

A Combined Process Algebraic, Agent and Fluid Flow Approach to Emergent Crowd Behaviour

Mieke Massink Diego Latella Andrea Bracciali* Jane Hillston
Istituto di Scienza e Tecnologie dell'Informazione 'A. Faedo', CNR School of Informatics
Area della Ricerca, Via Moruzzi 1 University of Edinburgh
Pisa, Italy Edinburgh, U.K.
{massink,latella}@isti.cnr.it Jane.Hillston@ed.ac.uk

Emergent phenomena occur due to the pattern of non-linear and distributed local interactions between the elements of a system over time. Surprisingly, agent based crowd models in which the movement of each individual follows a limited set of simple rules often re-produce quite closely the emergent behaviour of crowds that can be observed in reality. An example of such phenomena is the spontaneous self-organisation of drinking parties in the squares of cities in Spain, also known as “El Botellón” [22]. We revisit this case study providing an elegant stochastic process algebraic model in Bio-PEPA amenable to several forms of analyses among which simulation and fluid flow analysis. We show that a fluid flow approximation, i.e. a deterministic reading of the average behaviour of the system, can provide an alternative and efficient way to study the same emergent behaviour as that explored in [22] where simulation was used instead. Besides empirical evidence also an analytical justification is provided for the good correspondence found between simulation results and the fluid flow approximation. Scalability features of the fluid flow approach may make it particularly useful when studying models of more complex city topologies with very large populations.

1 Introduction

In modern society the formation of crowds, intended as large concentrations of people, is a phenomenon that occurs frequently. Well known examples are crowds at large entertainment events in cities or other open-air facilities such as sport stadiums, but also crowds at large airports and train stations. Fortunately, such crowds usually occur and dissolve without serious problems. However, in some cases accidents happen with possibly major consequences such as loss of lives and a large number of injuries [24].

There is an ever stronger interest in being able to prevent such disasters and there exists an extensive literature on numerous approaches to the study of crowd formation, crowd management and emergency egress [23]. Simulation models play an important role in these approaches. In particular, agent based modelling has become popular in recent years because it may provide valuable information about the dynamics of systems that contain non-linear elements, chaos and random cause and effect. Several works in this area, e.g. work by Still [24], show that a crowd of people in which each individual follows a limited number of simple rules produces quite closely the emergent behaviour that can be observed in real human crowds. Emergent phenomena are known to occur due to the pattern of non-linear and distributed local interactions between the large number of elements of a system over time. In particular, work by Still shows that such simple local rules are in many cases sufficient to produce the observed behaviour and that no complicated rules of group behaviour or psychological parameters are necessary. His work and that of others have led to the development of several professional tools, based on agent simulation, for the realistic analysis and prediction of crowd behaviour in emergency situations. Such

*With University of Stirling from August 1, 2010.

analyses may help to detect architectural or organisational problems that may potentially cause loss of lives in the event of emergency situations occurring in, for example, airports, sport stadiums and open-air festivals.

However, for very large crowds, analysis via simulation may become time-consuming since each execution of the model produces only a single trajectory through the state space whereas many executions are needed to reach statistically relevant conclusions. Therefore, the costs of simulation based analysis often becomes prohibitive in situations in which a quick analysis is required to compare the consequences of different design options or when a large number of slightly different scenarios need to be analysed.

In a completely different field of research, namely that of the analysis of bio-chemical reactions, it has been shown that under certain conditions, such as the presence of a sufficiently large population, a deterministic continuous interpretation of models composed of many similar small independent components provide a good approximation of the average behaviour of the overall model. In the context of stochastic process algebras this insight has led to the development of an alternative formal fluid flow semantics for PEPA first and later for Bio-PEPA, a variant of PEPA originally devised for modelling biochemical processes, based on the generation of sets of ordinary differential equations (ODE) [16, 10]. An application of PEPA with this alternative semantics in the context of emergency egress [20, 5] has shown very good correspondence with results from the literature on evacuation times and node profiles, i.e. the average number of people present in a particular part of the building over time during egress. In that work only linear differential equations have been considered.

In this paper we revisit the case of self-organisation of crowds in a city as described by Rowe and Gomez in [22]. This case differs from that of emergency egress mentioned before because of the presence of non-linear aspects where the behaviour of the individual agents depends directly on other, similar agents present in the same environment. We show that with Bio-PEPA a fluid flow approximation can provide an alternative and computationally efficient way to study the same emergent behaviour as that explored in [22] where simulation was used instead. In that work the movement of a crowd in a city is studied under various assumptions about the likelihood that people remain in a square. The work was inspired by a typical social phenomenon observed in Spanish cities, on summer nights, called “El Botellón”, when crowds of youngsters wander between city squares in search of a party. Such self-organised parties sometimes lead to heavy drinking and noisy behaviour until late at night. It turned out to be hard to predict when and where a large party would take place. The aim of the work by Rowe and Gomez was to gain insight into the conditions under which parties self-organise. In their study agents follow two basic rules. The first rule defines when agents remain in a square, which depends on the “chat-probability”, i.e. the likelihood to meet someone in the square to chat with. The second rule defines how agents move between squares.

Rowe and Gomez develop an analytical model to approximate the threshold of this chat-probability below which people are freely moving through the city and above which large crowds start to form. They validated their theory by the simulation of a multi-agent model for a ring topology of 4 squares and up to 80 agents. Both the theory and the simulation results show that for a value of the chat probability $c = n/N$, where n is the number of squares and N the number of agents, a clear phase-transition can be observed between a steady-state situation in which agents are evenly distributed over the squares (when c is below the threshold) and a situation in which agents spontaneously gather in one or a few squares (when c is above the threshold).

In this paper, we approach the modelling of crowds by adopting the Bio-PEPA stochastic process algebra [11]. Bio-PEPA embeds a notion of spatial location, intended to model compartments, suitable to describe the city topology and locate agents within it. Moreover, the agent behaviour can be expressed as a function of the current state of the system, such as the number of people present in a square, an

abstraction of the act of sensing the environment, common to standard agent models. Such a function may contain non-linear elements, which makes Bio-PEPA also particularly interesting for the analysis of some forms of emergent behaviour, as we will see in later sections. The fluid flow results obtained with the Bio-PEPA model correspond surprisingly well to the simulation results obtained with the the same model and to those published by Rowe and Gomez [22]. Informally speaking, for models where the rates can be expressed as functions of the average density of the population this phenomenon is well known (assuming that the populations are sufficiently large), see e.g. Kurtz [18] and in the context of mean field analysis Le Boudec et al. [2]. However, the rate functions in the crowds model addressed in this paper cannot be expressed this way. We provide an alternative analytical explanation for the observed correspondence which is partially based on recent work by Hayden and Bradley [14] on PEPA.

By adopting this modelling and analysis approach, we gain an expressive linguistic description of agent behaviour from process algebras, which can be thought of modularly, can be easily revised, is grounded on a neat formal semantics also encompassing the stochastic aspects, and is supported by a suite of computational tools. Fluid flow approximation aims at tackling the state space combinatorial explosion that arises as a consequence of interleaving the behaviour of the many independent individuals. Fluid flow relies on an abstract quantitative description of the system, i.e. variables represent amounts of agents with independent but “homogeneous” behaviour, and on the approximating hypothesis that such quantities vary continuously over time.

Furthermore, with this approach, the agent view and the stochastic modelling are retained within a single framework, allowing the analysis of averaged and non-averaged behaviour to be carried out using the same model. For instance one may wish to understand and even tune the average behaviour of the system and then deepen some particular cases of interest. This may be particularly advantageous whenever the computational costs of stochastic simulation may only be justified when a final system design is considered, but would be prohibitive when exploring many design options and conditions during early phases of design. The analysis of the threshold for the self-organisation of crowds in a city is an example of such explorative analysis that would require many simulations to obtain an accurate view of the phase-transition. This is even more so when more complex city-topologies and a large number of agents are considered. A preliminary version of this work has been presented at the PASTA 2010 workshop [21].

The paper is organised as follows. Section 2 recalls the crowd model used in the case study by Rowe and Gomez [22]. Section 3 gives a brief overview of Bio-PEPA and its analysis environment. Section 4 describes the Bio-PEPA model of the collective behaviour of crowds in a city. Sections 5 and 6 provide both simulation and fluid flow results and compares them with the original results in [22]. Section 7 illustrates an extension of the model in which squares have different attractiveness. Finally, in Section 9 conclusions are presented and an outline of future research is given. Annex A presents the full specification of a Bio-PEPA model for a topology of four squares. Annex B illustrates the derivation of the set of ODEs associated with the model of Annex A. Annex C presents an analytical assessment of the formal relationship between the CTMC based formal semantics of the model and its fluid flow approximation.

2 Rowe and Gomez Model of Crowd Dynamics

In this section we briefly recall the model of movement of crowds between squares in a city as presented by Rowe and Gomez in [22]. Assume a city with n squares represented as a graph with vertices $\{1, 2, \dots, n\}$. People are simulated by “agents” that are following a simple set of rules. The number of

agents in square i , with $i \in \{0, 1, \dots, n\}$, at time t , with t representing discrete time steps, is represented by $p_i(t)$. The state of the system at t is given by the number of agents present in each square modelled by the vector $\mathbf{p}(t) = (p_1(t), p_2(t), \dots, p_n(t))$. The total number of agents N at any time t is constant: $N = \sum_{i=1}^n p_i(t)$.

Agents are located in squares. The rules guiding agents' behaviour are the following. The probability that an agent decides to remain in a square depends on how many other agents are present in the same square. If a square i contains $p_i > 0$ agents, the probability that an agent *leaves* the square is given by $(1 - c)^{p_i - 1}$. The parameter c (representing the *chat probability*, $0 \leq c \leq 1$) is the probability that an agent finds another one to talk to and thus remains in the square. Consequently, if the population of square i is p_i then the probability for an agent to find nobody to talk to is $(1 - c)^{p_i - 1}$. Note that when there is only one agent in the square, it decides to leave with probability 1, since there is nobody else to talk to. If an agent decides to move, it moves with equal probability to any neighbouring square reachable by a street. Considering an analytical model of the above discrete behaviour the *expected* number of agents that will leave square i at a given time step t is given by the function:

$$f_i(t) = p_i(t)(1 - c)^{p_i(t) - 1}$$

This models the fraction of the population in square i that does not find anyone to talk to in that square¹. The probability that an agent, which decided to leave square j , moves to the adjacent square i is given by the matrix A_{ij} :

$$A_{ij} = \text{con}_{ij} / d_j$$

where d_j is the degree of vertex j , i.e. the number of streets departing from square j , and con_{ij} denotes that square i is connected to square j :

$$\text{con}_{ij} = \begin{cases} 1 & \text{if } i \text{ is connected to } j \\ 0 & \text{otherwise} \end{cases}$$

Clearly, $\text{con}_{ij} = \text{con}_{ji}$ and we assume that adjacent squares are connected by at most one street. The expected distribution of agents over squares at time $t + 1$ can now be defined as:

$$\mathbf{p}(t + 1) = \mathbf{p}(t) - \mathbf{f}(t) + \mathbf{A}\mathbf{f}(t)$$

Clearly, from this formula it follows that a steady-state behaviour is reached when $\mathbf{f}(t) = \mathbf{A}\mathbf{f}(t)$. In other words, when the number of people entering a square is equal to the number leaving the square. Rowe and Gomez show that there are two possibilities for such a stable state. In one case the agents freely move between squares and their distribution is proportional to the number of streets connected to each square. In the second case agents gather in large groups in a small number of squares corresponding to emergent self-organisation of parties. Which of the two situations will occur depends critically on the value of the chat probability c . In case all squares have the same number of neighbouring squares a phase shift occurs at about $c = n/N$ where n is the number of squares and N the number of agents. For $c < n/N$ people freely move between squares whereas for $c > n/N$ agents self-organise into large groups. Simulation of the model confirms in an empirical way that this estimate for c is quite accurate when the population is large enough where large means about 60 agents or more in a 4-square topology.

¹Note that in this analytical model the number of agents $p_i(t)$ in square i is now approximated by a real number: the *expected* number of agents in square i at time t .

For topologies where each square has the same number of streets the critical value of c can be estimated in an analytical way. For less regular topologies and in case different squares have different chat probabilities and not all directions leaving from a square are equally likely to be taken by people it is very difficult to identify such critical values in an analytical way. Usually, in such cases simulation is used to analyse the models. However, when a large number of agents is involved, simulation may be extremely time consuming. In Section 4 we show that a combined process algebraic agent modelling and fluid flow approximation based on ODEs may provide a much faster way to obtain similar information for this class of models.

3 Bio-PEPA and Fluid Flow Analysis

In this section we give a short description of Bio-PEPA [10, 11, 9], a language that has recently been developed for the modelling and analysis of biochemical systems. The main components of a Bio-PEPA system are the “*species*” components, describing the behaviour of individual entities, and the *model component*, describing the interactions between the various species. The initial amounts of each type of entity or species are given in the model component.

The syntax of the Bio-PEPA components is defined as:

$$S ::= (\alpha, \kappa) \text{ op } S \mid S + S \mid C \quad \text{with op} = \downarrow \mid \uparrow \mid \oplus \mid \ominus \mid \odot \quad P ::= P \underset{\mathcal{L}}{\boxtimes} P \mid S(x)$$

where S is a *species component* and P is a *model component*. In the prefix term $(\alpha, \kappa) \text{ op } S$, κ is the *stoichiometry coefficient* of species S in action α . This arises from the original formulation of the process algebra for modelling biochemical reactions, where the stoichiometric coefficient captures how many molecules of a species are required for a reaction. However it may be interpreted more generally as the multiples of an entity involved in an occurring action. The default value of κ is 1 in which case we simply write α instead of (α, κ) . The *prefix combinator* “op” represents the role of S in the action, or conversely the impact that the action has on the species. Specifically, \downarrow indicates a *reactant* which will be consumed in the action, \uparrow a *product* which is produced as a result of the action, \oplus an *activator*, \ominus an *inhibitor* and \odot a generic *modifier*, all of which play a role in an action without being produced or consumed and have a defined meaning in the bio-chemical context. The operator “+” expresses the choice between possible actions, and the constant C is defined by an equation $C=S$. The process $P \underset{\mathcal{L}}{\boxtimes} Q$ denotes synchronisation between components P and Q , the set \mathcal{L} determines those actions on which the the components P and Q are forced to synchronise, with \boxtimes denoting a synchronisation on all common action types. In the model component $S(x)$, the parameter $x \in \mathbb{R}$ represents the initial amount of the species.

A Bio-PEPA *system* with *locations* consists of a set of species components, also called sequential processes, a model component, and a context (locations, kinetics rates, parameters, etc.). The prefix term $(\alpha, \kappa) \text{ op } S@l$ is used to specify that the action is performed by S in location l . The notation $\alpha[I \rightarrow J] \odot S$ is a shorthand for the pair of reactions $(\alpha, 1)\downarrow SI$ and $(\alpha, 1)\uparrow SJ$ that synchronise on action α and where SI denotes the population of agents in location I and SJ that in J . This shorthand is very convenient when modelling agents migrating from one location to another as we will see in the next section. Bio-PEPA is given an operational semantics [11]. In that context species amounts are abstracted by discrete levels, representing intervals of values. There are two relations over the processes: the *capability relation*, which supports the derivation of qualitative information, and the *stochastic relation*, defined in terms of the capability relation and equipped with rates for the associated action types. There is a rate function r_α associated with each action type α and its value is calculated in each state according to the current state of the system and its context. In modelling biochemistry this rate function is often the so-called *mass*

action law which stipulates that the rate of the reaction is the product of the amounts of its reactants and a rate constant. At each time instant each action is assumed to be governed by an exponentially distributed random variable which determines its duration: the rate function gives the parameter of this distribution. This gives rise to an underlying continuous time Markov chain (CTMC). The dynamic behaviour of processes is determined by a *race condition*: all actions which are enabled attempt to proceed but only the fastest succeeds.

In the models in this paper the rate function is not the mass-action law but is a functional rate reflecting the rate with which people leave a square. As in the original model by Rowe and Gomez, the functional rate depends on the number of people present in a square and the chat-probability, leading to a non-linear rate function. However, also in this case this function provides the parameter of an exponential distribution leading to an underlying CTMC.

The Bio-PEPA language is supported by a suite of software tools which automatically process Bio-PEPA models and generate internal representations suitable for different types of analysis [11, 8, 3]. These tools include mappings from Bio-PEPA to differential equations (supporting fluid flow approximation), stochastic simulation models [13], CTMCs with levels [10] and PRISM models [19].

3.1 Fluid Flow

As mentioned above, the Bio-PEPA semantics allows for the application of different analysis and evaluation techniques including fluid flow analysis originally defined in terms of the related stochastic process algebra PEPA. We give a very brief summary of the approach here; for details see [16, 6, 10]. The method to derive a set of ordinary differential equations for a Bio-PEPA model is illustrated in Appendix B for the crowd model that will be presented in the next section.

A Bio-PEPA model consists of a number of sequential components each of which represents a number of entities in a distinct state. The result of an action is to increase the number of some entities and decrease the number of others, these adjustments reflecting the stoichiometry with respect to the action. Thus we can represent the total state of the system at any time as a vector with entries capturing the counts of each species component. This gives rise to a discrete state system which undergoes discrete events. The idea of fluid flow analysis is to approximate these discrete jumps by continuous flows between the states of the system. This approximation becomes good when entities are present in such high numbers as to make the frequency of actions high and the relative change from each single event small. In this case we can derive a set of ordinary differential equations (ODEs) which approximate the average behaviour of the CTMC.

4 Modelling Crowd Movement with Bio-PEPA

Let us consider the same small city topology with 4 squares, as in Rowe and Gomez and as shown in Fig. 1, allowing bi-directional movement between squares. The first five lines of the Bio-PEPA specification below define this topology. The first line defines the default compartment *top* that contains all other compartments. The next four lines define square *A* to *D*. In this context *size* is used to denote a capacity in terms of number of agents. The size of the squares, defined by parameter *normal_square* = 100, is defined in such a way that all agents, 60 in this case, would fit in any single square and does not impose any further constraints.

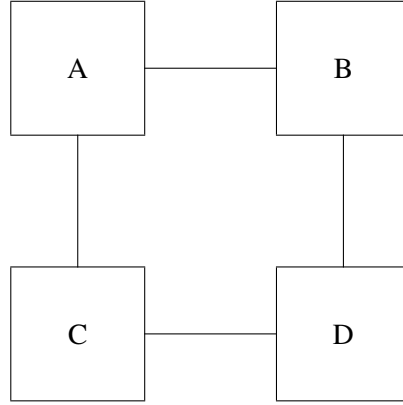


Figure 1: City plan with four squares

location top : *size = 1000, type = compartment;*
location sqA in top : *size = normal_square, type = compartment;*
location sqB in top : *size = normal_square, type = compartment;*
location sqC in top : *size = normal_square, type = compartment;*
location sqD in top : *size = normal_square, type = compartment;*

The model, which we will henceforth refer to as the ‘crowd model’, has two further parameters. The parameter c defines the chat-probability and the parameter d the degree or number of streets connected to a square. In the topology presented in Fig. 1 $d = 2$ for all squares.

$normal_square = 100;$
 $c = 0.05;$
 $d = 2;$

The activities modelling agents moving from square X to square Y will be denoted by f_{XtY} .

The associated functional rate (indicated by the keyword “kineticLawOf”) is defined in analogy to [22]. As explained in Section 2 there the setting is that of a discrete time model in which the expected number of agents that will leave square i at a given time step t is given by the function:

$$f_i(t) = p_i(t)(1 - c)^{p_i(t)-1}$$

Since the only information on the probability distribution available is the expected number of agents leaving a square per time unit, this same information can also be modelled as the rate parameter of an exponential distribution. In particular, letting $P@sqX$ model the number of people currently in square X, such a rate is:

$$P@sqX * (1 - c)^{(P@sqX-1)}$$

This models the rate of people leaving square X via any of its connecting streets leading to a neighbouring square.

If one also considers the uniform distribution of people over the outgoing streets of the square then the above rate needs to be divided by its degree d when agents leaving through a particular street are considered. In the four square ring topology the value of d is 2 for all squares. So the general rate with which agents leave square X via a particular street is:

$$(P@sqX * (1 - c)^{(P@sqX-1)})/d$$

This leads to the following functional rates for the crowd model, one for each direction of movement:

$$\begin{aligned} \text{kineticLawOf } fAtB & : (P@sqA * (1 - c)^{(P@sqA-1)})/d; \\ \text{kineticLawOf } fBtA & : (P@sqB * (1 - c)^{(P@sqB-1)})/d; \\ \text{kineticLawOf } fBtD & : (P@sqB * (1 - c)^{(P@sqB-1)})/d; \\ \text{kineticLawOf } fDtB & : (P@sqD * (1 - c)^{(P@sqD-1)})/d; \\ \text{kineticLawOf } fAtC & : (P@sqA * (1 - c)^{(P@sqA-1)})/d; \\ \text{kineticLawOf } fCtA & : (P@sqC * (1 - c)^{(P@sqC-1)})/d; \\ \text{kineticLawOf } fCtD & : (P@sqC * (1 - c)^{(P@sqC-1)})/d; \\ \text{kineticLawOf } fDtC & : (P@sqD * (1 - c)^{(P@sqD-1)})/d; \end{aligned}$$

Indeed, the actual rate of the action also depends on the number of people currently in a square X , besides the factor $(1 - c)^{(P@sqX-1)}$, with $c \in [0, 1]$. Note that this reduces the rate of outgoing agents the larger the population $P@sqX$ becomes.

The sequential component P below specifies the possible movements of a typical agent between squares. For example, $fAtB[sqA \rightarrow sqB] \odot P$ means that an agent present in square A moves to square B according to the functional rate defined for the action $fAtB$.

$$\begin{aligned} P = & fAtB[sqA \rightarrow sqB] \odot P + fBtA[sqB \rightarrow sqA] \odot P + \\ & fAtC[sqA \rightarrow sqC] \odot P + fCtA[sqC \rightarrow sqA] \odot P + \\ & fBtD[sqB \rightarrow sqD] \odot P + fDtB[sqD \rightarrow sqB] \odot P + \\ & fCtD[sqC \rightarrow sqD] \odot P + fDtC[sqD \rightarrow sqC] \odot P; \end{aligned}$$

Finally, the model component defines the initial conditions of an experiment, i.e. in which squares the agents are located initially, and the relative synchronisation pattern. Initially, there are 60 agents in square A . This is expressed by $P@sqA[60]$ in the composition shown below. All other squares are initially empty (i.e. $P@sqX[0]$ for $X \in \{B, C, D\}$). The fact that moving agents need to synchronise follows from the definition of the shorthand operator \rightarrow .

$$(P@sqA[60] \boxtimes_* P@sqB[0]) \boxtimes_* (P@sqC[0] \boxtimes_* P@sqD[0])$$

The total number of agents $P@sqA + P@sqB + P@sqC + P@sqD$ is invariant and amounts to 60 in this specific case.

5 Results for a Model with Four Squares

This section reports the analysis results for the model with four squares. The figures report both analysis via Gillespie stochastic simulation (G) [13], averaged over 10 independent runs, and fluid flow analy-

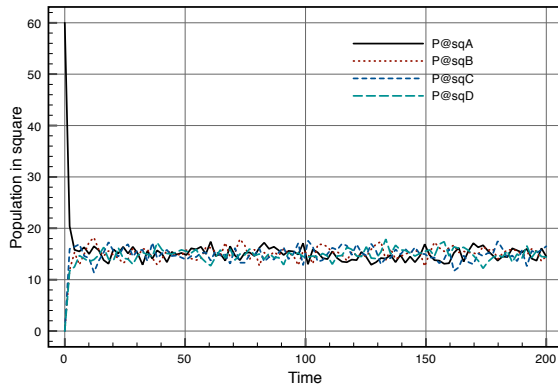
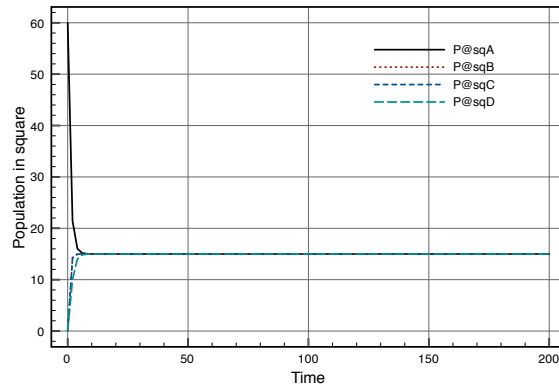
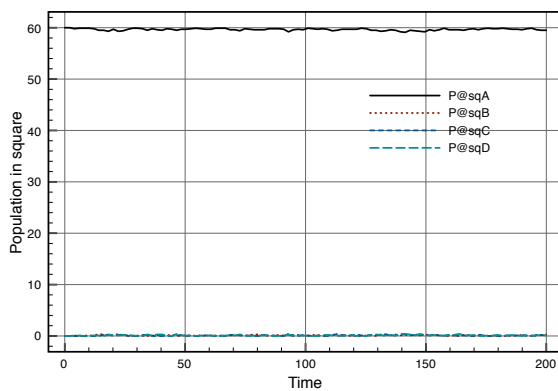
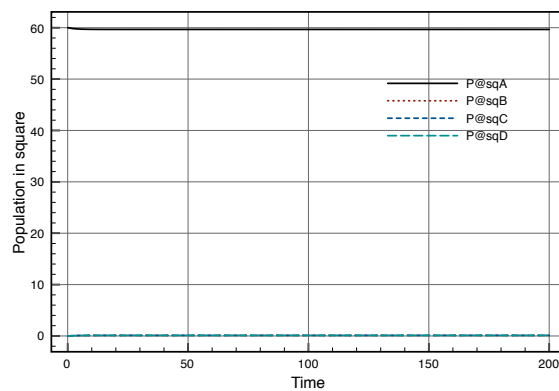
(a) Simulation results for $c=0.005$.(b) Fluid flow results for $c=0.005$.(c) Simulation results for $c=0.10$.(d) Fluid flow results for $c=0.10$.

Figure 2: Results for four squares with 60 agents in A initially

sis based on the adaptive numeric solution of sets of ODEs based on the adaptive step-size 5th order Dormand-Prince ODE solver [12].²

Figure 2(a) shows stochastic simulation results for a model with 60 agents initially in square A and for a value of $c = 0.005$ which is below the analytically estimated threshold of $c = n/N = 4/60 = 0.06666$. The results show that a dynamic equilibrium is reached, i.e. all agents distribute evenly over the four squares, which confirms the discrete event simulation results reported by Rowe and Gomez. Figure 2(c) shows the results of stochastic simulation for the same model, but for $c = 0.10$, a value above the threshold. The figure shows that the population settles rather quickly in a steady state in which almost all agents remain in square A. This is the second type of steady state observed also by Rowe and Gomez. Interestingly, the fluid flow analysis of the same model, for both values of c (Fig. 2(b) and Fig. 2(d)) show very good correspondence to the respective simulation results (Fig 2(a) resp. Fig. 2(c)).

Since these results show that both types of steady state emerge in this stochastic version of the crowd model and for both types of analysis, the question naturally arises whether fluid flow could be used as

²All analyses have been performed with the Bio-PEPA Eclipse Plug-in tool [8] on a Macintosh PowerPC G5.

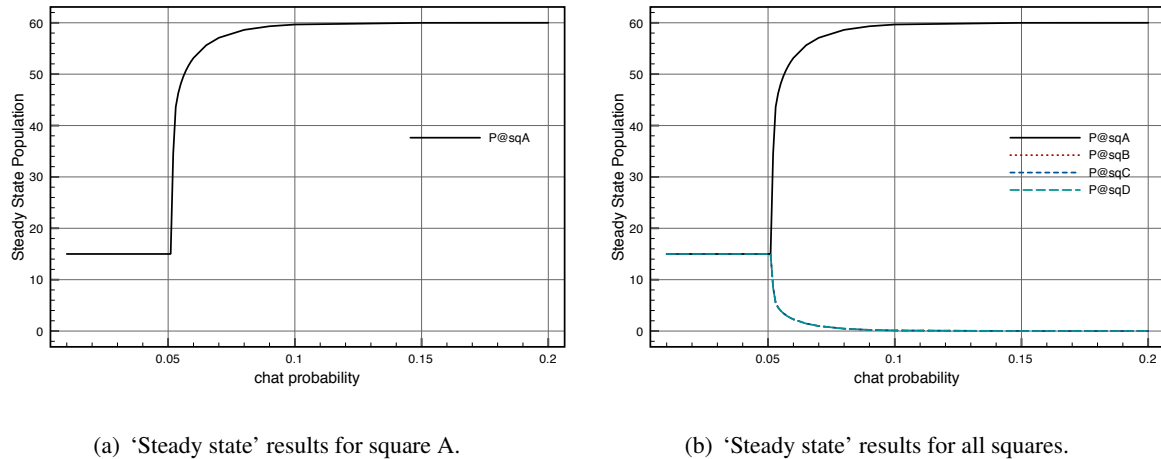


Figure 3: ‘Steady state’ results at $t=200$ and varying chat probabilities.

an efficient technique to investigate the behaviour of the model for various values of the chat-probability c , in particular those close to the critical threshold. To this purpose we have performed a fluid flow analysis for different chat probabilities ranging from 0.01 to 0.2 with steps of 0.01, except between 0.05 and 0.065 where the value is increased by steps of 0.001. Each fluid flow analysis takes less than a second independently of how many agents are considered in the model. If stochastic simulation would be used instead, the time for analysis would grow exponentially with the number of agents due to their interleaving behaviour. The results are shown in Fig. 3(a) for the number of people in square A on the long run (effectively at $t = 200$) starting with 60 people in square A initially. The figure shows clearly that for a chat probability below 0.05 in the steady state there are approximately 15 people in square A. In fact, in Fig. 3(b), where the steady state populations of all squares are shown, it is clear that for $c < 0.05$ the population is evenly distributed over the four squares.

For $c > 0.05$ the situation changes sharply. For these higher values of c the agent population tends to concentrate in square A, the square from which they started. All other squares remain essentially empty. A similar emergent behaviour is observed when the population is initially more evenly distributed over the squares, for example with 30 agents in A, and 10 each in B, C and D as shown in Figs. 4(a) and 4(b) for $c = 0.005$ and in Figs. 4(c) and 4(d) for $c = 0.10$. This shows that the grouping of agents in A also happens when starting from a configuration in which not all agents were already present in square A.

The results in Fig. 3 show a clear case of spontaneous *self-organisation* or *emergent* behaviour. In other words, it shows the phenomenon that a structure or pattern appears in a system without being imposed by a central authority or any external element. Note that the results have been obtained by a series of numerical solutions of the ODEs derived from the Bio-PEPA model (fluid flow analysis) for various values of c instead of via stochastic simulation. The results in Fig. 3(b) closely correspond to those obtained by simulation by Rowe and Gomez [22], apart from a re-normalisation factor for the size of the population. The ease and computational efficiency with which these accurate results can be produced by means of fluid flow analysis opens up a promising perspective on how process algebraic fluid flow analysis could be used as an alternative, efficient, scalable and formal approach to investigate emergent behaviour in the vicinity of critical parameter values for this class of models.

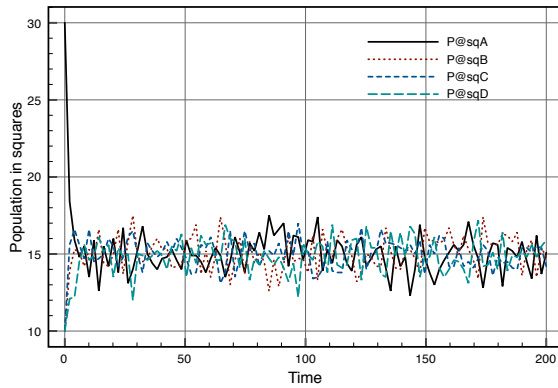
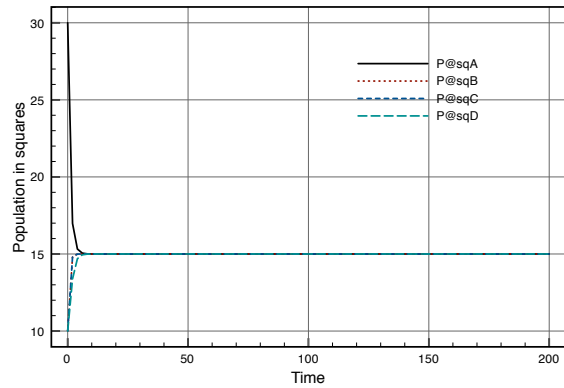
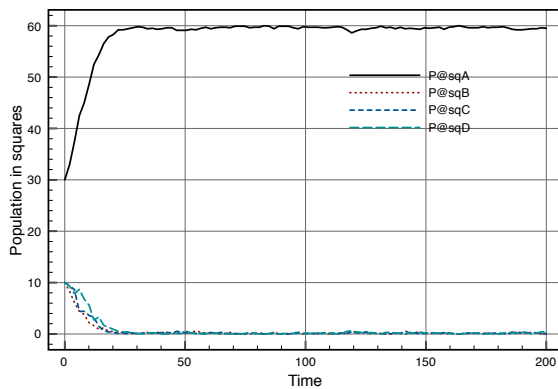
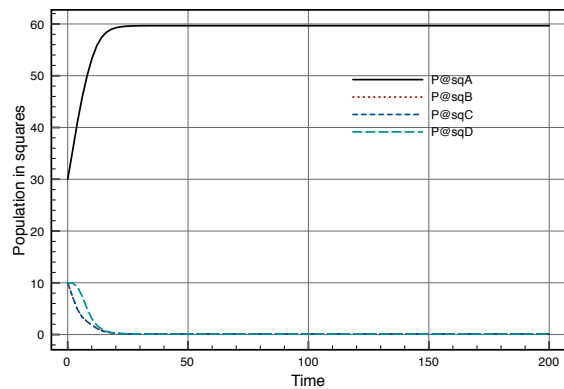
(a) Simulation results for $c=0.005$.(b) Fluid flow results for $c=0.005$.(c) Simulation results for $c=0.10$.(d) Fluid flow results for $c=0.10$.

Figure 4: Results for four squares with initially 30 agents in A, and 10 each in B, C and D.

An impression of how the distribution of agents over the four squares evolves for values of c that are close to the threshold of $c = 0.05$ is shown in Fig. 5. For $c = 0.05$ and $c = 0.051$ the agents still distribute uniformly over the four squares, though this takes a bit more time than for lower values of c . For $c = 0.052$ this situation is changing, and for $c = 0.053$ and higher clearly a different steady state is reached early on in which most agents group in the single square A. Note that for $c = 0.052$ a stable state has not yet been reached at time $t = 200$. However, this does not influence the overall picture. In general of course the idea is to collect data when a stable situation is reached.

Since the theory on the stability of fixed-points predicts that the steady state behaviour for values of $c > n/N = 4/60 = 0.066666$ is unstable (see [22]), we expect this to show up in the stochastic simulation and fluid approximation of our crowd model as well. Instability in this case means extreme sensitivity to the initial values of the number of agents in each square. If the agents are initially distributed perfectly evenly over the squares, any of the four squares is equally likely to be the square where agents will gather in the long run. This is illustrated by the results in Fig. 6 where single simulation runs are shown for the same model starting with 30 agents in each square initially and for $c = 0.1$. Fig. 7 shows the results for two experiments, comprising stochastic simulations of 100 runs each, with the same initial distribution of agents and with $c = 0.1$. In this case it will take longer before an average steady state is reached.

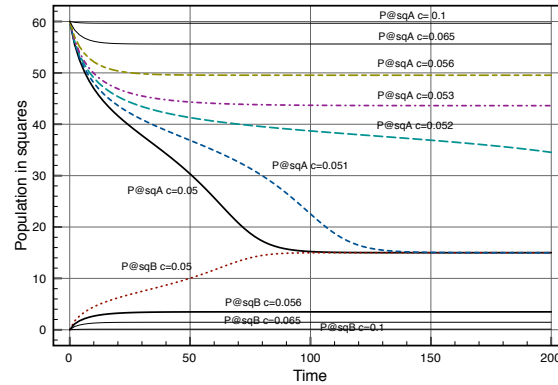
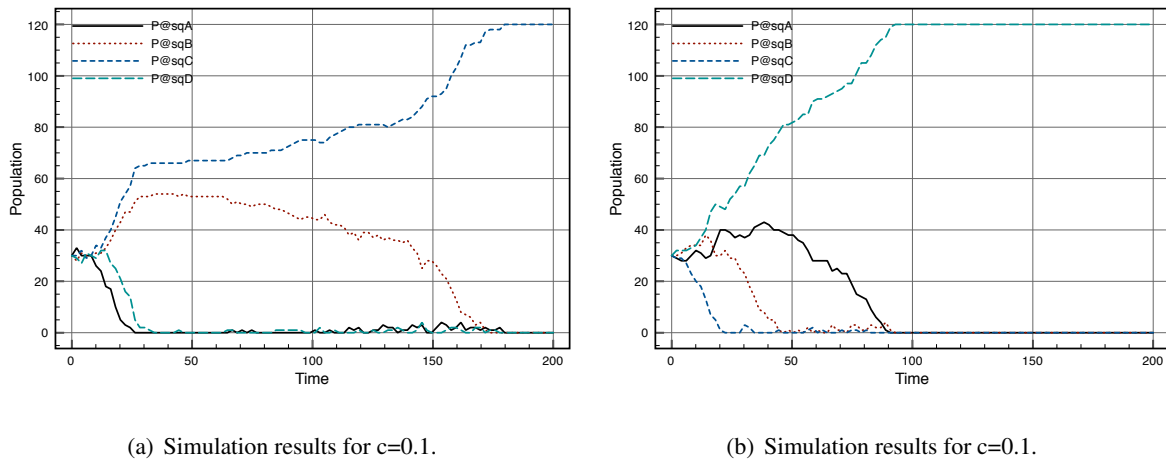


Figure 5: Fluid flow results for square A (and partially B) for varying chat probabilities around the critical value 0.05.



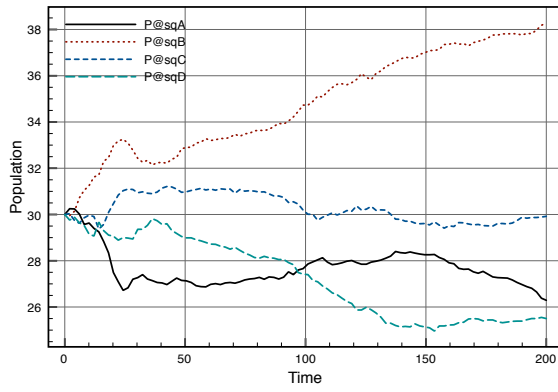
(a) Simulation results for $c=0.1$.

(b) Simulation results for $c=0.1$.

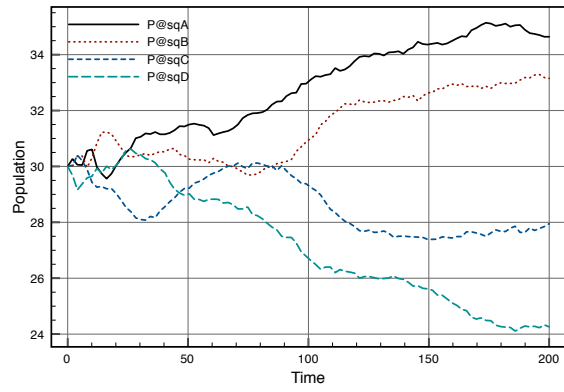
Figure 6: Two single simulation runs for the same model with initially 30 agents in each square

For a sufficiently high number of runs one would expect that in the steady state the graph would show that there are on average 30 agents in each square. Such an outcome in this case does *not* mean that for $c = 0.1$ on average an equal distribution of agents over squares is reached, but rather that, on average, every square has the same probability (0.25 in this case) to have *all* agents gathered in it in the long run. This is a subtle but very important point. This interpretation of the fluid flow results is confirmed by the simulation results in Fig. 8 which shows a convergence of the population in each square to 30 on average over 100,000 simulation runs. Note the different scales for the population in these figures with respect to those in Fig. 6 and Fig. 7. The fluid flow approximation of the model with initially 30 agents in each square is shown in Fig. 8(c).

Although so far we have considered models with a relatively low number of agents the numerical approximation algorithms for ODEs are essentially insensitive, in terms of efficiency, to the size of the population considered as long as these populations are sufficiently large to guarantee sufficient precision. We have used 60 agents in this study to be able to compare our results with those obtained by Rowe and Gomez.

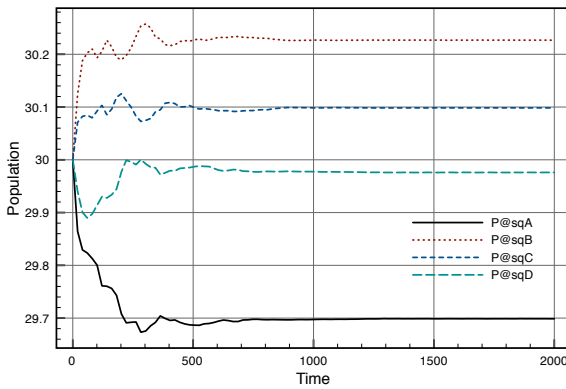


(a) Simulation results for $c=0.1$.

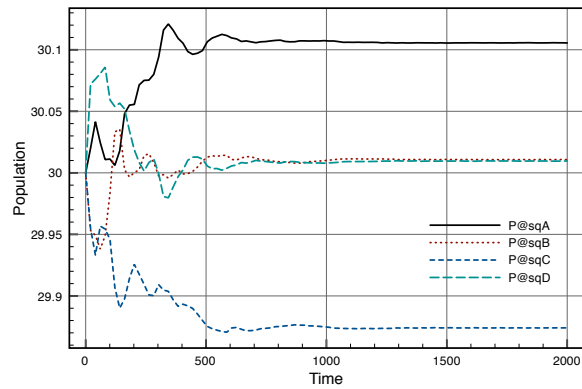


(b) Simulation results for $c=0.1$.

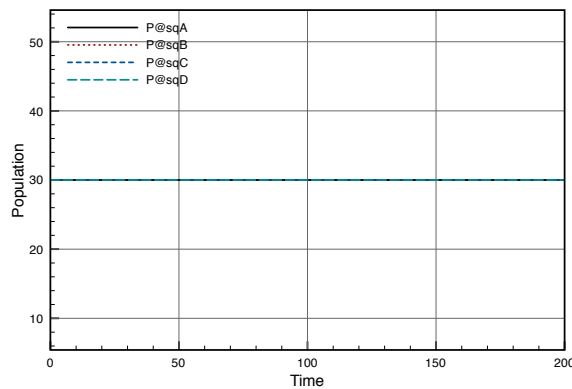
Figure 7: Two simulations of 100 runs each for the same model



(a) Simulation results for $c=0.1$.

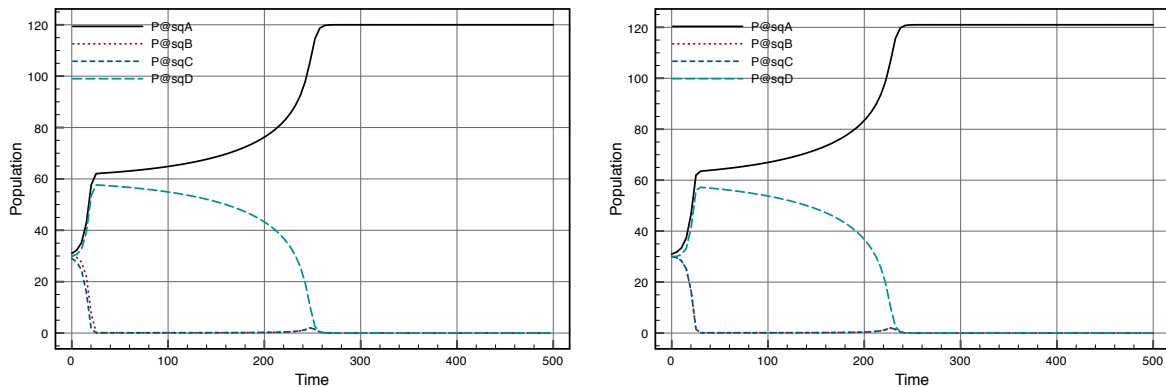


(b) Simulation results for $c=0.1$.



(c) Fluid flow results for $c=0.1$.

Figure 8: Two simulations of 100,000 runs each for the same model, and a fluid flow approximation



(a) Fluid flow results for A=31, B=30, C=29 and D=30

(b) Fluid flow results for A=31, B=30, C=30 and D=30

Figure 9: Fluid flow results for different initial population values for $c=0.1$

Figs. 9 (a), (b) and (c) show the results of a closer investigation of the effect of slight changes in the initial agent population values for chat-probability $c = 0.1$. The results seem to confirm that it is more likely that the agents gather in the square that has the highest initial population, even if its population is only marginally higher than that of the other squares.

In this section we have shown empirical evidence for the good correspondence between results obtained via simulation and those obtained via fluid flow approximation for the crowd model. In Appendix C an analytical motivation is provided for this correspondence.

6 Some Results for Nine Squares

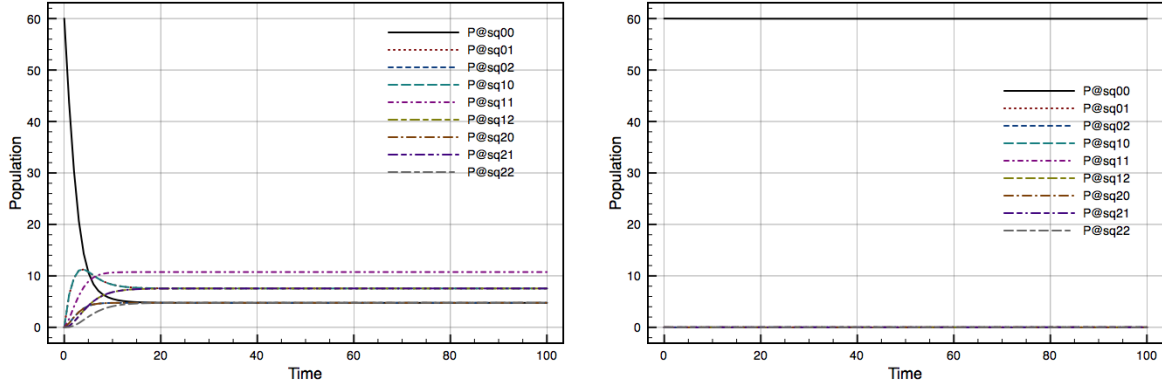
This section shows a few results for a model with nine squares in the shape of a 3×3 grid. Such a model can be obtained in a straightforward way extending the four squares model. The squares are numbered sq_{ij} where $0 \leq i, j \leq 2$ with i denoting the row index and j the column index.

Fig. 10(a) shows the result of a Fluid Flow analysis for $c = 0.02$ and 60 people in square sq_{00} initially. For these parameter values agents are distributing more or less evenly over the squares. The square with the highest degree, i.e. the one in the middle, gets relatively more agents. This is directly related to the degree of that node (4). The more streets are connected to a square, the more people it attracts simply because more people happen to pass by such a square. In fact, in [22] it is shown that the average number of people present under such circumstances is *proportional* to the number of streets connected to the square.

Fig. 10(b) shows the result for $c = 0.15$. All agents group in node sq_{00} and all other nodes remain empty. This is in line with results in [22] obtained via simulation.

7 Squares with Different Attractiveness

The four squares model presented in Section 4 can be enriched with a further parameter, an attractiveness factor, that models that some squares are more attractive than others, for example, due to the presence

(a) Fluid Flow analysis, $c=0.02$.(b) Fluid Flow analysis, $c=0.15$.Figure 10: Nine squares, 60 agents in square sq_{00} initially.

of bars. The idea is that the presence of bars has an impact on the probability that an agent remains in such a square. In the model with bars the agents follow two more rules: (1) the probability of an agent to remain in a square with a bar is higher than in a square without bar, (2) when agents leave a square they are more attracted by a neighbouring square with a bar than by one without bar.

The model in Section 2 can be adapted to include the above two rules in the following way. First the formula for the number of people expected to leave a square i with attractiveness α_i is adapted as follows:

$$f_i(t) = p_i(t)(1 - \alpha_i c)^{(p_i(t)-1)}$$

where $\alpha_i > 0$ is such that $0 \leq \alpha_i c \leq 1$. In order to model rule (2) the probabilities of agents moving to adjacent squares need to be defined. Note that in this case an agent that decides to leave a square may move with a higher probability to one adjacent square rather than to another one. However, the total probability of such an agent to move to any of the adjacent squares needs to be equal to 1. This leads to the following matrix with elements A_{ij} denoting the probability that an agent moves from square j to square i :

$$A_{ij} = \frac{\text{con}_{ij} \alpha_i}{\sum_k \text{con}_{kj} \alpha_k}$$

The probability to move from square j to square i now depends on the relative attractiveness α_i of that of square i (connected to j) w.r.t. the total attractiveness $\sum_k \text{con}_{kj} \alpha_k$ of the adjacent squares of square j . Note that indeed the sum of the probabilities to go to any of the adjacent squares i of square j is 1:

$$\sum_i A_{ij} = \sum_i \frac{\text{con}_{ij} \alpha_i}{\sum_k \text{con}_{kj} \alpha_k} = 1$$

It can easily be shown that also in this case there is a fixed point satisfying $A\mathbf{f} = \mathbf{f}$. In particular the vector $\mathbf{v} = (v_1, v_2, \dots, v_n)$ is such an eigenvector of A (with bars) with eigenvalue 1, where v_i is defined as $v_i = \alpha_i \sum_j \text{con}_{ij} \alpha_j$:

$$\begin{aligned}
(\mathbf{Av})_i &= \sum_j A_{ij} v_j \\
&= \{\text{Def. of } A_{ij} \text{ and } v_j\} \sum_j \frac{\text{con}_{ij} \alpha_i}{\sum_k \text{con}_{kj} \alpha_k} \alpha_j \sum_h \text{con}_{jh} \alpha_h \\
&= \{\text{For all } i, j : \text{con}_{ij} = \text{con}_{ji}\} \sum_j \frac{\text{con}_{ij} \alpha_i}{\sum_k \text{con}_{jk} \alpha_k} \alpha_j \sum_h \text{con}_{jh} \alpha_h \\
&= \{\text{Replace } k \text{ with } h\} \sum_j \frac{\text{con}_{ij} \alpha_i}{\sum_h \text{con}_{jh} \alpha_h} \alpha_j \sum_h \text{con}_{jh} \alpha_h \\
&= \{\text{Algebra}\} \sum_j \text{con}_{ij} \alpha_i \alpha_j \\
&= \alpha_i \sum_j \text{con}_{ij} \alpha_j \\
&= v_i
\end{aligned}$$

The form of the fixed point v_i shows that there is relatively more traffic through attractive squares but also through squares in the vicinity of attractive ones (assuming that the attractiveness values are larger than 1). As a consequence it could be expected that there are also more people in squares of high attractiveness. In this case it is more difficult to analyse the stability of this fixed point and to find an analytical estimate for the critical transition value of c . However, as before, we can develop a Bio-PEPA model capturing squares with attractions and use fluid flow analysis to investigate phase transitions and to obtain an estimate for the critical transition value of c .

To obtain the new Bio-PEPA model we only need to adapt the functional rates of the model presented in Section 4. In the functional rate definition the chat-factor c is multiplied by the attractiveness factor α_i for any square i . Moreover, instead of dividing by the degree of the node we now multiply by the probability to go from square i to square j . This probability depends on the relative attractiveness of the squares adjacent to i and for square j is $\alpha_j / \sum_k \text{con}_{ik} \alpha_k$. For example, in the four square topology of Section 4 the functional rate $fAtB$ to go from A to B is:

$$(P@sqA * (1 - \alpha_A c)^{(P@sqA-1)}) * (\alpha_B / (\alpha_B + \alpha_C))$$

This leads to the following Bio-PEPA model assuming that square D has attractiveness factor 2 and all other squares factor 1:

location top : *size = 1000, type = compartment;*
location sqA in top : *size = normal_square, type = compartment;*
location sqB in top : *size = normal_square, type = compartment;*
location sqC in top : *size = normal_square, type = compartment;*
location sqD in top : *size = normal_square, type = compartment;*

$$\begin{aligned}
\text{normal_square} &= 100; \\
c &= 0.05; \\
\alpha_A &= 1; \\
\alpha_B &= 1; \\
\alpha_C &= 1; \\
\alpha_D &= 2;
\end{aligned}$$

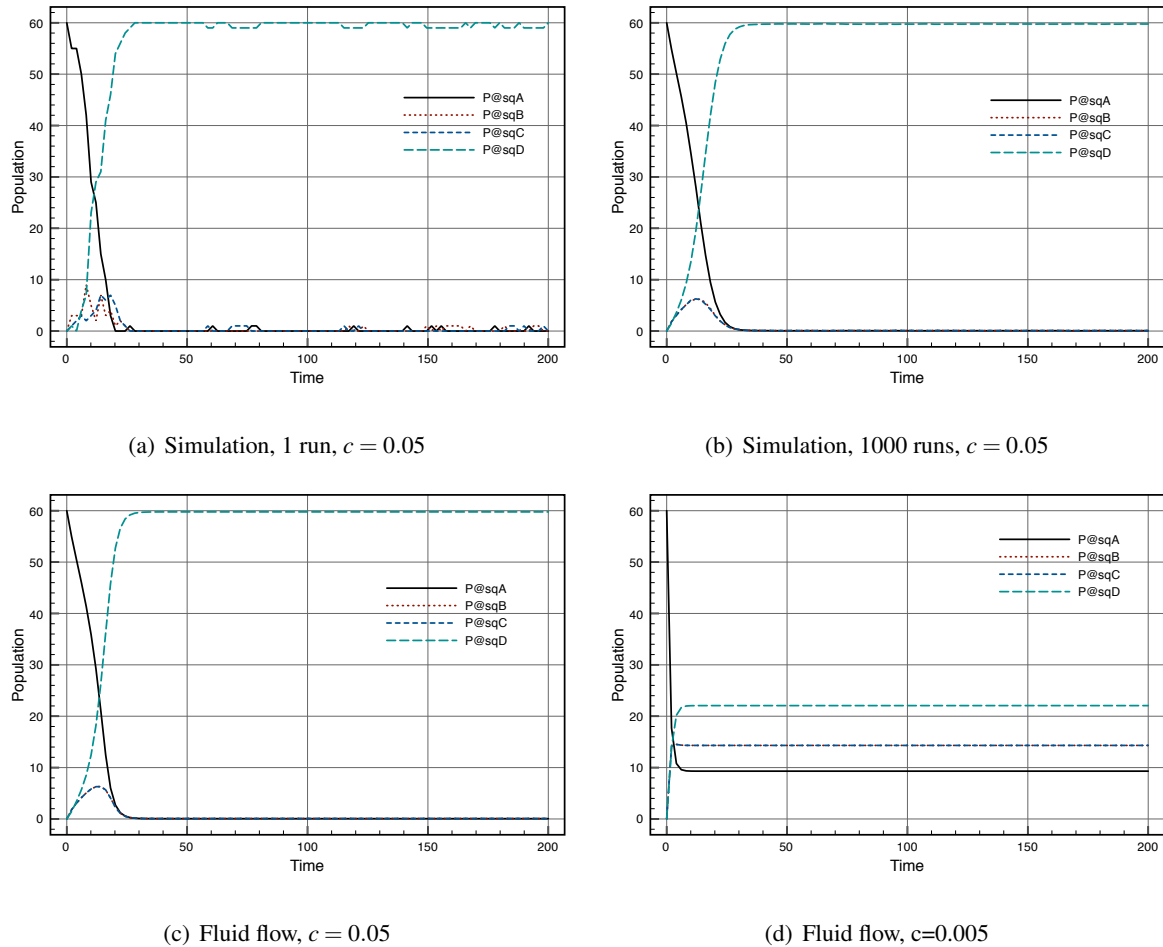
$$\begin{aligned}
\text{kineticLawOf } fAtB & : (P@sqA * (1 - \alpha_{AC})^{(P@sqA-1)}) * (\alpha_B / (\alpha_B + \alpha_C)); \\
\text{kineticLawOf } fBtA & : (P@sqB * (1 - \alpha_{BC})^{(P@sqB-1)}) * (\alpha_A / (\alpha_A + \alpha_D)); \\
\text{kineticLawOf } fBtD & : (P@sqB * (1 - \alpha_{BC})^{(P@sqB-1)}) * (\alpha_D / (\alpha_A + \alpha_D)); \\
\text{kineticLawOf } fDtB & : (P@sqD * (1 - \alpha_{DC})^{(P@sqD-1)}) * (\alpha_B / (\alpha_B + \alpha_C)); \\
\text{kineticLawOf } fAtC & : (P@sqA * (1 - \alpha_{AC})^{(P@sqA-1)}) * (\alpha_C / (\alpha_B + \alpha_C)); \\
\text{kineticLawOf } fCtA & : (P@sqC * (1 - \alpha_{CC})^{(P@sqC-1)}) * (\alpha_A / (\alpha_A + \alpha_D)); \\
\text{kineticLawOf } fCtD & : (P@sqC * (1 - \alpha_{CC})^{(P@sqC-1)}) * (\alpha_D / (\alpha_A + \alpha_D)); \\
\text{kineticLawOf } fDtC & : (P@sqD * (1 - \alpha_{DC})^{(P@sqD-1)}) * (\alpha_C / (\alpha_B + \alpha_C));
\end{aligned}$$

$$\begin{aligned}
P = & fAtB[sqA \rightarrow sqB] \odot P + fBtA[sqB \rightarrow sqA] \odot P + \\
& fAtC[sqA \rightarrow sqC] \odot P + fCtA[sqC \rightarrow sqA] \odot P + \\
& fBtD[sqB \rightarrow sqD] \odot P + fDtB[sqD \rightarrow sqB] \odot P + \\
& fCtD[sqC \rightarrow sqD] \odot P + fDtC[sqD \rightarrow sqC] \odot P;
\end{aligned}$$

$$(P@sqA[60] \boxtimes_* P@sqB[0]) \boxtimes_* (P@sqC[0] \boxtimes_* P@sqD[0])$$

Fig. 11(a) shows a single simulation run of the model with $c = 0.05$, and $\alpha_D = 2$ and $\alpha_X = 1$ for $X \in \{A, B, C\}$. Initially there are 60 agents in square A and the other squares are empty. The figure clearly shows that the agents are attracted to square D for this value of c . This behaviour is confirmed by the results of a simulation with 1000 independent runs shown in Fig. 11(b) and a fluid flow approximation shown in Fig. 11(c). The correspondence between the simulation results and the fluid flow approximation is again good. Fig. 11(d) shows a fluid flow approximation of the same model, but with a much smaller value $c = 0.005$. This indicates that also for this model for values of c below a certain threshold the agents distribute more or less evenly over the squares. However, more attractive squares, and those adjacent to those squares get relatively more agents.

In Fig. 12(a) the effect of the value of c is explored for the model with bars with $\alpha_D = 2$ and $\alpha_X = 1$ for $X \in \{A, B, C\}$. The other initial values are as for the model without bars, so 60 agents in A and none in B, C and D . In Fig. 12(a) a kind of double phase-shift can be observed. For very small values of $c < 0.0001$ the steady state of the model shows a distribution of agents over squares that is approximately proportional to the values in the eigenvector of the model which is $(2, 3, 3, 4)$. In fact, for so small values of c the average number of agents in square A is $(2/12) * 60 = 10$, and similarly for the other squares. With the increase of the chat-factor this distribution of agents in the steady state is slowly changing towards one in which there is a clear tendency of agents to stay in the attractive square D . Around the value $c = 0.07$ another much more abrupt transition can be observed. Simulations show that for values of c between approximately 0.07 and 0.1 the fixed point is unstable which means that sometimes the agents group in square D and sometimes in square A in the long term. With the value of c getting closer to 0.1 the agents group more and more often in A rather than in D . For values of $c > 0.1$ agents do no longer group in the attractive square D in the long run, but remain in their initial square A . Fig. 12(b) and Fig. 12(c) show two single simulation runs for the same model with $c = 0.08$. They illustrate that in one case agents group in square D whereas in the other case the agents remain mainly in square A .

Figure 11: Attraction of agents to square D with $\alpha_D = 2$

8 Scalability

In this section we provide some data illustrating the scalability of a fluid flow analysis with respect to stochastic simulation for two crowd models. Table 1 shows the evaluation times for the crowd model of the simple ring topology with four squares (4SQ) for Gillespie's stochastic simulation with 10 independent runs (GIL(10)) and fluid flow (ODE) respectively. For a population of 60 agents initially in one of the squares, $c = 0.005$ and over a period of 200 time units simulation takes 241 ms. whereas fluid flow analysis takes 11 ms. For a very large population of 600,000 agents, stochastic simulation takes approximately 39 minutes whereas fluid flow analysis takes only 6 ms., which is even faster than for a population of 60. The latter can be explained by the fact that the numerical algorithm for fluid flow has an adaptive step-size. Often a larger population of individual independent agents shows more smooth behaviour, allowing for larger steps taken by the algorithm thus saving evaluation time.

Table 2 shows the evaluation times for the crowd model of nine squares placed in a 3 by 3 grid as in Sect. 6 (9SQ). Evaluation times of fluid flow for this larger model are still extremely low and essentially constant with respect to the size of the population. The evaluation time for simulation in this table now refers to a *single* run and, in case of a population of 600,000 agents, amounts to approximately 27

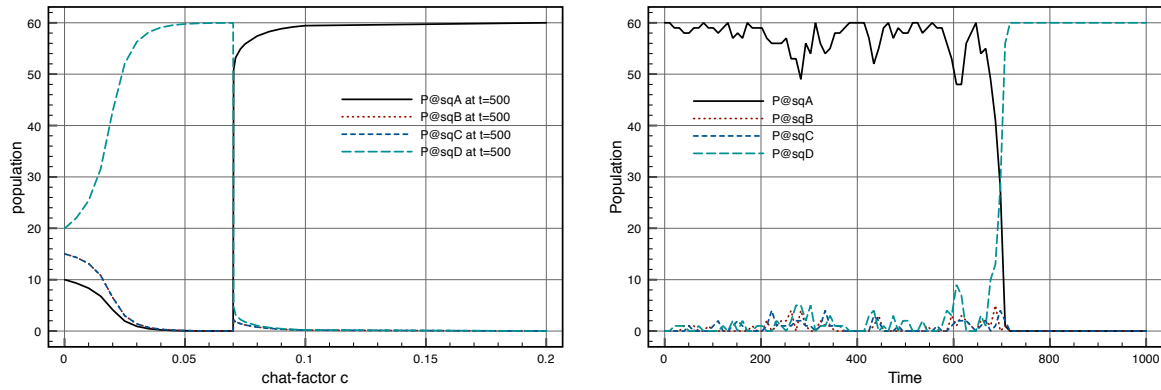
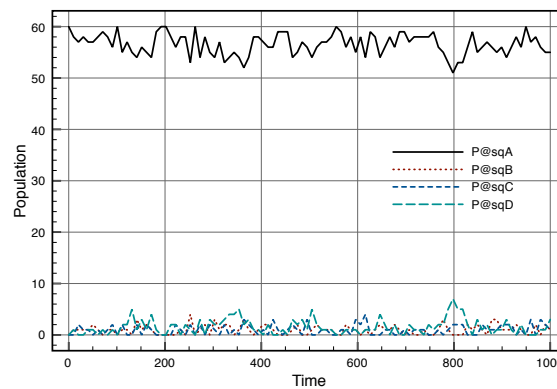
(a) Fluid flow for various values of c (b) Simulation single run for $c = 0.08$ (c) Simulation single run for $c = 0.08$

Figure 12: Phase transitions for model with bars

minutes.

The results have been obtained with the Bio-PEPA plugin tool for Eclipse Helios on a MacPro4,1.

9 Conclusions and Further Work

The modelling and analysis of crowd dynamics appears to be an active and open research topic. We have explored the application of the stochastic process algebra Bio-PEPA to model a simple but interesting case study. This concerns the emergent self-organisation of parties in the squares of a city where people moving between squares are modelled as independently behaving agents following a few simple local rules. Bio-PEPA is enhanced with various analysis techniques among which stochastic simulation, but in particular an efficient and scalable fluid flow approximation. Such an approximation provides a very efficient method to obtain the average number of agents in a particular state over time. In the case study at hand fluid flow approximation provides an analysis of the number of people, on average, that are present in the various squares when time evolves.

We have shown how this approach can be used to investigate non-linear emergent behaviour that

Population	period	c	GIL (10)	ODE
60	200	0.005	241 ms.	11 ms.
600	200	0.0005	2,359 ms.	10 ms.
60,000	200	0.000005	237,854 ms.	8 ms.
600,000	200	0.0000005	2,357,161 ms.	6 ms.

Table 1: Evaluation times for SQ4 over 200 time units.

Population	period	c	GIL (1)	ODE
60	500	0.005	155 ms.	21 ms.
600	500	0.0005	1,673 ms.	20 ms.
60,000	500	0.000005	165,415 ms.	20 ms.
600,000	500	0.0000005	1,627,018 ms.	20 ms.

Table 2: Evaluation times for 9SQ over 500 time units.

arises from the distributed local interaction between agents. The results are shown to correspond well to those found in the literature where they were obtained by means of more elaborate and time-consuming discrete event simulation. We have also modelled and analysed an extension of the crowd model with squares that have different attractiveness. Also this extended model shows non-linear behaviour and fluid flow analysis has shown to be a viable approach to identify the critical threshold value in a pragmatic and efficient way compared to traditional analytical or simulation based methods.

The advantage of the fluid flow approximation with respect to simulation is that it is much faster than simulation if one is interested in the average, possibly non-linear, behaviour of a system over time. Furthermore, Bio-PEPA is based on a modular, high-level language providing notions of locality and context dependency. These features make Bio-PEPA a promising candidate for the modelling of a class of systems that goes beyond the bio-molecular applications it was originally designed for [5, 11, 1]. Besides providing empirical evidence of the correspondence between simulation and fluid flow results we also provided an analytical analysis that explains this correspondence.

Future work is developing along a few main directions. We are interested in developing further linguistic abstractions to more precisely describe the dynamics of systems with a large number of mobile agents displaced in a, possibly open, physical environment. We are furthermore interested in conducting more fundamental research on the fluid flow approach and its relationship to emergent non-linear behaviour, in particular its relation to mean field analysis [7] and the formal relationship between discrete and continuous models. Along these lines we are currently studying a version of the crowd model with density dependent functional rates in which agents consider only a fraction of the total population in a square in order to decide to stay or leave the square.

Acknowledgments. The authors would like to thank Stephen Gilmore, Maria Luisa Gorriero, Allan Clark and Adam Duguid (The University of Edinburgh) for their support with the Bio-PEPA plug-in, Jeremy Bradley and Richard Hayden (Imperial College London) and Luca Bortolussi (University of Trieste) for their helpful discussions on approximation and Michael D. Harrison (Newcastle University) for his suggestion to have a closer look at the “El Botellón” case study in relation to Bio-PEPA. This research has been partially funded by the CNR project RSTL-XXL and by the Italian PRIN MIUR project PaCO. Jane Hillston is supported by the EPSRC ARF EP/c543696/01.

References

- [1] O. E. Akman, F. Ciocchetta, A. Degasperis & M. L. Guerriero (2009): *Modelling biological clocks with Bio-PEPA: Stochasticity and robustness for the neurospora crassa circadian network*. In: In proceedings of CSBM, LNCS 5688, Springer, pp. 52–67.
- [2] M. Benaïm & J.-Y. Le Boudec (2008): *A Class of Mean Field Interaction Models for Computer and Communication Systems*. *Performance Evaluation* 65(11-12), pp. 823–838.
- [3] Bio-PEPA Home Page. <http://www.biopepa.org/>.
- [4] L. Bortolussi (2008): *On the Approximation of Stochastic Concurrent Constraint Programming by Master Equation*. *ENTCS* 220, pp. 163–180. Available at <http://portal.acm.org/citation.cfm?id=1480238.1480262>.
- [5] A. Bracciali, J. Hillston, D. Latella & M. Massink (2010): *Reconciling Population and Agent Models for Crowd Dynamics*. In: *To appear in proceedings of 3rd International Workshop on Logics, Agents, and Mobility (LAM'10)*.
- [6] M. Calder, S. Gilmore & J. Hillston (2005): *Automatically deriving ODEs from process algebra models of signalling pathways*. In: *Gordon Plotkin (Ed.), Proceedings of Computational Methods in Systems Biology (CMSB 2005)*, pp. 204–215.
- [7] A. Chaitreau, J.-Y. Le Boudec & N. Ristanovic (2009): *The Age of Gossip: Spatial Mean Field Regime*. In: *Proceedings of SIGMETRICS/Performance'09*, ACM, pp. 109–120.
- [8] F. Ciocchetta, A. Duguid, S. Gilmore, M. L. Guerriero & Hillston J. (2009): *The Bio-PEPA Tool Suite*. In: *Proc. of the 6th Int. Conf. on Quantitative Evaluation of SysTems (QEST 2009)*, pp. 309–310.
- [9] F. Ciocchetta & M. L. Guerriero (2009): *Modelling Biological Compartments in Bio-PEPA*. *ENTCS* 227, pp. 77–95. Available at <http://dx.doi.org/10.1016/j.entcs.2008.12.105>.
- [10] F. Ciocchetta & J. Hillston (2008): *Bio-PEPA: An Extension of the Process Algebra PEPA for Biochemical Networks*. *ENTCS* 194(3), pp. 103–117. Available at <http://dx.doi.org/10.1016/j.entcs.2007.12.008>.
- [11] F. Ciocchetta & J. Hillston (2009): *Bio-PEPA: A framework for the modelling and analysis of biological systems*. *TCS* 410(33-34), pp. 3065–3084.
- [12] J. R. Dormand & P. J. Prince (1980): *A family of embedded Runge-Kutta formulae*. *Journal of Computational and Applied Mathematics* 6(1), pp. 19–26.
- [13] D. T. Gillespie (1977): *Exact Stochastic Simulation of Coupled Chemical Reactions*. *The Journal of Physical Chemistry* 81(25), pp. 2340–2361.
- [14] R. A. Hayden & J. T. Bradley (2010): *A fluid analysis framework for a Markovian process algebra*. *TCS* 411(22-24), pp. 2260–2297.
- [15] R.A. Hayden (2008). *Addressing the state space explosion problem for PEPA models through fluid-flow approximation*. Msci thesis, Imperial College London.
- [16] J. Hillston (2005): *Fluid flow approximation of PEPA models*. In: *Proceedings of QEST'05*, IEEE Computer Society, pp. 33–43.
- [17] J. L. W. V. Jensen (1906): *Sur les fonctions convexes et les inégalités entre les valeurs moyennes*. *Acta Mathematica* 30(1), pp. 175–193.
- [18] T. G. Kurtz (1970): *Solutions of Ordinary Differential Equations as Limits of Pure Markov Processes*. *Journal of Applied Probability* 7(1), pp. 49–58.
- [19] M. Kwiatkowska, G. Norman & D. Parker (2009): *PRISM: Probabilistic Model Checking for Performance and Reliability Analysis*. *ACM SIGMETRICS Performance Evaluation Review* .
- [20] M. Massink, D. Latella, A. Bracciali & M. Harrison (2010): *A Scalable Fluid Flow Process Algebraic Approach to Emergency Egress Analysis*. In: *Proceedings of the 8th International Conference on Software Engineering and Formal Methods (SEFM 2010)*, IEEE, pp. 169–180.

- [21] M. Massink, D. Latella, A. Bracciali & J. Hillston (2010). *Modelling Crowd Dynamics in Bio-PEPA – Extended Abstract*. Available at <http://pastaworkshop.org/proceedings/>. Participants proceedings of PASTA 2010, no formal proceedings will appear.
- [22] J. E. Rowe & R. Gomez (2003): *El Botellón: Modeling the Movement of Crowds in a City*. *Complex Systems* 14, pp. 363–370.
- [23] G. Santos & B. E. Aguirre (2005): *A critical review of emergency evacuation simulation models*. In: *Proceedings of the NIST Workshop on Building Occupant Movement during Fire Emergencies, June 10-11, 2004*, NIST/BFRL Publications Online, Gaithersburg, MD, USA, pp. 27–52.
- [24] G. K. Still (2000). *Crowd Dynamics*. Ph.D. Thesis, University of Warwick.

Appendix

A Bio-PEPA Model for Four Squares

```
location top : size = 1000, type = compartment ;
location sqA in top : size = normal_square, type = compartment ;
location sqB in top : size = normal_square, type = compartment ;
location sqC in top : size = normal_square, type = compartment ;
location sqD in top : size = normal_square, type = compartment ;
```

```
normal_square = 100 ;
c = 0.05 ; // chat-probability or conservatism factor
d = 2 ; // degree of node
```

```
kineticLawOf fAtB : (P@sqA * (1-c)^(P@sqA-1))/d ;
kineticLawOf fBtA : (P@sqB * (1-c)^(P@sqB-1))/d ;
kineticLawOf fBtD : (P@sqB * (1-c)^(P@sqB-1))/d ;
kineticLawOf fDtB : (P@sqD * (1-c)^(P@sqD-1))/d ;
kineticLawOf fAtC : (P@sqA * (1-c)^(P@sqA-1))/d ;
kineticLawOf fCtA : (P@sqC * (1-c)^(P@sqC-1))/d ;
kineticLawOf fCtD : (P@sqC * (1-c)^(P@sqC-1))/d ;
kineticLawOf fDtC : (P@sqD * (1-c)^(P@sqD-1))/d ;
```

```
P = fAtB[sqA->sqB](.)P + fBtA[sqB->sqA](.)P +
    fAtC[sqA->sqC](.)P + fCtA[sqC->sqA](.)P +
    fBtD[sqB->sqD](.)P + fDtB[sqD->sqB](.)P +
    fCtD[sqC->sqD](.)P + fDtC[sqD->sqC](.)P ;
```

```
(P@sqA[60] <*> P@sqB[0]) <*> (P@sqC[0] <*> P@sqD[0])
```

	fAtB	fBtA	fAtC	fCtA	fBtD	fDtB	fCtD	fDtC	
PsqA	-1	+1	-1	+1	0	0	0	0	x_{PsqA}
PsqB	+1	-1	0	0	-1	+1	0	0	x_{PsqB}
PsqC	0	0	+1	-1	0	0	-1	+1	x_{PsqC}
PsqD	0	0	0	0	+1	-1	+1	-1	x_{PsqD}

Table 3: Matrix D associated with the crowd model.

B The Crowd Model as a Set of Ordinary Differential Equations

The Bio-PEPA model can be translated into a set of ODEs following a similar method as that described by Calder et al. for PEPA (reagent centric view) in [6] and adapted to Bio-PEPA by Ciocchetta and Hillston in [10]. There the method is explained in terms of biochemical reactions which is the field of application for which Bio-PEPA was originally designed. Here we present the same method in more neutral terms using the specification in Section 4 as a running example.

The first step we perform is to replace the shorthand notation used in the model in Section 4 by a notation in plain Bio-PEPA. This way we obtain separate definitions for the population in each of the four squares. The species components become:

$$\begin{aligned}
PsqA &= (fAtB, 1)\downarrow PsqA + (fAtC, 1)\downarrow PsqA + \\
&\quad (fBtA, 1)\uparrow PsqA + (fCtA, 1)\uparrow PsqA \\
PsqB &= (fBtA, 1)\downarrow PsqB + (fBtD, 1)\downarrow PsqB + \\
&\quad (fAtB, 1)\uparrow PsqB + (fDtB, 1)\uparrow PsqB \\
PsqC &= (fCtA, 1)\downarrow PsqC + (fCtD, 1)\downarrow PsqC + \\
&\quad (fAtC, 1)\uparrow PsqC + (fDtC, 1)\uparrow PsqC \\
PsqD &= (fDtB, 1)\downarrow PsqD + (fDtC, 1)\downarrow PsqD + \\
&\quad (fBtD, 1)\uparrow PsqD + (fCtD, 1)\uparrow PsqD
\end{aligned}$$

The model component becomes:

$$(PsqA[60] \bowtie_{*} PsqB[0]) \bowtie_{*} (PsqC[0] \bowtie_{*} PsqD[0])$$

Also in the definitions of the functional rates we need to replace $P@sqX$ by $PsqX$ for all X in $\{A,B,C,D\}$.

The derivation of the set of ODEs from the Bio-PEPA specification is now based on the following three steps:

1. The first step is the definition of a matrix D of $(n \times m)$. Here n is the number of species components, in this case there are four: $PsqA$, $PsqB$, $PsqC$ and $PsqD$. The size m is the number of actions, in this case there are eight: $fAtB$, $fBtA$, ..., $fCtD$. The elements of the matrix D_{ij} indicate whether species component i increases or decreases when action j is performed, and how many elements of species component i are involved when the action is performed. For example, if $i = PsqA$ and $j = fAtB$ then we observe in the model definition that when $fAtB$ is performed the number of agents in square A reduces by 1. In the matrix this is presented as '-1' at place D_{ij} . The matrix D is also called the stoichiometry matrix in the context of biochemistry. For the crowds model we obtain the matrix in Table 3 where each species component C_i is also associated with a variable x_i .

- The second step is the definition of the functional rate vector ($m \times 1$) \mathbf{v}_{fr} . This vector contains the functional rate as defined by the kinetic laws for each action. Note that the kinetic laws in this model are a function of time since they depend in the number of agents present in a particular square at time instant t . For the crowd example we obtain the vector:

$$\mathbf{v}_{fr}^T(t) = (fAtB(t), fBtA(t), fAtC(t), fCtA(t), fBtD(t), fDtB(t), fCtD(t), fDtC(t))$$

where T indicates the transpose of the vector. Note the slight overload of names $fXtY$ to stand both for the action as for its associated functional rate.

- Finally, the vector of species variables ($n \times 1$) \mathbf{x} is defined, which for the crowds example becomes:

$$\mathbf{x}^T(t) = (x_{P_{sqA}}(t), x_{P_{sqB}}(t), x_{P_{sqC}}(t), x_{P_{sqD}}(t))$$

The system of ODEs can now be obtained as:

$$\frac{d\mathbf{x}(t)}{dt} = D \times \mathbf{v}_{fr}(t)$$

For the crowd model we obtain the following set of ODEs:

$$\begin{aligned} \frac{d\mathbf{x}_{P_{sqA}}(t)}{dt} &= -fAtB(t) + fBtA(t) - fAtC(t) + fCtA(t) \\ \frac{d\mathbf{x}_{P_{sqB}}(t)}{dt} &= +fAtB(t) - fBtA(t) - fBtD(t) + fDtB(t) \\ \frac{d\mathbf{x}_{P_{sqC}}(t)}{dt} &= +fAtC(t) - fCtA(t) - fCtD(t) + fDtC(t) \\ \frac{d\mathbf{x}_{P_{sqD}}(t)}{dt} &= +fBtD(t) - fDtB(t) + fCtD(t) - fDtC(t) \end{aligned}$$

If we substitute the functional rates by their definition and consider the crowd model for a 2×2 grid of squares then we obtain the following ODE for $\mathbf{x}_{P_{sqA}}$, the other equations are similar:

$$\begin{aligned} \frac{d\mathbf{x}_{P_{sqA}}(t)}{dt} &= -(\mathbf{x}_{P_{sqA}}(t) * (1 - c)^{(\mathbf{x}_{P_{sqA}}(t)-1)})/2 \\ &\quad + (\mathbf{x}_{P_{sqB}}(t) * (1 - c)^{(\mathbf{x}_{P_{sqB}}(t)-1)})/2 \\ &\quad - (\mathbf{x}_{P_{sqA}}(t) * (1 - c)^{(\mathbf{x}_{P_{sqA}}(t)-1)})/2 \\ &\quad + (\mathbf{x}_{P_{sqC}}(t) * (1 - c)^{(\mathbf{x}_{P_{sqC}}(t)-1)})/2 \end{aligned}$$

Note that this set of ODEs is non-linear for this model. In fact, as shown by Rowe and Gomez in [22], the system gives rise to stable and unstable fixed points depending critically on the value of c .

The automatic derivation of sets of ODE from Bio-PEPA specifications is available via the experimental Bio-PEPA plug-in tool [8] developed at the University of Edinburgh.

C Formal Relationship Between CTMC and ODE Interpretation

The results in the previous sections show a very good correspondence between the results obtained via fluid flow approximation and those obtained, on the one hand, with Gillespie's stochastic simulation algorithm applied on exactly the same Bio-PEPA specification and, on the other hand, with discrete event simulation results found by Rowe and Gomez [22]. In this section we provide a justification for this correspondence from an analytical perspective.

There exist several theories that address the relation between the interpretation of the model as a large set of individual, independently behaving and interacting agents, as is the case for simulation, and a continuous deterministic interpretation of the same model as occurs with a fluid flow approximation.

Perhaps the most well-known is the theory by Kurtz [18]. Informally speaking, Kurtz shows an exact relation (in the limit when the population goes to infinite) between the two above mentioned interpretations when the rate-functions can be expressed in terms of the average *density* of the population. A similar requirement needs to be satisfied in the context of the theory of mean field analysis in for example recent work by Le Boudec et al. [2]. Unfortunately, in the crowd model the rate-functions cannot be expressed in terms of the density of the population because the exponent of the factor $(1 - c)^{(P_X - 1)}$ requires the absolute number P_X of agents present in square X .

A third approach, by Hayden and Bradley [14], has recently been applied to assess the quality of the fluid approximation for (grouped) PEPA models. In that approach the Chapman-Kolmogorov forward equations (C-K) are derived from a typical central state of the aggregated CTMC³ associated to a PEPA model. These equations are then used in the moment generating function from which, by partial differentiation, ordinary differential equations are obtained for the expected value over time of each sequential component in the PEPA model. In this section we adapt the approach to the Bio-PEPA crowds model which is characterised by non-linear rate functions.

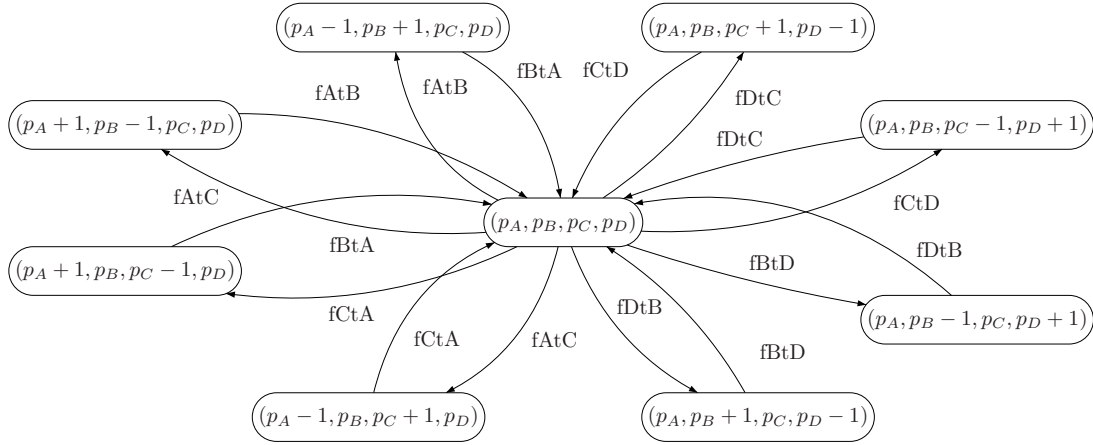


Figure 13: A central state of the crowds model.

Let p_A , p_B , p_C and p_D denote the number of agents in square A , B , C and D , respectively. A central state of the aggregated CTMC⁴ of the crowds model is shown in Fig. 13. Let $prob_{(p_A, p_B, p_C, p_D)}(t)$ denote the transient probability of being in the aggregated CTMC state with p_X agents in square X , for X in $\{A, B, C, D\}$ at time t . From the Bio-PEPA model in Sect.4 it follows that the Chapman-Kolmogorov forward equations, governing the evolution of the state probabilities over time of the underlying aggregated

³For a formal definition see for example [14].

⁴For a definition see for example [14].

CTMC can be presented as follows:

$$\begin{aligned}
\frac{dprob_{(p_A, p_B, p_C, p_D)}(t)}{dt} = & ((p_A + 1) \cdot (1 - c)^{p_A})/2 \cdot prob_{(p_A+1, p_B-1, p_C, p_D)}(t) \\
& + ((p_B + 1) \cdot (1 - c)^{p_B})/2 \cdot prob_{(p_A-1, p_B+1, p_C, p_D)}(t) \\
& + ((p_C + 1) \cdot (1 - c)^{p_C})/2 \cdot prob_{(p_A, p_B, p_C+1, p_D-1)}(t) \\
& + ((p_D + 1) \cdot (1 - c)^{p_D})/2 \cdot prob_{(p_A, p_B, p_C-1, p_D+1)}(t) \\
& + ((p_A + 1) \cdot (1 - c)^{p_A})/2 \cdot prob_{(p_A+1, p_B, p_C-1, p_D)}(t) \\
& + ((p_C + 1) \cdot (1 - c)^{p_C})/2 \cdot prob_{(p_A-1, p_B, p_C+1, p_D)}(t) \\
& + ((p_B + 1) \cdot (1 - c)^{p_B})/2 \cdot prob_{(p_A, p_B+1, p_C, p_D-1)}(t) \\
& + ((p_D + 1) \cdot (1 - c)^{p_D})/2 \cdot prob_{(p_A, p_B-1, p_C, p_D+1)}(t) \\
& - ((p_A) \cdot (1 - c)^{p_A-1})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& - ((p_B) \cdot (1 - c)^{p_B-1})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& - ((p_C) \cdot (1 - c)^{p_C-1})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& - ((p_D) \cdot (1 - c)^{p_D-1})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& - ((p_C) \cdot (1 - c)^{p_C-1})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& - ((p_D) \cdot (1 - c)^{p_D-1})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& - ((p_D) \cdot (1 - c)^{p_D-1})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t)
\end{aligned} \tag{1}$$

Each of the eight first summands in Equation (1) appears only when the state (p_A, p_B, p_C, p_D) has the corresponding incoming transitions in the aggregated state space.

Let us now consider as an example, following the approach outlined in [15, 14], how an ODE for the function $P_A(t)$ can be obtained. First note that the expected value of $P_A(t)$ is given by:

$$\mathbb{E}[P_A(t)] = \sum_{(p_A, p_B, p_C, p_D)} p_A \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \tag{2}$$

where the sum ranges over all possible states of the model. This leads to the following derivation:

$$\begin{aligned}
\frac{d\mathbb{E}[P_A(t)]}{dt} &= \{\text{By def. of expected value}\} \\
\frac{d\sum_{(p_A, p_B, p_C, p_D)} p_A \cdot prob_{(p_A, p_B, p_C, p_D)}(t)}{dt} &= \{\text{By distribution of differentiation}\} \\
\sum_{(p_A, p_B, p_C, p_D)} p_A \cdot \frac{dprob_{(p_A, p_B, p_C, p_D)}(t)}{dt} &= \{\text{By definition of C-K equations}\} \\
&\sum_{(p_A, p_B, p_C, p_D)} [\\
& ((p_A - 1) \cdot p_A \cdot (1 - c)^{(p_A-1)})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& + ((p_A + 1) \cdot p_B \cdot (1 - c)^{(p_B)})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& + (p_A \cdot p_C \cdot (1 - c)^{(p_C-1)})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& + (p_A \cdot p_D \cdot (1 - c)^{(p_D-1)})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& + ((p_A - 1) \cdot p_A \cdot (1 - c)^{(p_A-1)})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& + ((p_A + 1) \cdot p_C \cdot (1 - c)^{(p_C-1)})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& + (p_A \cdot p_B \cdot (1 - c)^{(p_B-1)})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& + (p_A \cdot p_D \cdot (1 - c)^{(p_D-1)})/2 \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& - (p_A \cdot p_A \cdot (1 - c)^{(p_A-1)}) \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& - (p_A \cdot p_B \cdot (1 - c)^{(p_B-1)}) \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& - (p_A \cdot p_C \cdot (1 - c)^{(p_C-1)}) \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
& - (p_A \cdot p_D \cdot (1 - c)^{(p_D-1)}) \cdot prob_{(p_A, p_B, p_C, p_D)}(t) \\
&]
\end{aligned}$$

If $(P_A(t), P_B(t), P_C(t), P_D(t))$ is the state of the aggregated CTMC at time t , then by cancelling terms in the above equation one obtains:

$$\sum_{(P_A, P_B, P_C, P_D)} \left[\begin{aligned} & -p_A \cdot (1-c)^{(P_A-1)} \cdot \text{prob}_{(P_A, P_B, P_C, P_D)}(t) \\ & + (p_B \cdot (1-c)^{(P_B-1)} \cdot \text{prob}_{(P_A, P_B, P_C, P_D)}(t))/2 \\ & + (p_C \cdot (1-c)^{(P_C-1)} \cdot \text{prob}_{(P_A, P_B, P_C, P_D)}(t))/2 \end{aligned} \right]$$

This yields in terms of expectations the following ODE for $P_A(t)$ ⁵:

$$\frac{d\mathbb{E}[P_A(t)]}{dt} = \begin{aligned} & -\mathbb{E}[P_A(t) \cdot (1-c)^{(P_A(t)-1)}] \\ & + (\mathbb{E}[P_B(t) \cdot (1-c)^{(P_B(t)-1})])/2 \\ & + (\mathbb{E}[P_C(t) \cdot (1-c)^{(P_C(t)-1})])/2 \end{aligned} \quad (3)$$

If at this point, as shown in [14], the functions $(1-c)^{(X-1)}$ were just constant rates, then expectation would just distribute over multiplication and one would obtain an equation in terms of expectations of populations. However, in our case the rate is a more complicated function and, in general, expectation does not distribute over an arbitrary function, i.e. $\mathbb{E}[\phi(X)] \neq \phi(\mathbb{E}[X])$. This means that exact equality cannot be obtained this way. As an alternative we consider whether $\mathbb{E}[P_A(t)](1-c)^{(\mathbb{E}[P_A(t)]-1)}$ could be expected to approximate $\mathbb{E}[P_A(t) \cdot (1-c)^{(P_A(t)-1)}]$. To address this question we recall Jensen's work [17] from which it is known that for strictly convex functions ϕ the following inequality holds:

$$\mathbb{E}[\phi(X)] \geq \phi(\mathbb{E}[X])$$

with *equality* when ϕ is linear or when it is a constant. The reverse inequality holds in case ϕ is strictly concave. So this requires a closer investigation of the particular function at hand. In Fig. 14 graphs are shown of the function $\phi(x) = x(1-c)^x$ for values of $x \in [0 \dots 60]$ and for various values of chat probability c . It can be observed that for very small values of c the function is almost linear for the number of agents considered. For larger values of c the function is mostly concave and tends to an almost constant value. This implies that, informally speaking, any hypothetical probability distribution of the values of x would be mapped on an increasingly "shrinking" version of the distribution of $\phi(x)$. This means that $\mathbb{E}[P_A(t)](1-c)^{(\mathbb{E}[P_A(t)]-1)}$ approximates $\mathbb{E}[P_A(t) \cdot (1-c)^{(P_A(t)-1)}]$ indeed rather well.

Distributing expectation over the function the following ODE for the expected value of $P_A(t)$ is obtained:

$$\frac{d\mathbb{E}[P_A(t)]}{dt} = \begin{aligned} & -\mathbb{E}[P_A(t)](1-c)^{(\mathbb{E}[P_A(t)]-1)} \\ & + (\mathbb{E}[P_B(t)](1-c)^{(\mathbb{E}[P_B(t)]-1)})/2 \\ & + (\mathbb{E}[P_C(t)](1-c)^{(\mathbb{E}[P_C(t)]-1)})/2 \end{aligned} \quad (4)$$

This ODE is identical to the one obtained in the previous section for variable $x_{P_{sqA}}(t)$. In a similar way the ODEs for the other stochastic variables can be obtained. This explains why the results obtained in this paper for fluid flow and for stochastic simulation correspond so closely.

The above approximation shown in Equation (4) can also be obtained as a first order approximation of the time-dependent Taylor expansion (see [4]) of the function $\Phi(X(t)) = X(t)(1-c)^{(X(t)-1)}$ around the average value $\mathbb{E}[X(t)]$. Of course, also in this case further inspection of the function Φ is required in order to get an idea of the quality of this approximation.

⁵An alternative way to obtain Equation (3) from Equation (2) is described by Bortolussi in [4]. There the well-known Master Equation is used which is an equation very similar to the Chapman-Kolmogorov forward equations.

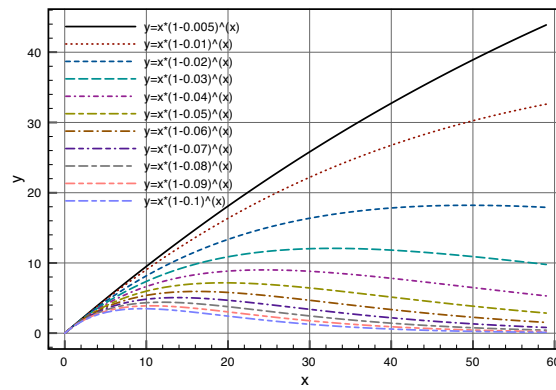


Figure 14: Function $x(1 - c)^x$ for up to 60 agents and different chat probabilities.