{R}(t) < 0$  in some interval  $t \in [t_1 - \epsilon, t_1)$ .

In practice, for instance [3], it seems such a solution is usually artificially continued such that  $R_0(t) = 0$  and  $R_1(t) = M$  are fixed for  $t \in [t_1, \infty)$ , with the further evolution of  $P_0(t)$  and  $P_1(t)$  governed by the original ODEs. Such a “solution” is shown in Figure 2c.

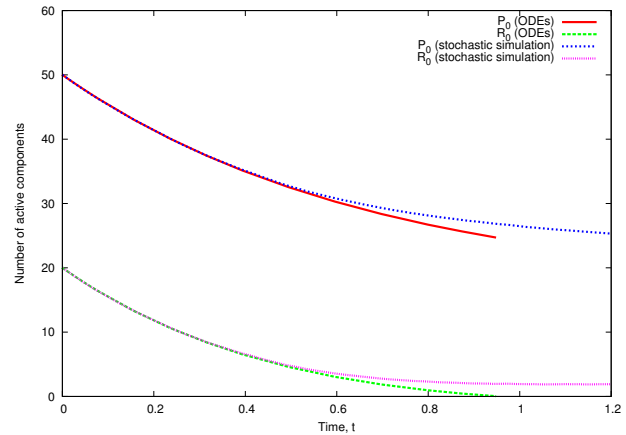
We see that under this interpretation, this passive fluid semantics performs rather poorly, especially in reflecting accurately the long-term situation when activity is effectively blocked due to a lack of  $R_0$  components, although it is often able to identify the approximate point at which the effects of blocking due to starvation become evident. Indeed, the number of  $P_0$  components is totally misrepresented and the behaviour is as if the  $R_0$  components have been structurally removed from the model.

<sup>1</sup>In all of the following figures, we will compare an ODE-derived approximation with the stochastic expectation of the quantity in question, obtained through costly repeated stochastic simulation.

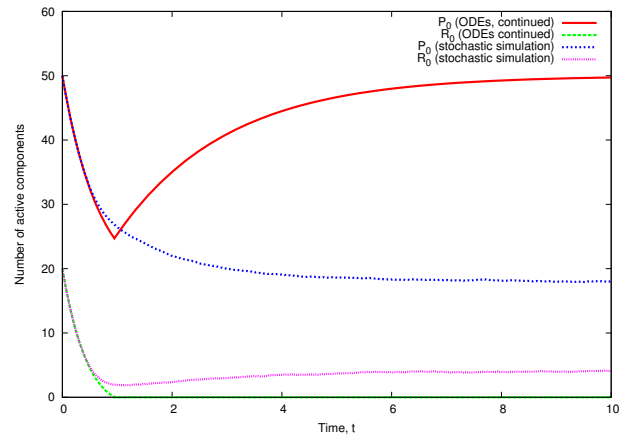
<sup>2</sup>Of course, even if we do not demand correspondence of right-hand derivatives with the ODEs at  $t_1$ , such a solution has negative component counts which are meaningless in the context of the original model.



(a) Rates:  $r = 4.0$ ,  $q_1 = 2.0$  and  $q_2 = 4.0$ . Initial conditions:  $N = 50 P_0$  and  $M = 50 R_0$  components.



(b) Rates:  $r = 1.0$ ,  $q_1 = 0.5$  and  $q_2 = 1.0$ . Initial conditions:  $N = 50 P_0$  and  $M = 20 R_0$  components. ODE solution only shown up to where it exists formally.



(c) Rates:  $r = 1.0$ ,  $q_1 = 0.5$  and  $q_2 = 1.0$ . Initial conditions:  $N = 50 P_0$  and  $M = 20 R_0$  components. ODE solution artificially-continued past where it exists formally.

Figure 2: ODE and stochastic simulation solution comparison for model of Equation (7).

In cases such as that of Figure 2a, where blocking is less likely (and in terms of the ODEs, the indicator function does not come into play), this passive fluid semantics is (predictably) more accurate.

## 2. A NEW PASSIVE FLUID SEMANTICS

We now proceed to show how instances of passive cooperation in a (grouped) PEPA model may be replaced by equivalent instances of active cooperation. This will allow us to apply the fluid analysis techniques for models with only active cooperation (as detailed in Section 1.2) to models with passive cooperation, yielding an alternative fluid semantics for passive cooperation. We will see that this approach is much more promising than the existing technique described earlier.

### 2.1 Motivation

We begin by considering again the example of Equation (7). The aggregate CTMC rate of the shared  $a$ -action is governed by the number of  $P_0$  and  $R_0$  components in the current aggregate CTMC state (say at time  $t$ ) and is given by:

$$\mathbf{I}(n_{R, R_0}(t)) \cdot n_{P, P_0}(t) \cdot r$$

Thus, if the number of  $R_0$  components is not zero, the rate is  $n_{P, P_0}(t) \cdot r$  and zero otherwise. Now since the size of each component group is a fixed property of the model and all of its derivative states, we know an upper bound on the unsynchronised rate at which the active partner(s) in a passive cooperation can perform the shared action. In the case of this example, this is  $N \cdot r$ , that is, an upper bound on the number of  $P_0$  components possible in any derivative state is the size of the P component group,  $N$ . Using this insight, we can write the aggregate CTMC rate at time  $t$  equivalently as:

$$\min(n_{P, P_0}(t) \cdot r, n_{R, R_0}(t) \cdot N \cdot r) = r \cdot \min(n_{P, P_0}(t), n_{R, R_0}(t) \cdot N)$$

If  $n_{R, R_0}(t) = 0$ , then the expression is zero as desired. Otherwise,  $n_{R, R_0}(t) \geq 1$ , and thus  $n_{R, R_0}(t) \cdot N \geq N \geq n_{P, P_0}(t)$ , so the expression is  $n_{P, P_0}(t) \cdot r$  as desired. Now the key point to note is that the above expression has the form of the aggregate CTMC rate found in an *active* cooperation. Indeed, it suggests the construction of the following equivalent grouped PEPA model, exhibiting only active cooperation:

$$\begin{aligned} P_0 &\stackrel{\text{def}}{=} (a, r).P_1 & R_0 &\stackrel{\text{def}}{=} (a, N \cdot r).R_1 \\ P_1 &\stackrel{\text{def}}{=} (b, q_1).P_0 & R_1 &\stackrel{\text{def}}{=} (b, q_2).R_0 \end{aligned} \quad (9)$$

$$\text{Sys} \stackrel{\text{def}}{=} P\{P_0[N]\} \boxtimes_{\{a\}} R\{R_0[M]\}$$

All we have changed is the definition of  $R_0$  from  $(a, \top).R_1$  to  $(a, N \cdot r).R_1$ . Both the model of Equation (7) and of Equation (9) have identical (aggregated) derivation graphs (and thus underlying (aggregated) CTMCs). Passive cooperation can thus be viewed as syntactic sugar for a particular active cooperation. In fact, there is of course an infinite family of grouped PEPA models exhibiting only active cooperation equivalent to the original passive model, we may choose any rate  $q \geq N \cdot r$  in  $R_0 \stackrel{\text{def}}{=} (a, q).R_1$ .

We will see that this point of view is a better one to adopt when considering the analysis of models involving passive cooperation in a fluid sense.

## 2.2 Formal translation

We now define the above transformation formally for an arbitrary grouped PEPA model. The first notion we define is for a given component group and action, the lowest rate at which standard PEPA components within that component group must perform the action to be *effectively passive*.

DEFINITION 4 (EXT. RATE OF EFFECTIVE PASSIVITY). For a grouped PEPA model  $G$ , action  $\alpha \neq \tau$  and component group  $H$ , the external rate of effective passivity is  $\mathcal{E}_\alpha(G, H) := \mathcal{E}'_\alpha(G, H, 0)$ , where  $\mathcal{E}'_\alpha(\cdot, \cdot, \cdot)$  is defined as:

$$\mathcal{E}'_\alpha(M_1 \boxtimes_L M_2, H, r) := \begin{cases} \mathcal{E}'_\alpha(M_1, H, \max(r, r_\alpha^{\max}(M_2))) & \text{if } H \in \mathcal{G}(M_1), \alpha \in L \\ \mathcal{E}'_\alpha(M_1, H, r) & \text{if } H \in \mathcal{G}(M_1), \alpha \notin L \\ \mathcal{E}'_\alpha(M_2, H, \max(r, r_\alpha^{\max}(M_1))) & \text{if } H \in \mathcal{G}(M_2), \alpha \in L \\ \mathcal{E}'_\alpha(M_2, H, r) & \text{if } H \in \mathcal{G}(M_2), \alpha \notin L \end{cases}$$

$$\mathcal{E}_\alpha(M/L, H, r) := \begin{cases} \mathcal{E}_\alpha(M, H, r) & \text{if } \alpha \notin L \\ \mathcal{E}_\alpha(M, H, 0) & \text{if } \alpha \in L \end{cases}$$

$$\mathcal{E}_\alpha(H\{D\}, H, r) := r$$

Note that for any legal (synchronised) passive enabling of an action within component group  $H$ ,  $\mathcal{E}_\alpha(G, H) \neq 0$ .  $r_\alpha^{\max}(\cdot)$  is the *maximum real apparent rate*.

DEFINITION 5 (MAXIMUM REAL APPARENT RATE). For a grouped PEPA model  $G$  and action  $\alpha \neq \tau$ , the maximum real apparent rate is  $r_\alpha^{\max}(G)$ , where  $r_\alpha^{\max}(\cdot)$  is defined as:

$$\begin{aligned} r_\alpha^{\max}(M_1 \boxtimes_L M_2) &:= \begin{cases} \min(r_\alpha^{\max}(M_1), r_\alpha^{\max}(M_2)) & \text{if } \alpha \in L \\ r_\alpha^{\max}(M_1) + r_\alpha^{\max}(M_2) & \text{if } \alpha \notin L \end{cases} \\ r_\alpha^{\max}(M/L) &:= \begin{cases} r_\alpha^{\max}(M) & \text{if } \alpha \notin L \\ 0 & \text{if } \alpha \in L \end{cases} \\ r_\alpha^{\max}(Y\{D\}) &:= \mathcal{S}(Y\{D\}) \times \max_{P \in \mathcal{J}(Y\{D\})} r_\alpha^{\max}(P) \end{aligned}$$

where for standard PEPA components

$$r_\alpha^{\max}(P) := \max_{P_i \in ds(P)} \sum_{\substack{\lambda_j \in \mathbb{R} \\ \xrightarrow{(\alpha, \lambda_j)} P_i}} \lambda_j$$

$\mathcal{S}(\cdot)$  and  $\mathcal{J}(\cdot)$  are respectively the size of the component group and the set of standard PEPA components in the component group.

It should be noted here that naming this quantity maximum real apparent rate is not entirely accurate. Indeed, it is not necessarily the case that:

$$r_\alpha^{\max}(G) = \max\{r_\alpha(G_i) : G_i \in ds(G) \text{ and } r_\alpha(G_i) \in \mathbb{R}\}$$

since not all component group configurations are necessarily reachable. Using this definition instead might result in a tighter (lower) version of the external rate of effective passivity.<sup>3</sup> However, the evaluation requires a potentially costly

<sup>3</sup>Whether this is actually desirable is related to the question of which of the infinite family of equivalent active models to choose which is discussed in Section 2.4.

expansion of  $ds(G)$ , whereas the definition we have used does not.

It is not sufficient simply to replace all standard PEPA prefix sub-components  $(\alpha, \top).P$  within component group  $H$  with  $(\alpha, \mathcal{E}_\alpha(G, H)).P$ . Firstly, as well as being effectively passive to other component groups, such standard PEPA components must also maintain their effective passivity within their enclosing standard PEPA structure. The following function on standard PEPA components computes an *internal rate of effective passivity* for this purpose.

**DEFINITION 6 (INT. RATE OF EFFECTIVE PASSIVITY).**

For any standard PEPA component  $P$  and action type  $\alpha \neq \tau$ , the internal rate of effective passivity  $\mathcal{E}_\alpha^{\text{int}}(P) := \mathcal{E}_\alpha^{\text{int}}(P, 0)$  is a lower bound with which passive rates can be replaced while still maintaining effective passivity within the standard PEPA component, where  $\mathcal{E}_\alpha^{\text{int}}(\cdot, \cdot)$  is defined as:

$$\mathcal{E}_\alpha^{\text{int}}(P_1 \bowtie_L P_2, r) := \begin{cases} \max(\mathcal{E}_\alpha^{\text{int}}(P_1, \max(r, r_\alpha^{\text{max}}(P_2))), \\ \mathcal{E}_\alpha^{\text{int}}(P_2, \max(r, r_\alpha^{\text{max}}(P_1)))) \\ \text{if } \alpha \in L \\ \max(\mathcal{E}_\alpha^{\text{int}}(P_1, r), \mathcal{E}_\alpha^{\text{int}}(P_2, r)) \\ \text{if } \alpha \notin L \end{cases}$$

$$\mathcal{E}_\alpha^{\text{int}}(P/L, r) := \begin{cases} \mathcal{E}_\alpha^{\text{int}}(P, 0) & \text{if } \alpha \in L \\ \mathcal{E}_\alpha^{\text{int}}(P, r) & \text{if } \alpha \notin L \end{cases}$$

$$\mathcal{E}_\alpha^{\text{int}}(S, r) := r$$

Define  $\mathcal{E}_\alpha^{\text{int}}(G, H) := \max_{P \in \mathcal{J}(G, H)} \mathcal{E}_\alpha^{\text{int}}(P)$  and then it is sufficient to replace all standard prefix sub-components  $(\alpha, \top).P$  within component group  $H$  with:

$$(\alpha, \max(\mathcal{E}_\alpha(G, H), \mathcal{E}_\alpha^{\text{int}}(G, H))).P$$

in order to simultaneously guarantee both effective external and effective internal passivity.<sup>4</sup> We now have the desired result. We have omitted the proof for brevity.

**THEOREM 1.** Let  $G$  be a grouped PEPA model. For every component group  $H \in \mathcal{G}(G)$  and standard PEPA component  $P \in \mathcal{J}(H)$ , replace  $P$  in  $G$  with  $P'$  where  $P' := \mathcal{T}(P)$  to define the grouped PEPA model  $G'$ , where  $\mathcal{T}(\cdot)$  is defined as:

$$\mathcal{T}((\alpha, r).S) := \begin{cases} (\alpha, R).\mathcal{T}(S) & \text{if } r = \top \\ (\alpha, r).\mathcal{T}(S) & \text{otherwise} \end{cases}$$

where  $R := \max(\mathcal{E}_\alpha(G, H), \mathcal{E}_\alpha^{\text{int}}(G, H))$ .

$$\mathcal{T}(S_1 + S_2) := \mathcal{T}(S_1) + \mathcal{T}(S_2)$$

$$\mathcal{T}(P/L) := \mathcal{T}(P)/L$$

$$\mathcal{T}(P_1 \bowtie_L P_2) := \mathcal{T}(P_1) \bowtie_L \mathcal{T}(P_2)$$

Then  $G'$  has only active cooperation. Furthermore,  $G$  and  $G'$  have the same aggregated derivation graph (and thus underlying aggregated CTMC).

<sup>4</sup>The careful reader might wonder why we can't use each standard PEPA component's own internal rate of effective passivity instead of taking the maximum over the component group. This is because for each action type, we must be consistent within each component group in our replacing of passive rates with real rates in order to ensure the correct meaning of expressions such as  $\frac{P_0(t)\top}{(P_0(t)+P_1(t))\top} = \frac{P_0(t)}{P_0(t)+P_1(t)}$  is maintained (so that the  $\top$ s still cancel).

Of course, to state this completely formally, we must unify each standard PEPA component  $P$  in  $G$  with its replacement  $P'$  in  $G'$ .

## 2.3 Example translation

We now illustrate this result by means of a more complicated example grouped PEPA model:

$$\begin{aligned} P_0 &\stackrel{\text{def}}{=} (a, \top).P_1 & P_2 &\stackrel{\text{def}}{=} (a, \top).P_2 + (a, \top).P_3 \\ P_1 &\stackrel{\text{def}}{=} (a, r_1).P_0 & P_3 &\stackrel{\text{def}}{=} (a, r_2).P_2 \\ P &\stackrel{\text{def}}{=} P_0 \bowtie_{\{a\}} P_2 \\ R_0 &\stackrel{\text{def}}{=} (a, \top).R_1 & R_1 &\stackrel{\text{def}}{=} (a, r_3).R_0 \\ Q_0 &\stackrel{\text{def}}{=} (a, r_4).Q_1 & Q_1 &\stackrel{\text{def}}{=} (a, r_5).Q_0 \\ Sys &\stackrel{\text{def}}{=} \left[ P\{P[N]\} \bowtie_{\{a\}} R\{R_0[M]\} \right] \bowtie_{\{a\}} Q\{Q_0[O]\} \end{aligned}$$

For component group  $P$ :

$$\mathcal{E}_a(G, P) = \max(O \times \max(r_4, r_5), M \times r_3)$$

and  $\mathcal{E}_a^{\text{int}}(G, P) = \max(r_1, r_2)$ , so we could replace the  $\top$ s in  $P_0$  and  $P_2$  with a real value greater than or equal to the following expression:

$$\max(\max(O \times \max(r_4, r_5), M \times r_3), \max(r_1, r_2))$$

In the case of component group  $R$ , we have trivially  $\mathcal{E}_a^{\text{int}}(G, R) = 0$ . Also:

$$\mathcal{E}_a(G, R) = \max(O \times \max(r_4, r_5), N \times \max(r_1, r_2))$$

so we could replace the passive rate in  $R_0$  with the following expression:

$$\max(O \times \max(r_4, r_5), N \times \max(r_1, r_2))$$

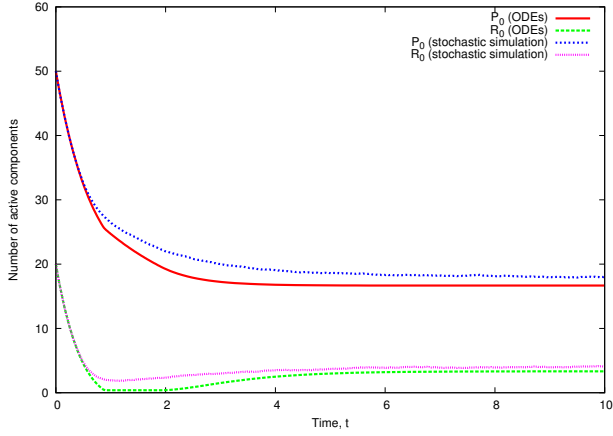
## 2.4 Evaluation

The key idea of this work is to use the result of the previous section (Theorem 1) to define a new fluid semantics for grouped PEPA models exhibiting passive cooperation. The initial idea is very simple: first we use Theorem 1 to translate the model in question to an equivalent model which exhibits only active cooperation, and then we simply apply the fluid semantics for active cooperation (presented in Section 1.2). Since the original model with passive cooperation has the same underlying aggregated CTMC as the translated one exhibiting only active cooperation, we have introduced no extra stages of approximation than when we start with a model that exhibits purely active cooperation.

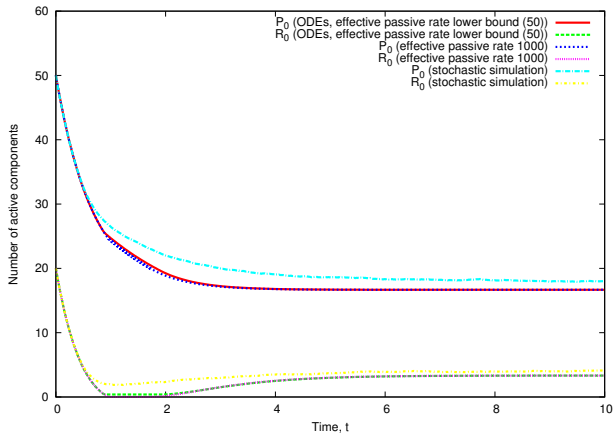
Let us return to the example of Equation (7). Applying Theorem 1 yields the following equivalent grouped PEPA model, exhibiting only active cooperation (the same as Equation (9)):

$$\begin{aligned} P_0 &\stackrel{\text{def}}{=} (a, r).P_1 & R_0 &\stackrel{\text{def}}{=} (a, N \cdot r).R_1 \\ P_1 &\stackrel{\text{def}}{=} (b, q_1).P_0 & R_1 &\stackrel{\text{def}}{=} (b, q_2).R_0 \\ Sys &\stackrel{\text{def}}{=} P\{P_0[N]\} \bowtie_{\{a\}} R\{R_0[M]\} \end{aligned} \quad (10)$$

Applying the standard fluid semantics for active cooperation from Section 1.2 yields the following system of ODEs



**Figure 3: ODE and stochastic simulation solution comparison for model of Equation (10). Rates used are  $r = 1.0$ ,  $q_1 = 0.5$  and  $q_2 = 1.0$ . Initial conditions are  $N = 50 P_0$  and  $M = 20 R_0$  components.**



**Figure 4: ODE (multiple effective passive rates) and stochastic simulation solution comparison for model of Equation (10). Rates used are  $r = 1.0$ ,  $q_1 = 0.5$  and  $q_2 = 1.0$ . Initial conditions are  $N = 50 P_0$  and  $M = 20 R_0$  components.**

(writing  $P_0(t)$  for  $v_P, P_0(t)$  as before):

$$\begin{aligned}\dot{P}_0(t) &= -\min(P_0(t) \cdot r, R_0(t) \cdot N \cdot r) + P_1(t) \cdot q_1 \\ \dot{P}_1(t) &= -P_1(t) \cdot q_1 + \min(P_0(t) \cdot r, R_0(t) \cdot N \cdot r) \\ \dot{R}_0(t) &= -\min(P_0(t) \cdot r, R_0(t) \cdot N \cdot r) + R_1(t) \cdot q_2 \\ \dot{R}_1(t) &= -R_1(t) \cdot q_2 + \min(P_0(t) \cdot r, R_0(t) \cdot N \cdot r)\end{aligned}$$

We notice an immediate improvement of the situation under this new passive fluid semantics. The ODEs are Lipschitz continuous and thus we are guaranteed a globally unique solution as was shown not necessarily to be the case with the existing passive fluid semantics. Figure 3 shows this solution for the interesting (blocking) parameters used to generate Figure 2b. On comparing these two figures, we see that a marked improvement under the new semantics is evident, not least is the fact that the fluid analysis is meaningful after the critical point when the  $R_0$  bottleneck takes effect.

As we have mentioned, there is not just one way of converting a model exhibiting passive cooperation to one with purely active cooperation. Indeed, Theorem 1 merely gives a lower bound required to maintain equivalence to the original model. It is then a natural question to ask what happens if higher rates are used. One immediate consequence not directly related to the PEPA model is that the ODEs will become numerically stiffer and thus more costly to integrate. Figure 4 shows that increasing the effective passive rate appears to have little effect at least for the example of Equation (7), but does slightly worsen the correspondence of the ODE results with those obtained through stochastic simulation. Indeed, ongoing work beyond the scope of this paper suggests a theoretical justification for picking the lowest possible effectively passive rate.

### 3. WORKED EXAMPLE

We now apply the techniques previously introduced to a larger, more realistic model of a real-world system: a client-server scenario with a two-stage fetch protocol and the potential for server breakdowns. As a grouped PEPA model, this system can be expressed as follows:

$$\begin{aligned}Client &\stackrel{\text{def}}{=} (request, r_{req}).Client\_waiting \\ Client\_waiting &\stackrel{\text{def}}{=} (data, \top).Client\_think \\ Client\_think &\stackrel{\text{def}}{=} (think, r_{think}).Client \\ \\ Server &\stackrel{\text{def}}{=} (request, \top).Server\_get \\ &\quad + (break, r_{break}).Server\_broken \\ Server\_get &\stackrel{\text{def}}{=} (data, r_{data}).Server \\ &\quad + (break, r_{break}).Server\_broken \\ Server\_broken &\stackrel{\text{def}}{=} (reset, r_{reset}).Server \\ \\ Sys &\stackrel{\text{def}}{=} Clients\{Client[N_C]\} \\ &\quad \boxtimes_L Servers\{Server[N_S]\}\end{aligned}\quad (11)$$

where  $L = \{request, data\}$ . We have a population of  $N_C$  clients and a population of  $N_S$  servers. The system uses a 2-stage fetch mechanism: a client requests data from the pool of servers; one of the servers receives the request, another server may then fetch the data for the client. At any stage, a server in the pool may fail and later reset itself.

Writing  $C(t)$  for  $v_{Clients}, Client(t)$  and similarly  $S_g(t)$  for  $v_{Servers}, Server\_get(t)$ , and so on, we can derive two different ODE systems for this model, one which uses the original fluid passive semantics:

$$\begin{aligned}\dot{C}(t) &= -\mathbf{I}(S(t)) \cdot C(t) \cdot r_{req} + C_t(t) \cdot r_{think} \\ \dot{C}_w(t) &= -\mathbf{I}(C(t)) \cdot S_g(t) \cdot r_{data} + \mathbf{I}(S(t)) \cdot C(t) \cdot r_{req} \\ \dot{C}_t(t) &= -C_t(t) \cdot r_{think} + \mathbf{I}(C(t)) \cdot S_g(t) \cdot r_{data} \\ \dot{S}(t) &= -\mathbf{I}(S(t)) \cdot C(t) \cdot r_{req} - S(t) \cdot r_{break} \\ &\quad + \mathbf{I}(C(t)) \cdot S_g(t) \cdot r_{data} + S_b(t) \cdot r_{reset} \\ \dot{S}_g(t) &= -\mathbf{I}(C(t)) \cdot S_g(t) \cdot r_{data} - S_g(t) \cdot r_{break} \\ &\quad + \mathbf{I}(S(t)) \cdot C(t) \cdot r_{req} \\ \dot{S}_b(t) &= -S_b(t) \cdot r_{reset} + S_g(t) \cdot r_{break} + S(t) \cdot r_{break}\end{aligned}\quad (12)$$



and one which uses the new semantics as proposed in this paper:

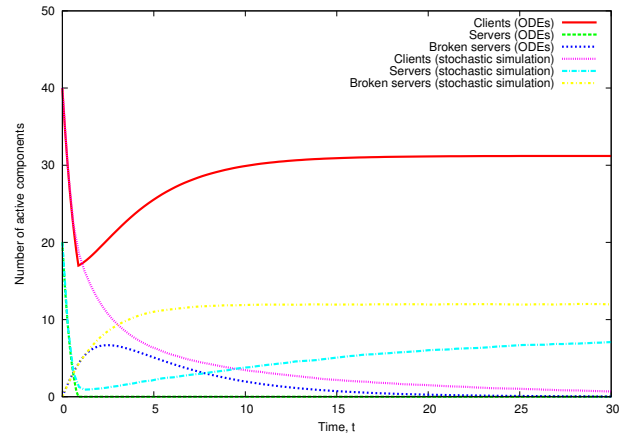
$$\begin{aligned}
 \dot{C}(t) &= -\min(S(t) \cdot N_C \cdot r_{req}, C(t) \cdot r_{req}) + C_t(t) \cdot r_{think} \\
 \dot{C}_w(t) &= -\min(C(t) \cdot N_S \cdot r_{data}, S_g(t) \cdot r_{data}) \\
 &\quad + \min(S(t) \cdot N_C \cdot r_{req}, C(t) \cdot r_{req}) \\
 \dot{C}_t(t) &= -C_t(t) \cdot r_{think} \\
 &\quad + \min(C(t) \cdot N_S \cdot r_{data}, S_g(t) \cdot r_{data}) \\
 \dot{S}(t) &= -\min(S(t) \cdot N_C \cdot r_{req}, C(t) \cdot r_{req}) - S(t) \cdot r_{break} \\
 &\quad + \min(C(t) \cdot N_S \cdot r_{data}, S_g(t) \cdot r_{data}) \\
 &\quad + S_b(t) \cdot r_{reset} \\
 \dot{S}_g(t) &= -\min(C(t) \cdot N_S \cdot r_{data}, S_g(t) \cdot r_{data}) \\
 &\quad - S_g(t) \cdot r_{break} \\
 &\quad + \min(S(t) \cdot N_C \cdot r_{req}, C(t) \cdot r_{req}) \\
 \dot{S}_b(t) &= -S_b(t) \cdot r_{reset} + S_g(t) \cdot r_{break} + S(t) \cdot r_{break} \quad (13)
 \end{aligned}$$

Figure 5a shows the result of using the old fluid passive semantics. Again we have artificially continued the solution beyond the point at which the number of servers reaches zero and there formally does not exist a (meaningful) solution to the ODEs of Equation (12). We see that as expected, this delivers very misleading results. Indeed, after the point of server exhaustion, the behaviour effectively switches to become that which we might expect if the definitions of *Server\_get* and *Server\_broken* in Equation (11) were to be modified to become absorbing. That is, servers are no longer replenished after having been used, so no clients are ever served again. This is of course a very misleading rendition of the client/server bottleneck scenario the original model is supposed to represent.

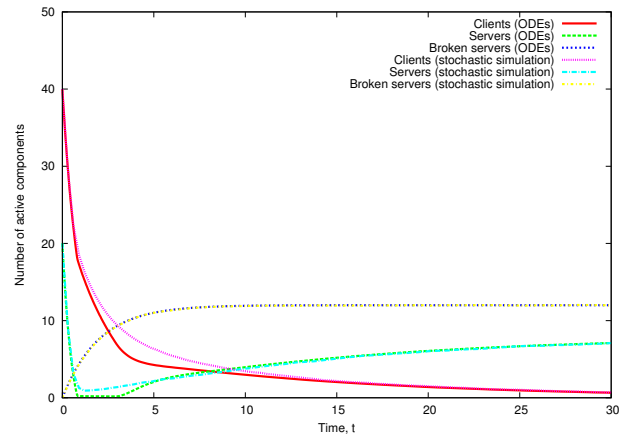
Under the new passive fluid semantics, we obtain Figure 5b by integrating the ODEs of Equation (13). The substantial improvement under this regime is very evident.

It is widely assumed in the case of active cooperation that the deterministic approximation  $v_{H,P}(t)$  defined by the ODEs is a low-variance approximation to the stochastic expectation of the corresponding  $n_{H,P}(t)$  stochastic process. That is, its correctness with respect to the underlying CTMC increases in some sense as the variance of the stochastic processes in question decreases. Indeed, work is ongoing to establish the exact relation. Since the new fluid passive semantics presented here are equivalent to an application of the existing fluid semantics for active cooperation, we presume their quantitative correctness is similarly related to variance.

It is important to be aware then that passive cooperation is more likely to lead to increased variance since when the expected number of passive partners is very low (a blocking scenario), a minor deviation from this expectation could mean zero passive partners. This is the difference between a fast rate of the shared action governed by the active partner to a rate of zero. This suggests that although the new fluid passive semantics presented here are a significant improvement, they may not perform as well as when analysing purely active models. Figure 6a shows the result of integrating the ODEs of Equation (13) for different parameters, designed to increase the variance in the counting stochastic processes. Thus, we have engineered a situation where the rate of the  $r_{req}$  shared action is very high when there are available servers up to the bottleneck point. As expected,



(a) Old ODE semantics.



(b) New ODE semantics.

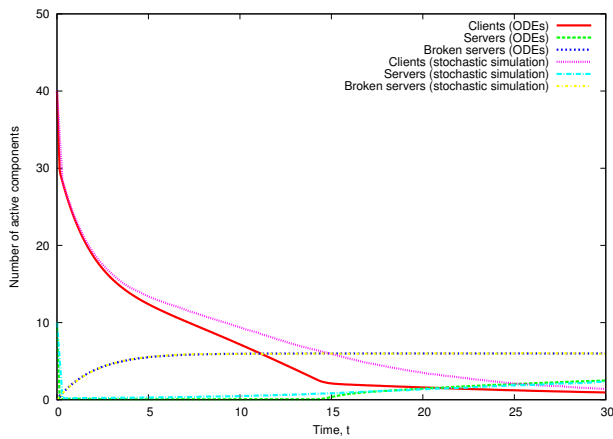
**Figure 5: ODE and stochastic simulation solution comparison for model of Equation (11). Rates used are  $r_{req} = 1.0$ ,  $r_{think} = 0.3$ ,  $r_{break} = 0.3$ ,  $r_{data} = 0.5$  and  $r_{reset} = 0.2$ . Initial conditions are  $N_C = 40$  Client and  $M = 20$  Server components.**

the correspondence is not as impressive as in Figure 5b, but again, exceedingly more useful than that obtained under the existing passive fluid semantics (Figure 6b).

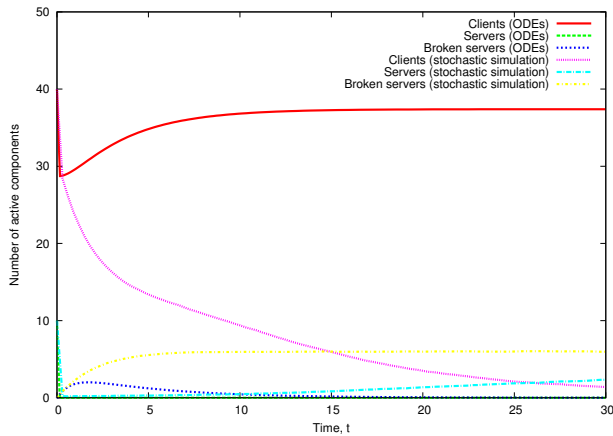
## 4. CONCLUSION

Passive cooperation is a key behaviour present in many real-world systems. At the same time, distributed, massively concurrent systems are becoming more prevalent and they automatically invoke the problem of state space explosion. As a result, these types of system inhibit traditional discrete state-space analysis. It is therefore a key requirement that emerging next-generation techniques of fluid-analysis for stochastic process algebras support accurate modelling of passive cooperation.

We have shown that the existing approach for handling such models in a fluid sense is often inaccurate and potentially very misleading in scenarios where blocking may occur (arguably the most interesting situations). Furthermore, the ODEs obtained through application of these semantics are not even guaranteed to have a meaningful global solution and it is necessary artificially (and without justification) to



(a) New ODE semantics.



(b) Old ODE semantics.

**Figure 6: ODE and stochastic simulation solution comparison for model of Equation (11). Rates used are  $r_{req} = 2.0$ ,  $r_{think} = 0.3$ ,  $r_{break} = 0.3$ ,  $r_{data} = 1.0$  and  $r_{reset} = 0.2$ . Initial conditions are  $N_C = 40$  Client and  $M = 10$  Server components.**

construct a continuation past where the formal solution is no longer defined.

However, we have also shown formally how to construct an equivalent model inexpensively which exhibits only active cooperation. From this model, the existing fluid semantics for active cooperation can be applied to yield fluid semantics for the original passive model. Such a fluid analysis has been shown empirically to yield a massive quantitative and qualitative improvement over the existing passive fluid semantics when compared with the expectation of the stochastic processes in question (the normal interpretation of the ODE approximation). The new semantics are also theoretically easier to handle since a meaningful global solution to the ODEs is guaranteed to exist.

Our plans for future work include investigating more formally which is the best of the infinite family of equivalent actively-synchronised models to choose for a given model involving passive cooperation. Furthermore, work is ongoing to address quantitatively the exact relation of the ODEs (both for active and thus also, under the new semantics, for passive synchronisation) to the underlying CTMC.

## 5. REFERENCES

- [1] H. Bowman, J. W. Bryans, and J. Derrick. Analysis of a multimedia stream using stochastic process algebras. *The Computer Journal*, 44(4):230–245, 2001.
- [2] J. T. Bradley, N. J. Dingle, S. T. Gilmore, and W. J. Knottenbelt. Derivation of passage-time densities in PEPA models using ipc: the Imperial PEPA Compiler. In G. Kotsis, editor, *MASCOTS'03, Proceedings of the 11th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunications Systems*, pages 344–351. IEEE Press, October 2003.
- [3] J. T. Bradley, S. T. Gilmore, and J. Hillston. Analysing distributed internet worm attacks using continuous state-space approximation of process algebra models. *Journal of Computer and System Sciences*, 74:1013–1032, July 2008.
- [4] M. Bravetti, S. Gilmore, C. Guidi, and M. Tribastone. Replicating web services for scalability. In G. Barthe and C. Fournet, editors, *Proceedings of the Third International Conference on Trustworthy Global Computing (TGC'07)*, volume 4912 of *LNCS*, pages 204–221. Springer-Verlag, 2008.
- [5] A. Duguid. Coping with the parallelism of BitTorrent: Conversion of PEPA to ODEs in dealing with state space explosion. In E. Asarin and P. Bouyer, editors, *Formal Modeling and Analysis of Timed Systems, 4th International Conference, FORMATS 2006, Paris, France, September 25-27, 2006, Proceedings*, volume 4202 of *LNCS*, pages 156–170. Springer, 2006.
- [6] J. M. Fourneau, L. Kloul, and F. Valois. Performance modelling of hierarchical cellular networks using PEPA. *Perf. Eval.*, 50(2–3):83–99, November 2002.
- [7] S. Gilmore and M. Tribastone. Evaluating the scalability of a web service-based distributed e-learning and course management system. In M. Bravetti, M. T. Núñez, and G. Zavattaro, editors, *Third International Workshop on Web Services and Formal Methods (WS-FM'06)*, volume 4184 of *LNCS*, pages 156–170, Vienna, Austria, 2006. Springer.
- [8] J. Hillston. *A Compositional Approach to Performance Modelling*, volume 12 of *Distinguished Dissertations in Computer Science*. Cambridge University Press, 1996.
- [9] J. Hillston. Fluid flow approximation of PEPA models. In *QEST'05, Proceedings of the 2nd International Conference on Quantitative Evaluation of Systems*, pages 33–42, Torino, September 2005. IEEE Press.
- [10] D. R. W. Holton. A PEPA specification of an industrial production cell. In S. Gilmore and J. Hillston, editors, *Process Algebra and Performance Modelling Workshop*, volume 38(7) of *Special Issue: The Computer Journal*, pages 542–551. CEPIS, Edinburgh, June 1995.
- [11] N. Thomas, J. T. Bradley, and W. J. Knottenbelt. Stochastic analysis of scheduling strategies in a GRID-based resource model. *IEE Software Engineering*, 151(5):232–239, September 2004.