

# Performance Analysis of Collective Adaptive Behaviour in Time and Space

Cheng Feng<sup>1,2</sup>

*School of Informatics  
University of Edinburgh  
Edinburgh, UK*

Marco Gribaudo<sup>3</sup>

*Dipartimento di Elettronica, Informazione e Bioingegneria  
Politecnico di Milano  
Milan, Italy*

Jane Hillston<sup>1,4</sup>

*School of Informatics  
University of Edinburgh  
Edinburgh, UK*

---

## Abstract

Many systems, both natural and engineered, exhibit collective adaptive behaviour. Natural examples are swarming animals and insects, and man-made examples include swarm robots and sensor networks. Indeed many systems have both human and ICT-based agents in play, distributed over some geographical region: an informatics environment as defined by Milner. Over recent years there has been increased interest in modelling these systems in order to understand their dynamic behaviour. Here we consider the subsequent issue of how to define useful performance measures for such systems, based on consideration of a simple, intuitive example.

*Keywords:* Spatio-temporal modelling, performance modelling, collective adaptive behaviour

---

## 1 Introduction

Systems which exhibit collective behaviour have many interesting properties. Examples from the natural world such as swarming animals and insects are often studied

---

<sup>1</sup> This work is partially supported by the QUANTICOL project 600708.

<sup>2</sup> Email: [s1109873@sms.ed.ac.uk](mailto:s1109873@sms.ed.ac.uk)

<sup>3</sup> Email: [marco.gribaudo@polimi.it](mailto:marco.gribaudo@polimi.it)

<sup>4</sup> Email: [jane.hillston@ed.ac.uk](mailto:jane.hillston@ed.ac.uk)

for their emergent behaviour, patterns which become apparent at the population-level but which were not readily apparent from the described behaviour of individuals. In engineered systems this emergent behaviour constitutes the *performance* of the system.

Increasingly large-scale, geographically distributed ICT systems are being developed to support human endeavour in a variety of ways. This can be considered to be the realisation of the *informatics environment* predicted by Milner and Weiser [10,13]. Such systems operate with human agents interacting almost transparently with computing elements. Examples include smart city applications such as smart transportation, smart grid and many modern automotive systems.

In such systems, their transparency and pervasiveness mean that it is perhaps more important than ever to investigate their behaviour from both a qualitative and quantitative point of view prior to deployment. Work is currently underway, for example in the QUANTICOL project [6] to develop modelling formalisms to capture the behaviour of these systems [5,9]. Here we start a complementary investigation into the types of measure that can be derived from spatio-temporal systems. Classic performance measures assume that there is a single locus of operation. When there is a (limited) spatial aspect to behaviour, state annotations are usually used to syntactically distinguish the different locations and regular performance measures are applied. We seek to take a more radical approach to support modelling in which space is modelled explicitly and exploited fully when characterising the behaviour of the system.

## 2 Leader and Follower Scenarios

We consider a simple scenario in which agents are moving in a two-dimensional grid, as shown in Figure 1. We assume that the grid is finite and that the boundaries are

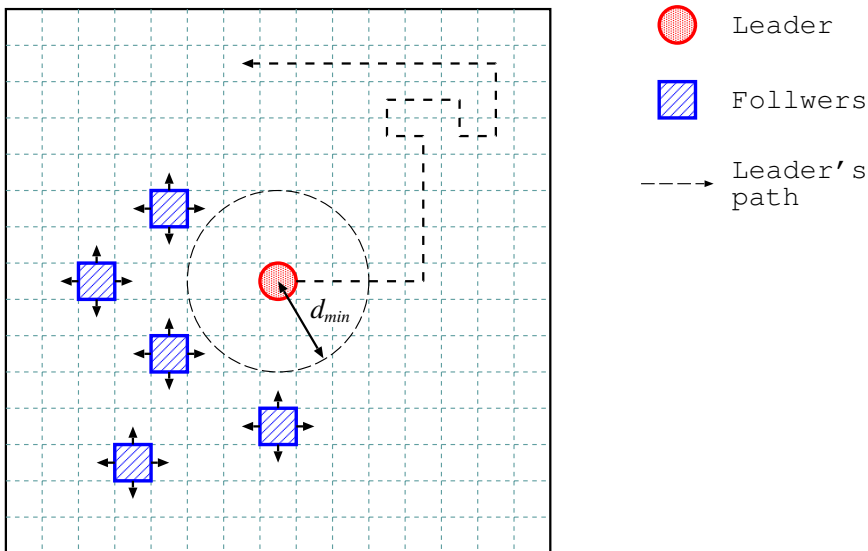


Fig. 1. The *Leader-Follower* scenario.

wrapped, meaning that essentially we are considering movement on a torus. Each agent moves one step at a time and at each step can move in any direction: north, east, south or west. We assume that there is one distinguished agent, the *Leader* who moves autonomously, performing a random walk over the grid. Any other agent is a *Follower*. The objective of a *Follower* is to mimic the movement of the *Leader*. However, there is a restriction that the *Follower* should keep a minimum distance  $d_{min}$  from the *Leader* and should definitely avoid collisions.

In the following we consider a number of scenarios of increasing complexity to illustrate our points. In particular, the model will be specified in PALOMA [5], a new process algebra that is designed for the modelling of spatially distributed collective and adaptive systems. Before describing the scenarios, we first give a brief introduction to PALOMA. For more details the interested reader is referred to [5].

## 2.1 PALOMA

PALOMA is novel stochastic process algebra that allows the expression of models representing systems comprised of populations of agents distributed over space. In PALOMA each agent is a finite state machine and the language is *conservative* in the sense that no agents are spawned or destroyed during the evolution of a model (although they can cease to change state). The language has a two level grammar:

$$\begin{aligned} X(\ell) &::= !(\alpha, r).X'(\ell') \mid ?(\alpha, p).X'(\ell') \mid X(\ell) + X(\ell) \\ P &::= X(\ell) \mid P \parallel P \end{aligned}$$

Agents are parameterised by a *location*, here denoted by  $\ell$ . Agents can undertake two types of actions, *spontaneous* actions, denoted  $!(\alpha, r)$ , and *induced* actions, denoted  $?( \alpha, p)$ . When an agent performs a spontaneous action, it does so with a given rate  $r$ , which is taken to be the parameter of an exponential distribution, where  $1/r$  is the expected duration of the action. Spontaneous actions are broadcast to the entire system, and can induce change in any other agent which enables an induced action with the matching type  $\alpha$ . An induced action has an associated probability  $p$ , which records the probability that the agent responds to a spontaneous action of the same type. In the style of the Calculus of Broadcasting Systems [11], this can be thought of as the probability that the agent *listens* as opposed to simply *hearing*. Alternative behaviours are represented by the standard choice operator,  $+$ . A choice between spontaneous actions is resolved via the race policy, based on their corresponding rates. We assume that there is never a choice between induced actions of the same type.

A model,  $P$ , consists of a number of agents composed in parallel. There is no direct communication between agents, for example in the style of shared actions in PEPA [7]. Instead, all communication/interaction is via spontaneous/induced actions. When an action is induced in an agent the extent of its impact is specified by a perception function,  $u(\alpha, \ell, X, \ell', X')$  where  $\alpha$  is the action type,  $\ell$  and  $X$  are the location and state of the receiver agent whereas  $\ell'$  and  $X'$  are the location and state of the sender agent. This is a further probability which, given the locations

of the two agents, their current states and action type involved, determines the likelihood that the induced action occurs. For example, the perception function might have value 1 when the two agents are within a communication radius  $r$  of each other, but a value of 0 whenever the distance between them is greater than  $r$ . Obviously this gives a rich set of possible styles of interaction, but note that each agent with an induced action chooses independently whether to respond or not.

## 2.2 Scenario 1: Passive Followers

In this scenario, we assume the *Leader* can either choose to take a rest with rate  $r_{rest}$ , or to move a step along a random direction at the rate of  $r_{mv}$ . Moreover, we use  $p_n, p_s, p_w, p_e$  to represent the probability to move north, south, west and east, respectively. Thus, the *Leader* agent can be described as follows:

$$\begin{aligned} L(x, y) ::= & !(rest, r_{rest}).L(x, y) + !(n, r_{mv}p_n).L(x, y + 1) + \\ & !(s, r_{mv}p_s).L(x, y - 1) + !(w, r_{mv}p_w).L(x - 1, y) + \\ & !(e, r_{mv}p_e).L(x + 1, y) \end{aligned}$$

where  $!(rest, r_{rest})$  denotes the *Leader* taking a rest spontaneously at the rate of  $r_{rest}$ , and when it does this, it remains in its current position.  $!(n, r_{mv}p_n)$  denotes the *Leader* moving a step north by doing an spontaneous action  $n$  at the rate of  $r_{mv} \times p_n$ .

Furthermore, we assume the *Follower* can only move passively when the *Leader* informs it to do so. Thus, we define the *Follower* agent as:

$$\begin{aligned} F(x, y) ::= & ?(n, p_l).F(x, y + 1) + ?(s, p_l).F(x, y - 1) + \\ & ?(w, p_l).F(x - 1, y) + ?(e, p_l).F(x + 1, y) \end{aligned}$$

where the *Follower* agent can move a step in a direction via an induced action, and  $p_l$  encodes the probability for the *Follower* to respond to the *Leader*'s movement action which means that the *Follower* may not obey the command from the *Leader* with probability  $1 - p_l$ .

The perception functions for actions  $n, s, w$  and  $e$  are simply defined as:

$$\begin{aligned} u(n, \ell, X, \ell', X') &= 1 \\ u(s, \ell, X, \ell', X') &= 1 \\ u(w, \ell, X, \ell', X') &= 1 \\ u(e, \ell, X, \ell', X') &= 1 \end{aligned}$$

which means that the *Follower* will definitely perceive the command from the *Leader*.

## 2.3 Scenario 2: Active Followers

In this scenario, we allow the *Follower* to be a little bit smarter. More specifically, we introduce an internal *Clock* agent which allows the *Follower* to move actively instead of just listening to the *Leader*'s command. The *Clock* agent is simply defined as:

$$\begin{aligned} Clock(x, y) ::= & !(cn, r_c).Clock(x, y) + !(cs, r_c).Clock(x, y) + \\ & !(cw, r_c).Clock(x, y) + !(ce, r_c).Clock(x, y) \end{aligned}$$

which means that the *Clock* agent will perform each self-jump action  $cn$ ,  $cs$ ,  $cw$  and  $ce$  spontaneously at the rate of  $r_c$ .

Then, the *Follower* agent becomes:

$$\begin{aligned} F(x, y) ::= & ?(n, p_l).F(x, y + 1) + ?(s, p_l).F(x, y - 1) + \\ & ?(w, p_l).F(x - 1, y) + ?(e, p_l).F(x + 1, y) + \\ & ?(cn, p_c).F(x, y + 1) + ?(cs, p_c).F(x, y - 1) + \\ & ?(cw, p_c).F(x - 1, y) + ?(ce, p_c).F(x + 1, y) \end{aligned}$$

where  $p_c$  encodes the probability for the *Follower* to *listen* to a clock instruction.

Then, we define the associated perception function for actions  $n$  and  $cn$  as:

$$u(n \mid cn, \ell, X, \ell', X') = \begin{cases} 1 & \text{if } ((dist(\ell.x, \ell.y + 1, L.x, L.y) < dist(\ell.x, \ell.y, L.x, L.y) \\ & \wedge dist(\ell.x, \ell.y + 1, L.x, L.y) > d_{min}) \\ & \vee ((dist(\ell.x, \ell.y + 1, L.x, L.y) > dist(\ell.x, \ell.y, L.x, L.y) \\ & \wedge dist(\ell.x, \ell.y, L.x, L.y) < d_{min})) \\ 0 & \text{otherwise} \end{cases}$$

which can be interpreted in the following way: the *Follower* will only perceive the  $n$  or  $cn$  action from the leader or the internal clock if a step north will let the *Follower* become closer to the *Leader* and the distance to the *Leader* is still larger than  $d_{min}$  (Figure 2a), or it will be farther from the *Leader* but the current distance to the *Leader* is less than  $d_{min}$  (Figure 2b). To save space, we will not show the perception functions for other actions as they are defined in a similar way.

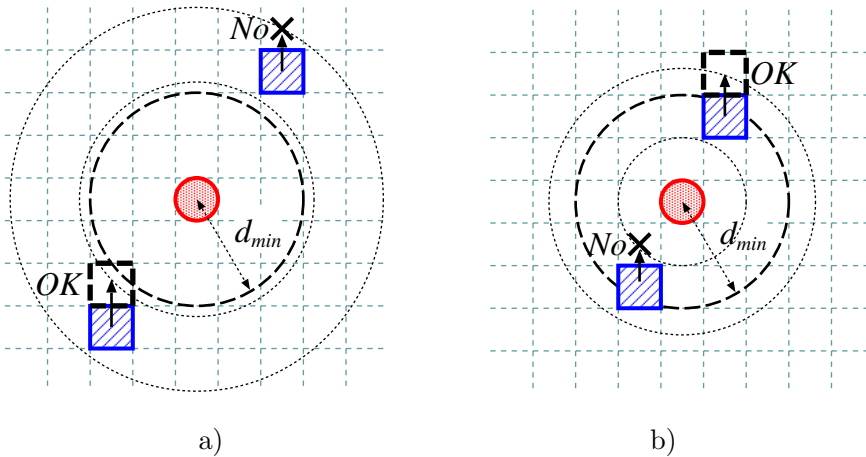


Fig. 2. Behaviour of the follower: a)  $dist(\ell.x, \ell.y + 1, L.x, L.y) > d_{min}$ , b)  $dist(\ell.x, \ell.y + 1, L.x, L.y) < d_{min}$ .

### 2.4 Scenario 3: Multiple Followers

Here, we put multiple *Followers* in the system in order to observe some interesting collective behaviour. We assume that *Followers* always try to avoid bumping into each other. Thus, we add a simple protocol to *Followers* by modifying the perception functions. For example, the perception functions for  $n$  and  $cn$  are modified as follows:

$$u(n \mid cn, \ell, X, \ell', X') = \begin{cases} 1 & \text{if } (|F(\ell.x, \ell.y + 1)| = 0 \wedge \\ & ((\text{dist}(\ell.x, \ell.y + 1, L.x, L.y) < \text{dist}(\ell.x, \ell.y, L.x, L.y) \\ & \wedge \text{dist}(\ell.x, \ell.y + 1, L.x, L.y) > d_{\min}) \\ & \vee ((\text{dist}(\ell.x, \ell.y + 1, L.x, L.y) > \text{dist}(\ell.x, \ell.y, L.x, L.y) \\ & \wedge \text{dist}(\ell.x, \ell.y, L.x, L.y) < d_{\min}))) \\ 0 & \text{otherwise} \end{cases}$$

where  $|F(\ell.x, \ell.y + 1)|$  denotes the number of *Followers* in location  $(\ell.x, \ell.y + 1)$ . This means that a *Follower* will only take a step north when there are no other *Follower* agents in that location. Again, we will not show the modified perception functions for other actions as they are changed in a similar way.

## 3 Performance Measures

Traditionally, performance measures derived from probability distributions can be broadly divided into three categories:

**State-based:** an expectation over the states of the system. In its simplest form this is the probability that a certain property holds (Boolean values attributed to states). Utilisation is an example of this type. But such measures can also be based on more meaningful values for states, such as queue length where the value for each state is the number of customers in a queue. When the probability distribution is the steady state distribution the derived values will be the average values, where at other times they will be transient, based on the transient probability. When spatial information is also represented in the system, the states of interest may be those in which certain spatial conditions are satisfied. Thus we might think of a form of spatial utilisation, the percentage of time that a particular location or set of locations are occupied.

**Rate-based:** an expectation over the rates of the system. Typical examples are throughput, loss probabilities, collision probability etc. Essentially these are also calculated as expectations over the states but the rewards associated with the states are now the rate at which events occur within the given state. Again either the transient or the steady state probability distribution may be used in the calculation of the expectation. Here again spatial conditions may be used to

identify the states of interest. For example, a collision relies on the state condition that two agents are in the same location at the same time.

**Time-based:** an average time, or a probability distribution with respect to time with respect to some behaviour. The classic example is perhaps response time which, via Little’s Law can be expressed in terms of throughput (a rate-based measure) and average number (a state-based measure). For non steady state measures, a passage time calculation will usually be required.

It is reasonable to expect that in spatio-temporal systems we will also be able to define space-based measures, analogous to time-based ones, which are derived from state and rate-based ones.

### 3.1 Performance measures of the Leader-Follower Scenarios

When we come to measuring our leader-follower system there are multiple different dimensions to consider and we may choose to abstract one of more either through projection or by averaging.

- The first dimension is *state*. This is the fundamental record of the behaviour of the system. We assume that the behaviour of the agents is characterised by random variables which range over the state space. In this simple example the agents do not have any logical state beyond their current position. But in general we can imagine that agents are also fulfilling some other role in addition to their motion and so they may have other characterisations of state, orthogonal to their location.
- The second dimension is *time*. In the simplest performance analysis we consider the behaviour of a single system or agent with respect to time. This may be transient or elapsed time, or abstract time, in the sense that consideration of steady state behaviour essentially removes the time dimension by assuming stationarity. In this dimension it makes sense to consider the rate at which events occur, the probability of an event occurring within a time bound, the cumulative probability of events, etc.
- The third dimension for our systems is *space*. Here we do not have an abstraction equivalent to steady state, but we do have the possibility to take average values over all space; for example, this is often done in ecological models. If there are different types of agents competing over space (e.g. predators and prey) the system may be characterised at a certain time by the total number of each type present disregarding spatial placement, even though interaction is location aware. This form of spatial abstraction, does not seem appealing from a performance perspective, but is often carried out as a mathematical expediency as more detailed representation is computationally expensive or intractable.
- The fourth dimension, for a collective system, is the *population* of individuals. Here we again typically make an abstraction by shifting to a count or proportion of individuals exhibiting certain characteristics rather than retaining full information about each individual. Thus we may wish to record the number of agents at

a given location at a given time, or calculate averages either with respect to time or with respect to location. Or we may consider the behaviour only at steady state when time is abstracted.

- Finally, when we analyse our system through discrete event simulation we have a fifth dimension which is the instances or *trajectories* of our system on which measurements are based.

### 3.2 Basic measures

Here we are particularly interested in spatio-temporal properties that incorporate both the second and third dimension. The simplest way of doing this is to consider a measure over one dimension at all points in the other. At the state level we can define for any agent  $A$ ,  $loc(A, t)$  be the location of  $A$  at time  $t$  as the projection of the spatial dimension onto the time dimension. Conversely, we can define  $visit(A, l)$  to be the set of time instants in which agent  $A$  was at location  $l$ . Measures  $loc$  and  $visit$  are complementary in the sense that:

$$t \in visit(A, l) \implies loc(A, t) = l$$

and

$$loc(A, t) = l \implies t \in visit(A, l)$$

Note that there is a big difference in the codomain of the two measures, since  $loc$  returns a single point in space, while  $visit$  is a relation that returns a set of locations. The difficulty lies in the fact that while an agent can only have one location at a time, it can be at that location multiple times. To define a simpler function we can for example restrict to the last visit or the first visit, and define  $firstVisit(A, l) = \min\{visit(A, l)\}$  and  $lastVisit(A, l) = \max\{visit(A, l)\}$ . This however might not be a proper function, since location  $l$  might not be visited by agent  $A$  in the considered scenario. An alternative would be define functions over the relation. For example, we can express the *age* of an agent with respect to a location,

$$age(A, l) = |visit(A, l)|$$

where for a relation  $\mathcal{R}$ ,  $|\mathcal{R}|$  is the size of the relation. In effect, this counts the number of times that agent  $A$  has visited location  $l$ .

We can use simulation to derive a large number  $K$  of trajectories and use it to estimate the probability that any of the previous properties hold. We can regard this as projecting the measure onto the fifth dimension, the space of trajectories. For example, we can express the probability that an agent  $A$  was at location  $l$  at time  $t$  as:

$$\mathbb{P}(A \text{ is in } l@t) = \frac{\sum_{i=1}^K \mathbf{1}(loc(A, t) = l \in traj_i)}{K}$$

and the probability that a location  $l$  is visited by agent  $A$  exactly  $n$  times as:

$$\mathbb{P}(A \text{ visits } l \text{ exactly } n \text{ times}) = \frac{\sum_{i=1}^K \mathbf{1}(age(A, l) = n \in traj_i)}{K}$$



where  $\mathbf{1}(\text{predicate})$  is an indicator function which has the value 1 when the predicate is true and the value 0 otherwise.

Figure 3 shows for every location  $l$  the probability of the leader being there at times  $t = 10, 100, 500s$ . In this case the leader performs a step in one of the four directions on average every 4s ( $r_{mv} = 0.25$ ) and never rests ( $r_{rest} = 0$ ). The four directions are all equally probable ( $p_n = p_e = p_s = p_w = 1/4$ ). The grid is  $50 \times 50$  and the leader starts in position  $(27, 27)$ . The number of simulation runs used is  $K = 200,000$ . Note that the distribution tends to a bivariate normal distribution centered in the initial position of the leader, with the variance that increases with time. This is natural since the leader moves according to a pure random walk in the four directions.

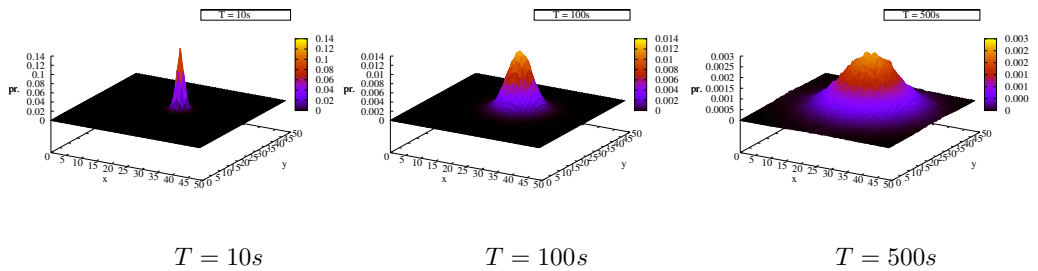


Fig. 3. Cell occupancy probability for the leader at different time instants.

Figure 4 shows the distribution of the *age* for four different locations  $l_1 = (26, 26)$ ,  $l_2 = (28, 28)$ ,  $l_3 = (25, 25)$  and  $l_4 = (24, 24)$ . Since the leader starts at  $(27, 27)$ , locations  $l_1$  and  $l_2$  are at the same distance. Since the movement of the leader is not biased in any direction, they are identical. Locations  $l_3$  and  $l_4$  are at increasing distance: as can be clearly seen, the probability of not entering that location  $n = 0$  increases whilst the probability of entering it a larger number of times decreases with the distance from the initial position.

Let us now focus on the probability that a given location  $l$  is first/last visited by agent  $A$  at time  $t$  as:

$$\mathbb{P}(A \text{ first/last visits } l @ t) = \frac{\sum_{i=1}^K \mathbf{1}([first/last]Visit(A, l) \leq t \in traj_i)}{K}$$

This measure allows us to analyze another peculiarity: temporal distributions, when computed via simulation, are always affected by the finite duration of the considered trace. In particular,  $visit(A, l)$  will always correspond to the set of time instants in which location  $l$  was visited during the time horizon spanned by the simulation. If agent  $A$  will visit location  $l$  after the end of the simulation cycle, this will not be included into  $visit(A, l)$ . Figure 5 shows the probability distribution of the time at which a fixed location  $l = (26, 26)$  is either first visited or last visited. Two different temporal horizon lengths are considered for the simulations. It is interesting to see that the first and last visit to a location tend to coincide at the end of the simulation

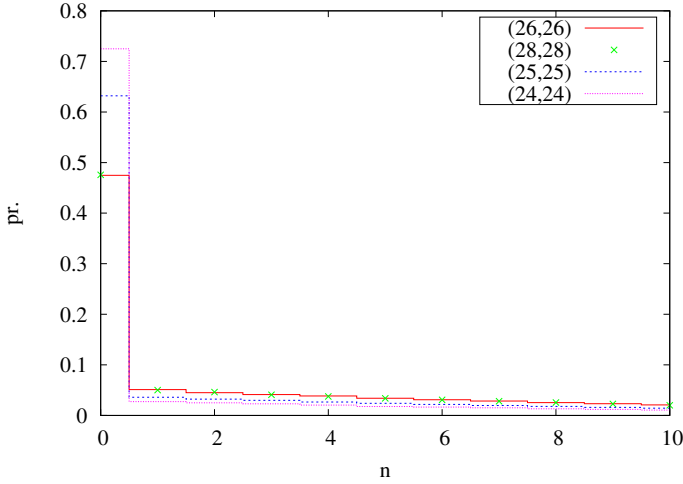


Fig. 4. Distribution of *age*, the number of passages at different positions, for the leader starting at (27,27).

period: this occurs because if the agent passes over the target cell only once during the considered time horizon, we have  $firstVisit(A, l) = lastVisit(A, l)$ . It can also be clearly seen that the distribution of the last passage time is a measure that clearly depends on the simulation interval, while the first passage time is less influenced, as long as the time horizon is large enough to allow the agent to reach the considered location.

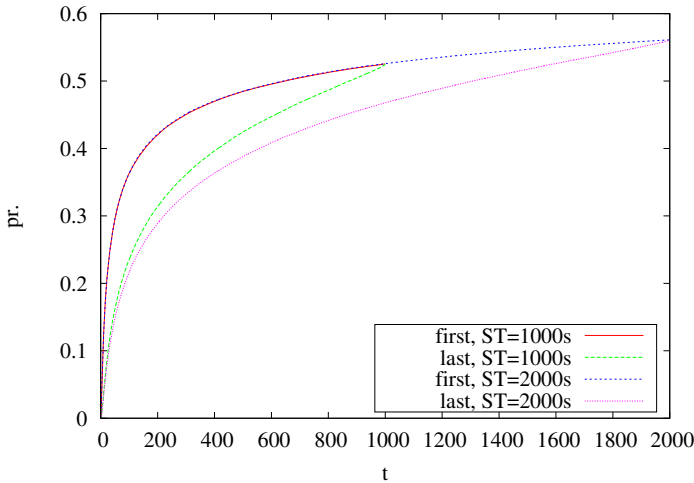


Fig. 5. Distribution of the time of the first and the last visit to location (26,26) for two different simulation horizons  $ST = 1000s$  and  $ST = 2000s$ .

### 3.3 Derived measures

From these basic measures we can construct more interesting ones such as distance, *dist*:

$$dist(A_1, A_2, t) = \|loc(A_1, t) - loc(A_2, t)\|$$

e.g.  $dist(F, L, t)$  is the distance between agent  $F$  and agent  $L$  at time  $t$ . Thus when we have a single follower we can plot  $dist(F, L, t)$  with respect to time to see how the distance between the follower and the leader evolves over time. Again, in a simulation study, the distance can be averaged across all the trajectories to have a global idea of the system behaviour. This is reported in Figure 6 for different behaviours of the leader and of the follower. In particular, both a random movement (Figure 6a) and a fixed route (Figure 6b) for the leader are considered, for different internal clock speed. In all cases, the follower always perceives the clock message, even if this is performed at a different rate ( $p_c = 1$ ). A very noisy channel is considered, with the probability of missing the direction message sent by the leader equal to 95% ( $p_l = 0.05$ ). The target minimum distance between the leader and the follower is 2.1. For the random walk case, the distance from the leader becomes too high only when the follower does not perform any action to catch the leader ( $r_c = 0$ ). When the clock is considered, the follower can always maintain a good distance from the leader for the two considered speeds of  $r_c = 1/16$  and  $r_c = 5/32$ . This is because the random motion confines the leader to an area that is always relatively close to its initial position. For the follower it is enough to perform some infrequent check to catch up with the leader. If instead a fixed route is followed, the messages sent by the leader become of paramount importance. In this case, the follower is not able to catch up with the leader, unless it performs clock actions at a rate much larger than the speed at which the leader is moving.

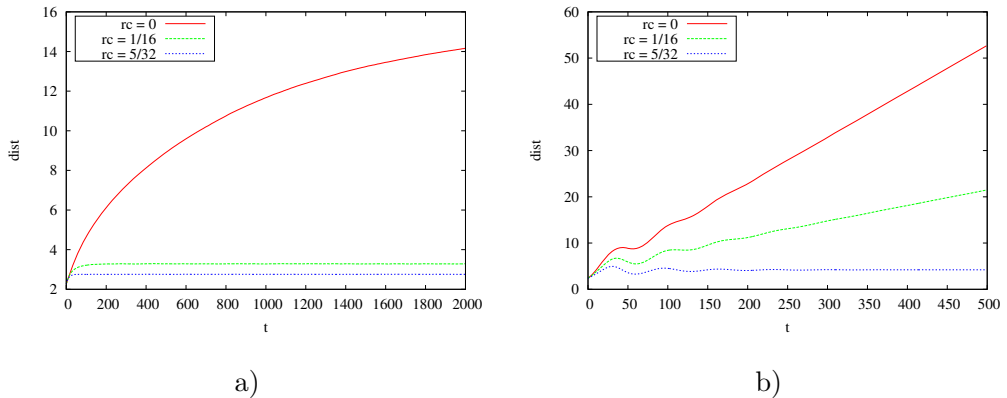


Fig. 6. Average distance between the leader and the follower for different clock rates: a) random walk, b) leader with a predetermined route.

We also assume that we can detect collision. Two agents  $A_1$  and  $A_2$  are said to collide if there exists  $t$  such that  $loc(A_1, t) = loc(A_2, t)$ . In the simple scenarios, 1 and 2, which have only a single follower, we are then interested in the cases where  $loc(F, t) = loc(L, t)$ . Again, we can use simulation to calculate the probability of a collision at a particular time. We can estimate the probability of collision at time  $t$  as

$$\mathbb{P}(\text{collision}@t) = \frac{\sum_{i=1}^K \mathbf{1}(loc(F, t) = loc(L, t) \in \text{tra}_{j_i})}{K}$$

This is shown in Figure 7 for the same cases considered before. For the random

movement, the case in which the follower does not perform any recovery action has a large hit probability at the beginning, and rapidly reduces with time. This however is caused by the fact that the leader tends to go far away from the follower thus reducing the hit probability. This is even more visible in the route based movement case, where the hit probability is zero due to the fact that the follower starts losing the leader from the beginning. In the other cases, it is clear that the hit probability converges to a limit value, that is a characteristic of the follower's behaviour, and depends on his parameters. For the route based movement, this is observed only for the case in which the follower reacts at a very high speed, since that is the only one where it is able not to lose the leader.

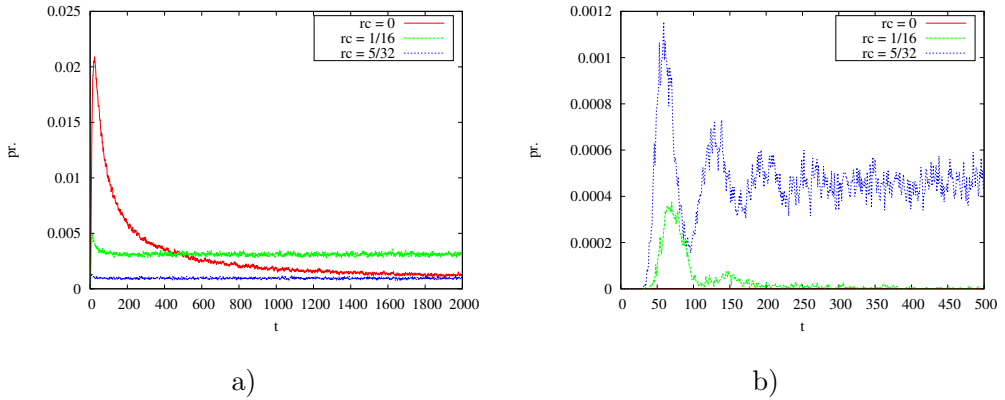


Fig. 7. Hit probability between the leader and the follower for different clock rates: a) random walk, b) leader with a route.

Collision may be generalised to being within the minimal distance  $d_{min}$ :

$$\mathbb{P}(too\ close@t) = \frac{\sum_{i=1}^K \mathbf{1}(dist(F, L, t) < d_{min} \in traj_i)}{K}$$

Figure 8 shows such probabilities for different values of  $d_{min}$  for the case with  $r_c = 5/32$ . As can be seen, while the hit probability can be very low, by enlarging the distance, the probability of the follower being at the considered contour becomes more tangible.

### 3.4 Temporal distance measures

We could also calculate a temporal distance  $tdist$  between the agents at a specific location  $l$ , comparing their first visit times:

$$tdist(A_1, A_2, l) = \begin{cases} |firstVisit(A_1, l) - firstVisit(A_2, l)| & \text{if } age(A_1, l) > 0 \\ & \wedge age(A_2, l) > 0 \\ +\infty & \text{otherwise} \end{cases}$$

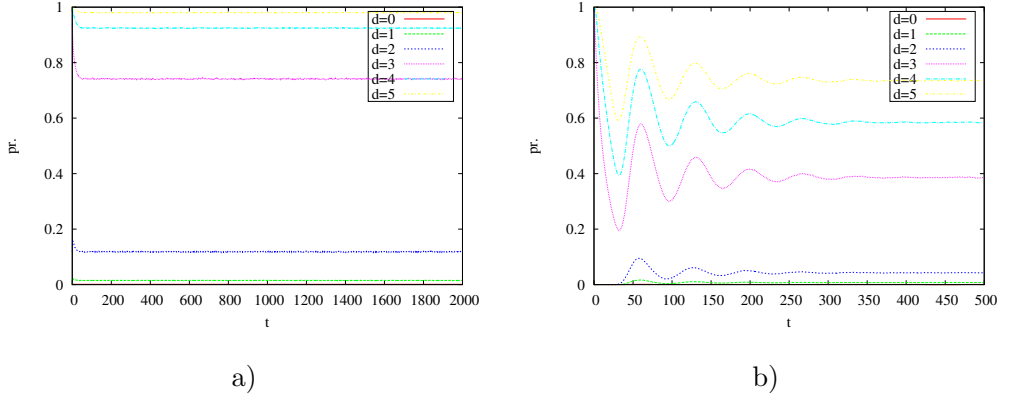


Fig. 8. Probability of the follower being within a given distance from the leader for: a) random walk, b) leader with a route.

Note that considered temporal distance is meaningful only if both agents have visited location  $l$ . We have set the temporal distance to  $+\infty$  if either of the agents has not passed through the considered location.

When we consider trajectories, we can use the previous definition to compute characteristics with respect to space rather than time. For example, we can define the probability that the passage of two agents in a location  $l$  is too close in time as:

$$\mathbb{P}(\text{too close}@l) = \frac{\sum_{i=1}^K \mathbf{1}(t\text{dist}(F, L, l) < t_{\min} \in \text{traj}_i)}{K}$$

where  $t_{\min}$  is some minimal separation in time that we seek to enforce. Note that by definition of  $t\text{dist}$ , this probability will be zero if the both agents have not passed through  $l$ . Figure 9 shows the probability that the temporal distance is less than  $t_{\min} = 5s$ ,  $t_{\min} = 10s$  and  $t_{\min} = 15s$ , for the case with  $r_c = 5/32$  and random movement for all the locations  $l = (x, y) \in [21, 31] \times [21, 31]$ . Although the maximum is in all cases for the initial location of the leader  $l = (27, 27)$ , the shapes tend to be less symmetric as the threshold increases. This reflects the fact that the protocol tends to keep the initial displacement between the leader and the follower.

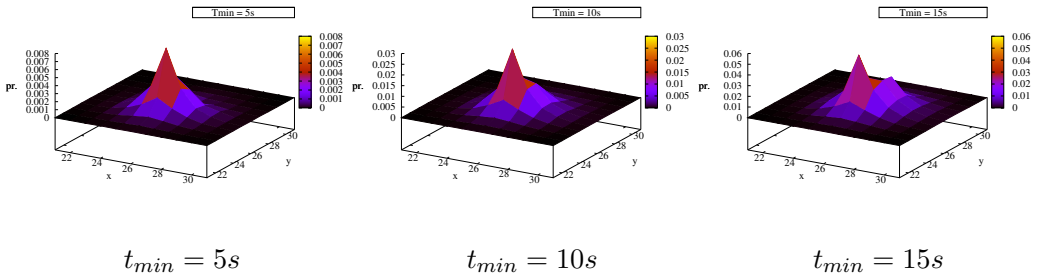


Fig. 9. Probability that the temporal distance for the cells in the range  $[21, 31] \times [21, 31]$  is less than the considered thresholds  $t_{\min}$ .

### 3.5 Multiple followers

When we have multiple followers we can think of defining the same measures for each of the followers, but what we would really like is some measure that reflects the collective behaviour.

Thus thinking of the distance from the leader, when there are  $N$  followers we can define  $\widehat{dist}$  as the average distance at time  $t$  as follows:

$$\widehat{dist}(L, t) = \frac{1}{N} \sum_{i=1}^N dist(F_i, L, t)$$

The problem with this is that it can have the same value for very different distributions of followers over space.

Instead we might think of some form of *contours* recording the number, or proportion of agents that are within increasing distances from a given location at a given time. Thus at time  $t$ , if there are  $N$  agents, the contour  $C(l, 1, t)$  would be defined as

$$C(l, 1, t) = \frac{1}{N} \sum_{i=1}^N \mathbf{1}(\|loc(A_i, t) - l\| \leq 1)$$

and in general

$$C(l, n, t) = \frac{1}{N} \sum_{i=1}^N \mathbf{1}(\|loc(A_i, t) - l\| \leq n)$$

In the specific case of the followers and the leader we can define

$$C(loc(L, t), n, t) = \frac{1}{N} \sum_{i=1}^N \mathbf{1}(\|loc(F_i, t) - loc(L, t)\| \leq n)$$

Figure 10 shows the contour values with 4 and 8 followers. As can be seen from the figure, in the simulation with 4 followers, followers are less likely to break the minimum distance but more likely to keep a good distance from the leader than in the simulation with 8 followers. This is because followers always try to avoid bumping into each other. Thus with more followers, the probability of not perceiving the movement action is also higher (see the perception function in Section 2.4).

Analogously we can think of time contours,

$$C(t, \delta, l) = \frac{1}{N} \sum_{i=1}^N \sum_{j=0}^{\delta} \mathbf{1}(loc(A_i, t + j) = l)$$

or

$$C(t, \delta, l) = \frac{1}{N} \sum_{i=1}^N \mathbf{1}(loc(A_i, t + \delta) = l)$$

depending on whether we consider the time contours cumulatively or not.

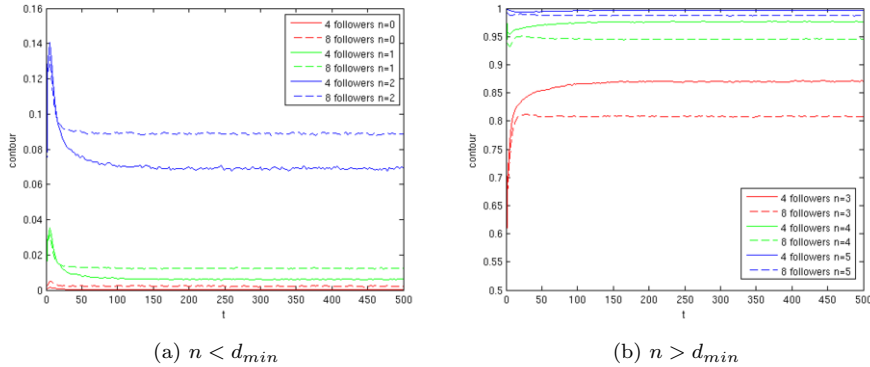


Fig. 10. The contour  $C(\text{loc}(L, t), n, t)$  with different number of followers, where  $d_{\min} = 2.1$

## 4 Conclusions and future work

In this paper we have made an initial study of the types of measures that it can be interesting to study in systems in which there are both temporal and spatial aspects of behaviour. These are important characteristics in many collective adaptive systems (CAS), which are geographically distributed systems comprised of interacting but autonomous agents. We have illustrated the ideas with a simple leader-follower system studied in a number of different scenarios via simulation.

Whilst our example system is simple, it is sufficient to highlight the rich forms of information that can be derived from models of CAS, and it is easy to see how the measures we investigated could be adapted to real-life systems. For example, smart transportation systems are examples of CAS [12], where regulatory requirements impose spatial-temporal conditions. Bus operators are subject to the *headway requirement* on frequent routes, which can be regarded as a special case of our leader-follower scenario. Here the timetable would play the role of the leader whilst buses providing the service are the followers. The headway requirement imposes conditions on the spatial and temporal separation of the followers in order to ensure that there is a regular service for the users. In this case the behaviour of the leader is deterministic but the behaviour of the followers, the buses, is subject to stochastic factors, such as traffic and weather conditions as well as human interaction.

In future work we will investigate our identified measures further to see how well they match to the user and operator performance requirements for CAS. In the current work we have worked from first principles, assessing the data available from our simple scenario and the spatio-temporal measures that can be built. An alternative approach would be to work with a spatio-temporal logic to define properties of interest. The use of temporal logic in the context of Markovian-based performance models is well-established [2] and supported by tools such as PRISM [8]. Spatial Logics have also been studied for many years [1] but to the best of our knowledge has yet to be applied in the quantitative context of CTMCs, although recent applications include data verification for CAS [4]. There is little formal treatment of the combination of spatial and temporal logic although it has been considered in an informal way in the analysis of video sequences [3]. Here spatial until formulae

were interleaved with temporal until formulae to express conditions on the relative positions of objects in an image as time progressed. In quantified spatio-temporal logic we would seek to attribute a value to such properties, just as probabilities are associated to temporal properties expressed in CSL.

## References

- [1] Aiello, M., I. Pratt-Hartmann and J. van Benthem, editors, “Handbook of Spatial Logics,” Springer, 2007.
- [2] Baier, C., B. R. Haverkort, H. Hermanns and J.-P. Katoen, *Model-checking algorithms for continuous-time markov chains*, IEEE Trans. Software Eng. **29** (2003), pp. 524–541.
- [3] Bimbo, A. D., E. Vicario and D. Zingoni, *Symbolic description and visual querying of image sequences using spatio-temporal logic*, IEEE Trans. Knowl. Data Eng. **7** (1995), pp. 609–622.
- [4] Ciancia, V., S. Gilmore, D. Latella, M. Loreti and M. Massink, *Data verification for collective adaptive systems: spatial model-checking of vehicle location data*, in: *Proceedings of the 2nd FoCAS Workshop on Fundamentals of Collective Adaptive Systems*, London, England, 2014, to appear.
- [5] Feng, C. and J. Hillston, *PALOMA: A process algebra for located markovian agents*, 11th International Conference on the Quantitative Evaluation of Systems (2014), to appear.
- [6] FET Proactive FOCAS Project 600708, *QUANTICOL: A quantitative approach to management and design of collective and adaptive behaviours*, [www.quanticol.eu](http://www.quanticol.eu).
- [7] Hillston, J., “A Compositional Approach to Performance Modelling,” CUP, 2005.
- [8] Kwiatkowska, M. Z., G. Norman and D. Parker, *PRISM 4.0: Verification of probabilistic real-time systems*, in: *Computer Aided Verification - 23rd International Conference, CAV 2011, Snowbird, UT, USA, July 14-20, 2011. Proceedings*, Lecture Notes in Computer Science **6806** (2011), pp. 585–591.
- [9] Latella, D., M. Loreti, M. Massink and V. Senni, *Stochastically timed predicate-based communication primitives for autonomic computing*, Technical report, QUANTICOL project (2014).
- [10] Milner, R., “The Space and Motion of Communicating Agents,” Cambridge University Press, 2009.
- [11] Prasad, K., *A calculus of broadcasting systems*, Science of Computer Programming **25** (1995), pp. 285–327.
- [12] Reijbergen, D. and S. Gilmore, *Formal punctuality analysis of frequent bus services using headway data*, in: *Proceedings of the 11th European Performance Engineering Workshop*, Florence, Italy, 2014, to appear.
- [13] Weiser, M., *The computer for the 21st century*, Scientific American **265** (1991), pp. 94–104.