



## Deliverable3.4: Synchronization

Status and Version:	Version 1.0	
Date of issue:	19.12.2007	
Distribution:	Project Internal	
Author(s):	Name	Partner
	Paul Marrow	BT
	José Halloy	ULB
	Fabrice Saffre	BT
Checked by:	Antonio Manzalini	TI



## Table of Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	Purpose and Scope	4
1.2	Document History	4
<b>2</b>	<b>Related work/state of the art</b>	<b>4</b>
2.1	Introduction	4
2.2	Properties of oscillatory systems	5
2.2.1	Amplitude	5
2.2.2	Phase	5
2.2.3	Frequency	6
2.2.4	Period	6
2.2.5	Oscillation and synchronization	6
2.3	Physical models of oscillators	6
2.3.1	Reaction-diffusion	6
2.3.2	Chaotic dynamics	7
2.4	Biological inspiration for models of synchronization	7
2.4.1	Synchronization in natural neural networks	8
2.4.2	Synchronization in biological development	8
2.4.3	Biological group behaviour	8
2.4.3.1	Predator-avoidance behavior	9
2.4.3.2	Synchronized insect emergence	9
2.4.3.3	Foraging behaviour	10
2.4.3.4	Group behaviour in extreme environments	10
2.4.3.5	Mating behaviour	11
2.4.3.6	Lekking	11
2.4.3.7	Firefly signalling	12
2.5	Biologically-inspired models of synchronization	12
2.5.1	Biological pacemaker-based models	12
2.5.2	Artificial neural network models	13
2.5.3	Firefly-inspired models	13
2.6	General issues surrounding synchronization in computer systems	14
2.7	Implications for CASCADAS	14
<b>3</b>	<b>Synchronization through pattern-matching (“type 1”)</b>	<b>15</b>
3.1	Introduction:	15
3.2	Minimum compliant overlay	15
3.3	Realistic overlay creation/maintenance	18
3.3.1	Random activity pattern	18
3.3.2	Time zone-defined activity pattern	20
3.3.3	Regular discrepancies in time zone-defined activity patterns	22



3.3.4	Movement between time zones	25
<b>4</b>	<b>Synchronization through individual adjustments (“type 2”)</b>	<b>27</b>
4.1	Minimal model based on imitation	27
4.2	Rhythm within a single population	30
4.3	The case of two populations out of phase	34
4.4	Summary	37
<b>5</b>	<b>Discussion and conclusions</b>	<b>38</b>
<b>6</b>	<b>References</b>	<b>38</b>



## 1 Introduction

### 1.1 Purpose and Scope

This document reports on work carried out in Work Package 3 of the CASCADAS project (Self-Organized Component Aggregation and Emergent System Properties) on algorithms using synchronization in self-organisation.

The document is laid out as follows. Section 2 reviews the state of the art in related work. Section 3 reports on simulations exploring a particular class of simulation algorithm and their potential for application to networks of ACEs. Section 4 explores mathematical models of synchronization and their implication for applications of ACEs. Finally, section 5 draws conclusions about the relevance of synchronization to self-organisation in distributed systems based on autonomous components as developed elsewhere in the CASCADAS project.

### 1.2 Document History

Version	Date	Authors	Comment
0.01	24/10/07	Paul Marrow	Initial document
0.02	07/11/07	Paul Marrow	Basic content
0.03	14/11/07	Paul Marrow	Integrating from previous document
0.04	29/11/07	Paul Marrow	Further information
0.05	14/12/07	Fabrice Saffre	First complete draft
1.0	19/12/07	Fabrice Saffre	First release of complete document

## 2 Related work/state of the art

### 2.1 Introduction

Research into the consequences of synchronization for self-organizing systems has a long history, stretching back to the observation of the properties of oscillating pendulums by Huygens in the 17<sup>th</sup> century (cited by [1], original sources not available). More recently research has taken several directions:

- Physical models of oscillators
- Mathematical models of biological oscillators
- Investigation of the biological basis for synchronized behaviour



Each of these areas may be relevant in identifying how to exploit synchronization in autonomic and situation-aware dynamically adaptable services as provided by interacting ACEs being developed in the CASCADAS project.

## 2.2 Properties of oscillatory systems

Much of the basis of studies of synchronizing systems can be seen to derive from studies of oscillation in very simple, regular waveforms, especially sinusoidal waves. Any system based on oscillating waves has a number of features which can be easily compared with other oscillating systems. In no particular order of relevance we can compare

### 2.2.1 Amplitude

This is the magnitude of oscillation of a wave (see Figure 1).

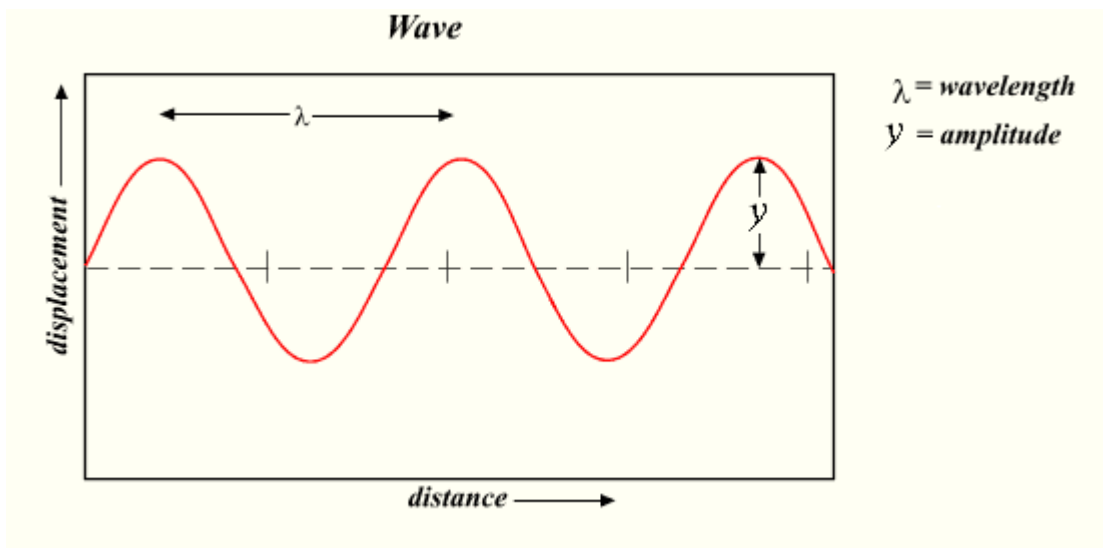


Figure 1. Amplitude and other properties of a regular waveform. From [2].

If it is a regular waveform, this will be a fixed value. If two waves combine in a synchronization process “amplitude-locking” may be an aspect of synchronization if the amplitude of the combined waveform is a sum or product of the amplitudes of the two original waves and some sort of threshold is required before a synchronizing activity takes place. But these are many assumptions which depend upon other properties of waveforms, so amplitude may not be essential for synchronization, especially, as in some of the more complex waveforms observed in nature, amplitude varies dramatically and unpredictably over time.

### 2.2.2 Phase

Phase is the degree argument of the angle of oscillation of a wave function from its initial condition. In contrast to amplitude-locking, if two waves reach “phase-locking” this is a strong basis for synchronization because it means the two waves have reached the same “shape” at the same time, and if synchronization involves particular behaviours during



particular stages of the oscillation these behaviours will be simultaneous. However, this is not a guarantee for synchronization, it will work where the properties of waves are regular and constant, and where the two waveforms have properties that allow phase-locking, which in many cases may be impossible because of other properties of the systems (such as frequency, see below.)

### **2.2.3 Frequency**

Frequency is the number of occurrences of a repeated event during a unit of time. In the case of an oscillatory system, frequency measures the number of oscillations during a time unit (or time interval). For a sinusoidal wave, this is equivalent to rate of change of phase. “Frequency-locking” may be highly relevant to synchronization because it allows periodic events to coincide, and thus evidence of synchronization may become visible from outside the oscillating system. But it does not guarantee synchronization, because other properties of oscillatory systems need to be similar as well.

### **2.2.4 Period**

Period is the inverse of frequency: the time period in which a given number of oscillations take place. For a regular waveform this will be fixed, but it may vary for a more complex waveform. If two oscillators can converge their periods, they may be able to synchronize, but because there are such a lot of other properties of oscillatory systems that may vary, this is not certain.

### **2.2.5 Oscillation and synchronization**

Oscillation can be well-defined, but does this always lead to useful understanding of synchronization? If we have an oscillatory system consisting of multiple oscillators which are already in phase and have identical amplitude, then it may be argued that we are already observing a synchronized system. But insight from this may be very difficult to apply to irregular real-world systems, because how do you move from irregular, disorganized, possibly random, initial states, to an organized, synchronized state. In the following sections, a number of models and real-world examples of oscillatory systems are discussed, which show aspects of synchronization but may not always conform to the requirements needed for transition from oscillation to complete synchronization, or be sufficiently well understood to be transferable to computational applications.

## **2.3 Physical models of oscillators**

Theoretical physics provides a sound basis for the study of synchronization, but it also generates a great deal of complexity from apparently simple beginnings. It also provides examples of convergence of oscillatory systems (such as phase-locking) that may appear to imply synchronization but may not be sufficient.

### **2.3.1 Reaction-diffusion**

Reaction-diffusion is a phenomenon found in a number of chemical and biochemical/biological systems (e.g. Murray [3]), leading to the persistence of stable states in apparently unstable reacting systems. Consider particles diffusing in three dimensions, which might represent physical particles, chemicals, biochemicals or biological entities. If there are more than one type then they may be reacting with each other, and this is indeed what is observed in many chemical or biochemical contexts. Reaction-diffusion models



take into account both the apparently random diffusion and the reaction between different elements. This may lead to very different outcomes than if there was no interaction. Murray describes a generalized reaction diffusion system by

$$\frac{\partial u}{\partial t} = f + \nabla \cdot (D \nabla u) \quad (\text{Equation 2.1})$$

where  $f$  is a vector and  $u$  should also be represented in vector form. This describes a broad class of reaction diffusion systems, but assumptions have to be made before such models can be applied to more realistic systems.

For example, Ermentrout and Troy [4] looked at “phase-locking” in a reaction-diffusion system with a linear frequency gradient (see equations 2.1 and 2.2).

$$\frac{\partial z}{\partial t} = z \left( 1 + i\omega(x) - z\bar{z} \right) + d \frac{\partial^2 z}{\partial x^2} \quad \text{where } 0 < x < 1 \quad (\text{Equation 2.2})$$

$$d \frac{\partial z}{\partial x} = 0 \quad \text{where } x = 0, 1 \quad (\text{Equation 2.3})$$

Oscillations can persist and it is shown in [4] in their models that the persistence of such oscillations could include phase-locking. Although this could include synchronization of a waveform it does not necessarily imply synchronization of the amplitude of the wave so may not satisfy all the conditions required for synchronization in more general systems.

Other reaction-diffusion models and their consequences are discussed further in [3].

### 2.3.2 Chaotic dynamics

Rosenblum et al. [1] consider a complex form of dynamics in the context of chaotic oscillators. Despite the occurrence of chaotic dynamics phase-locking can also occur while the amplitude of the chaotic oscillators remains uncorrelated and chaotic. There is also a weaker form of synchronization possible where frequencies are locked and phase differences oscillate in a random walk. Arguably both these forms of synchronization do not satisfy all the requirements of synchronization for applications that depend on it, or for natural systems, because of the unpredictability of the underlying dynamics.

The limitations of these physical models of synchronizing oscillators is that they cannot account for all the complexity found in natural systems that also exhibit aspects of synchronization.

## 2.4 Biological inspiration for models of synchronization

The biological connection to synchronization was initially observed by Winfree in his 1967 paper [5]; his productive investigations in this area led to several books and many other papers.

Winfree initially observed that many biological scenarios could be described as consisting of (nearly) identical oscillators that were coupled in some way. The nature of the oscillation might vary, but for a wide range of forms of oscillations, providing that coupling existed



between the oscillators, behaviour could converge from random or at least disorganised oscillations to more regular rhythms, in some cases complete synchronization.

The different biological scenarios where these conditions could at least approximately, given the nature of biological systems, be satisfied, varied enormously.

### 2.4.1 Synchronization in natural neural networks

Living organisms' nervous systems have provided a tremendous inspiration for computational and mathematical modeling in general in the form of (artificial) neural network models. The initial word “artificial” is often omitted but biological neural networks are more complicated than is usually assumed in artificial neural network models. Nevertheless some interesting properties of EEG studies from neural networks in the human brain studied by Koppel et al. [6] show synchronization of neural rhythms even in these very complicated structures. Koppel et al. look at  $\gamma$ - and  $\beta$ -rhythms in the human hippocampus, and find that both these forms of neural activity in the human brain can show synchronization, although by different means.  $\gamma$ -rhythms do this through local activity, while  $\beta$ -rhythms exhibit more long-distance synchronization.

### 2.4.2 Synchronization in biological development

The slime mold *Dictyostelium discoideum* attracts much attention in developmental biology: in conditions of abundant food supply in its soil habitat it exists as a unicellular amoeba. If starved of food amoebae combine into a multicellular “slug”. The fact that this occurs is of interest from a biological perspective in itself, but the mechanisms behind it are of wider interest. Nagano [7] reviews the multicellular aggregation process that occurs in *Dictyostelium*, and discusses the modeling of this process.

The widely studied non-linear Belousov-Zhabotinskii reaction (reviewed in [3] ch. 7) has been observed in *Dictyostelium* development, and has provided a basis for the understanding of the aggregation process of individual amoebae. There is also a question of how do separate cells synchronize their movement so as to carry out this aggregation.

The secretion of cyclic AMP (cAMP) into the external environment has been suggested as a trigger, but it cannot be very specific if many cells are secreting it at once. Research into this issue, reviewed in [7], suggests that modified receptors lead to variation in feedback to cAMP, leading to spirals in biochemical density in the local environment around an amoeba, causing convergence from many cells secreting at different levels to secretion at uniform levels, and resulting movement towards each other. This has been supported by modeling and by experimental studies. It has been argued that models of this form, although not depending upon the same biological basis, can explain other types of biological synchronization.

For applications of synchronization in distributed computer systems, because this is an extremely well-understood system, it may be good in situations where some form of broad spread of messaging in local environments is allowed. But the increase in messaging costs as messages spread further afield, and the lack of gain from convergence to multicellularity or the equivalent in a distributed system, makes it less likely to be useful in a wider context.

### 2.4.3 Biological group behaviour

One thing that has been emphasized in mathematical and physical models of synchronization is the importance of cooperation between the oscillators or nodes in





generating the synchronized conclusion from a less-organised start. Since an important aspect of animal behaviour, and to a lesser but not irrelevant degree the life of other organisms is cooperation, group living may also be a relevant area to consider when thinking about how to model synchronization in distributed autonomous systems.

Krause and Ruxton [8] and Couzin and Krause [9] consider the many benefits to animals of group living. While this does not always relate to synchronization, because it involves the coordination of behaviour of individuals who might otherwise behave in a disorganized way, it is likely that inspiration for applications using synchronization can be drawn from it.

### 2.4.3.1 Predator-avoidance behavior

An important aspect of animal group living is the enhanced ability to defend against predators. Krause and Ruxton [8] identify several different forms which this could take.

- The “many eyes theory” suggests that synchronization is useful because many prey individuals can watch for predators at once, thus being able to cover more ground and being more likely to take avoidance behaviour at the right time.
- The “dilution of risk” model assumes that even in a large group of prey it is impossible for all to get eaten because of the sheer numbers. If a large group is present at once, the chances are that the one(s) that get eaten will not be the single individual that would be exposed to a high risk of predation if on its own.
- Finally the “predator confusion” model which relates to both the above ideas assumes that even when presented with an abundance of potential prey it is difficult for predators to decide which prey to consume. There is considerable evidence from the shoaling behaviour of fish subject to predation that their motion is both designed to, and does in fact, confuse predatory organisms such as dolphins.

While this provides strong empirical evidence that the synchronization of behaviour among animals has benefits, the multiple stimuli leading to the behaviour in different organisms make it difficult to translate it into simple models that can be implemented in distributed computer systems where the synchronization of node behaviour may have considerable benefit.

### 2.4.3.2 Synchronized insect emergence

However a specific example of a prey species which does exhibit synchronized emergence and yet persists in the face of continuing predation, the 13 and 17 year cicadas (*Magicalcada*) can be explained by a relatively simple model which was first presented by Hoppensteadt and Keller [10] and is reviewed by Murray [3]. This model is based upon assumptions about predation and limitations on carrying capacity for the underground nymphs and the emergent cicadae. These insects live for only a short time while adults, during which they lay eggs which hatch rapidly as nymphs and attach themselves to nearby tree roots for nutrition.

Given a lifespan of  $k$  years, after which the cicadae die, and a fraction  $\mu$  of the nymphs which survive each year after predation and lack of resources, then  $n_{t-k}$  is the number of nymphs which become established under the ground in year  $t-k$  and

$$\mu^k n_{t-k}$$



will survive  $k$  years and emerge as adults to start the cycle again. Omitting a number of intermediate steps, and introducing a constant  $f$  that is a measure of the fertility of hatched nymphs established underground, as well as constants  $1 > \nu > 0$  and  $a > 0$  we can describe the predation on the emergent cicada population at time  $t$  as

$$p_t = \nu p_{t-1} + a\mu^k n_{t-k-1} \quad (\text{Equation 2.4}).$$

Given this information we can calculate the conditions for a stable balance between the number of nymphs under the ground  $n_t$  and predation on emergent cicada  $p_t$  as

$$T = \frac{af\mu^k}{(f\mu^k - 1)(1 - \nu)} < 1 \quad (\text{Equation 2.5}).$$

We can also calculate the conditions for a synchronized solution in which nymphs emerge as cicadas every  $k$ th year as

$$S = \frac{af\nu^{k-1}\mu^k}{(f\mu^k - 1)(1 - \nu^k)} < 1 \quad (\text{Equation 2.6}).$$

If  $T < 1$  a stable balanced steady state can exist, but it is also the case that  $0 < S < T < 1$  and so a synchronized solution can exist. If  $S > 1$  then  $T > 1$  and neither a steady state nor a synchronized solution can exist. If  $S < 1 < T$  no steady state exists but a synchronized one can, so an initially unsynchronized state can converge to synchronization despite the pressures of predation and resource limitation.

Hence this model explains what is at first a puzzling as well as agriculturally very damaging persistent ecological phenomenon can persist, but it can also provide the basis for the emergence of synchronization among a population of nodes in a computational network.

### 2.4.3.3 Foraging behaviour

In contrast to predator-avoidance behaviour is foraging behaviour. Many foragers are herbivores, and base their foraging around diurnal stimuli of the availability of their forage and predator-avoidance behaviour as above. But for predators foraging includes search for prey. When prey has been identified, synchronization of predator behaviour into groups may be particularly relevant, given that other predators act as scavengers, typically not killing prey directly, but seeking to obtain it from other predators. Fanshawe and FitzGibbon [11] showed that African wild dogs could defend their prey against scavenging hyenas when acting in groups rather than as individuals.

As in the case of predator-avoidance behaviour, the number of different stimuli involved in generating synchronization among many individuals that are foraging, either as herbivores or as carnivores/omnivores or scavengers, makes it difficult to reduce to simple models which can then be applied.

### 2.4.3.4 Group behaviour in extreme environments

Animals can benefit from synchronizing behaviour in groups when their environment becomes sufficiently extreme so as to cause them harm if they remained an isolated



individual. For example huddling of birds in cold environments reduces heat loss due to reduction of surface area of the overall flock exposed to the external environment. This has been proposed as an argument for penguin flocking in Antarctica and for roosting in winter among birds in temperate climates. Models exist which support this (reviewed in [8]), but the complexity of the factors affecting the social behaviour that leads to this makes it difficult to translate into computationally applicable ones. Similar models have been proposed for mammals huddling in burrows.

#### 2.4.3.5 Mating behaviour

It is a blatantly obvious statement that mating behaviour involving sexual reproduction both in animals and in other groups of organisms such as plants and fungi involves an element of synchronization. Indeed one of the earliest examples of synchronization reported in the scientific literature, but not followed up with detailed investigation, was the synchronization of menstrual cycles among women sharing accommodation.

Synchronization in the context of mating behaviour is required so that particular sexes are able to mate at particular times (in different species, the constraints on times when one or other sex can mate vary from extremely strict to non-existent), and so that individuals carry out particular behaviours at appropriate times both prior to mating itself and in mating.

An example of a species that requires a very restricted time to reproduce is the reef-building coral *Acropora millepora*. Since it is sessile multicellular individuals must spawn almost simultaneously in order to ensure reproduction. It turns that this is mediated by receptors that sense moonlight [12]. Since moonlight varies periodically according to the orbit of the Earth around the Sun and the gravitational locking of the Moon's orbit around the Earth, this provides a periodic source which can be detected all over the regions where this species of coral occurs. This could inspire a broadcast-based control of synchronization in a networked computer system, but such an application would arguably be very wasteful of resources in message passing.

The behaviours prior to mating may include displaying to others of the same sex in order to overcome a challenge of access to mates of the opposite sex (for simplification ignoring the possibility of organisms such as some fungi where in effect very large numbers of sexes are possible.); or displaying to others of the opposite sex in order to demonstrate the potential as a possible mate. For more information see Maynard Smith and Harper [13].

#### 2.4.3.6 Lekking

Lekking is an example of synchronization in mating behaviour. Leks (Sutherland 1996 [14]) are locations where male and female birds gather to display to each other and to breed. A high degree of synchronization is required in order to be present at the right location at the right time, and to display at the right time in order to compete effectively with others of the same sex. Males display, and as a result of this display, females choose mates, and mating takes place.

The advantages of this synchronised mating, other than the general advantages of living in groups discussed above, are difficult to establish, since the species involved do not have any physiological or anatomical limitations in mating as separate pairs. A suggestion is that they allow females to better assess the range of male quality available, in effect browsing the male population. A lek may also reduce the level of harassment of females by males seeking mates, because of the large number of other individuals present of both sexes.



Other suggestions have been made (summarised in [8], pp. 32-35), one that may be relevant for inspiring synchronization algorithms in distributed systems is that a lek may involve a queue for future reproduction (Kokko et al. 1999 [15]). Individuals may attend that have not yet reached sufficient quality to breed in order to establish a place in a queue which they will use to reach an opportunity to breed in the future. If different sexes present at a lek are translated into types that need to combine in order to deliver services at particular nodes, maybe such a queue can enforce quality of the service.

Such behaviour provides another empirical example of synchronization in animal behaviour, but is extremely complicated to translate into a form which could be applicable to the synchronization of networked nodes or ACEs in a CASCADAS application, because of the many different factors influencing the birds' behaviour. This is a continuing area of research.

#### **2.4.3.7 Firefly signalling**

Perhaps a better example relating to mating behaviour is that which has inspired many mathematical and computational modelers of synchronization, namely synchronizing display of flashing lights by male fireflies. A number of different species of firefly display this during the mating season, and it is thought to be a component of signaling breeding capability to females and out-competing other males during a tightly-constrained breeding season. Mirollo and Strogatz [16] suggest that Buck and Buck (1976 [17]) were the first to suggest that this synchronized display could be represented by models of simple oscillators. This has developed into a substantial body of literature, upon which computational models of the control of self-organising distributed systems can be built. The contrast with many of the above empirical examples of synchronization in the natural world is perhaps that male fireflies are carrying out relatively simple behaviour, over well-defined periods of time, and thus description of that behaviour in the form of well-defined functions seems a natural way to proceed. This can then lead to algorithms that can be implemented to synchronize distributed systems (see below).

### **2.5 Biologically-inspired models of synchronization**

Given the above physical basis and biological inspiration, there has been active study of mathematical models inspired by biological systems.

#### **2.5.1 Biological pacemaker-based models**

Mirollo and Strogatz [16] modeled oscillators inspired by cardiac pacemaker cells. Cardiac pacemaker cells are only one example of natural pacemakers. There are also natural neural networks that lead to circadian pacemakers, and other regular oscillatory behaviour in the hippocampus (see 2.4.1 above).

Mirollo and Strogatz modeled coupled biological oscillators where almost all initial conditions lead to situations where oscillators are firing simultaneously. This synchronization takes place through cooperation between oscillators, and they argue, is analogous to phase transitions in statistical physics.

It is interesting that they identify three different types of synchronization, which they call

1. synchrony
2. phase-locking



### 3. frequency-locking

Synchrony in their terms refers to multiple oscillators with identical behaviour carrying out this behaviour in unison. They argue that the complexity of natural systems means that this is never actually achieved in natural systems, but in some it approaches that. For example they describe examples from natural firefly populations, often used as inspiration for models (see further below) where the spread in firing times is extremely small compared to the period of oscillation. Clearly this is very similar to the type of synchronization we are looking to implement in distributed computer systems.

In some of the models they study they observe phase-locking, which they define as a situation where oscillators do not fire at the same time, but the phase difference between oscillators is constant and usually non-zero. In an observed system this could result in a repeated step-wise sequence of different firings. This regularity could be useful in progress towards emergent behaviour, but does not appear as well-defined as synchrony above.

Finally, they refer to frequency-locking, where oscillators fire with the same average frequency but with no fixed phase relationship. This seems like a much weaker relationship from which it would be difficult to obtain synchronization.

## 2.5.2 Artificial neural network models

Given the complexity of behaviour of natural neural networks, it is not surprising that artificial neural network models, which originally arose from a set of assumptions simplified from the features of real neural interaction, have been used to study synchronization.

Goel and Ermentrout [18] studied synchrony and stability in a variety of oscillators modeled through artificial neural networks. Specifically they looked at pulse-coupled artificial neural networks and tried different firing patterns to look at the emergence of different patterns and the stability and synchronization or otherwise of those patterns. They discovered the persistence of waves in rings, and rotating waves in two-dimensional lattices, and referred to two forms of “weak” and “strong” coupling. But given the special assumptions of the models upon which these findings were based, although a support for synchronization could be found, whether one could generalize from it is unclear.

## 2.5.3 Firefly-inspired models

As mentioned above this has been a tremendously productive source of inspiration, probably because of the simplicity of the behaviour and the clarity in which it can be described. Babalogu et al. [19] use a heartbeat-inspired model for control of nodes in an overlay network, but they go beyond the situation when nodes are always reliable and consider what happens when some nodes are unreliable. What other information can they draw upon to stimulate synchronization among the nodes? They present a model for flash synchronization inspired by flash signalling among fireflies. In a number of species of fireflies, males emit flashes at dusk, and are able to globally synchronise their signalling despite each individual only having local knowledge of the swarm. In their model the flash signaling overcomes the problems of only local knowledge of the pacemaker stimulus initially placed on the model. This can support synchronization in an overlay network despite only local knowledge at any particular node.



## 2.6 General issues surrounding synchronization in computer systems

Babalogu et al. refer to *clock synchrony* in relation to computer networks as being an important motivation for synchronization algorithms. But in dynamic overlay networks time synchronization may not be feasible, or at least difficult to measure. Wireless sensors networks require a degree of synchrony in the context of their deployment, but this is constrained by the demands of their physical environment in a manner unlike computer networks or dynamic overlay networks. Babalogu et al.'s emphasis on heartbeat synchronization moves away from the idea of a reference clock, and generates the synchronization reference from within the group itself.

Hu and Servetto [20] address the issue of errors in synchronization across wireless networks, which typically grow as hop number increases. They tackle this scalability problem through a model which incorporates the Mirollo and Strogatz [16] model as a special case, but is more generally applicable. They find that this results in the averaging out of errors across the network so that all nodes in a multi-hop network can maintain identical timing.

Iwanicki et al. [21] use gossip across a network to induce synchronization, since clock synchronization as discussed above may not be feasible in many highly dynamic distributed systems. In their study they use local decisions about sampling of synchronization and spreading of information by gossiping, but this in creating synchronization across a network, and their model has been supported by experiments scaling up to large numbers of nodes.

These represent a few of the different approaches to overcoming the problems of synchronization in computer networks, but provide a positive basis, since they do not make the demanding assumptions that clock synchronization requires, and yet appear to work more effectively.

## 2.7 Implications for CASCADAS

Not all models of oscillatory or synchronizing systems appear directly applicable to the distributed autonomous elements of CASCADAS. As discussed above, although oscillators may converge in some properties (such as phase-locking), this does not guarantee that they will synchronize. Also, while there are many empirical examples of synchronization from the natural world [5, 6, 7, 8, 9, 12, 13, 14] the factors causing them are often so complex as to make it difficult to apply the insights from these observations in the form of simple models to computer systems.

More positively, a number of natural systems have inspired highly useful models: reaction-diffusion [3, 4]; synchronized insect emergence [10]; biological pacemakers and firefly signaling [16, 19]; neural networks [6, 18]. Given that these systems are well-understood and can be explained in terms of relatively simple mathematical models, they may have considerable potential to inspire applications of synchronization in networked nodes.

Finally, a body of work has begun in distributed computing aiming to develop new ways of dealing with synchronization, because existing techniques often don't work as systems scale up [19, 20, 21]. These represent some very useful exploration of the problem, but additional approaches are needed because they may be dependent on the context in which they are deployed, and the CASCADAS project is generating some new challenges for distributed systems.



### 3 Synchronization through pattern-matching (“type 1”)

#### 3.1 Introduction:

We envisage the problem of forming virtual teams of computing devices (or software entities), based on their collaborative needs (i.e. number of partners required) and individual activity pattern (i.e. hours of the day when they are expected to be “active” or “online”). The rationale is identical to that of Grid computing in general, namely that in the case of a heterogeneous workload (i.e. local “bursts” in demand), forming virtual organisations can help balancing the load and making better use of idle resources. The main difference is that we seek to form these collaborative teams without centralised brokering, through a pattern-matching exercise resulting in the progressive rewiring of a P2P overlay.

#### 3.2 Minimum compliant overlay

As a first step, we have sought to identify the key properties of the minimum overlay allowing every node to reach its target number of backups at any time during a daily cycle. This typically involves having more than the target number of neighbours, in order to be able to compensate for periods in which one or more of them become unavailable.

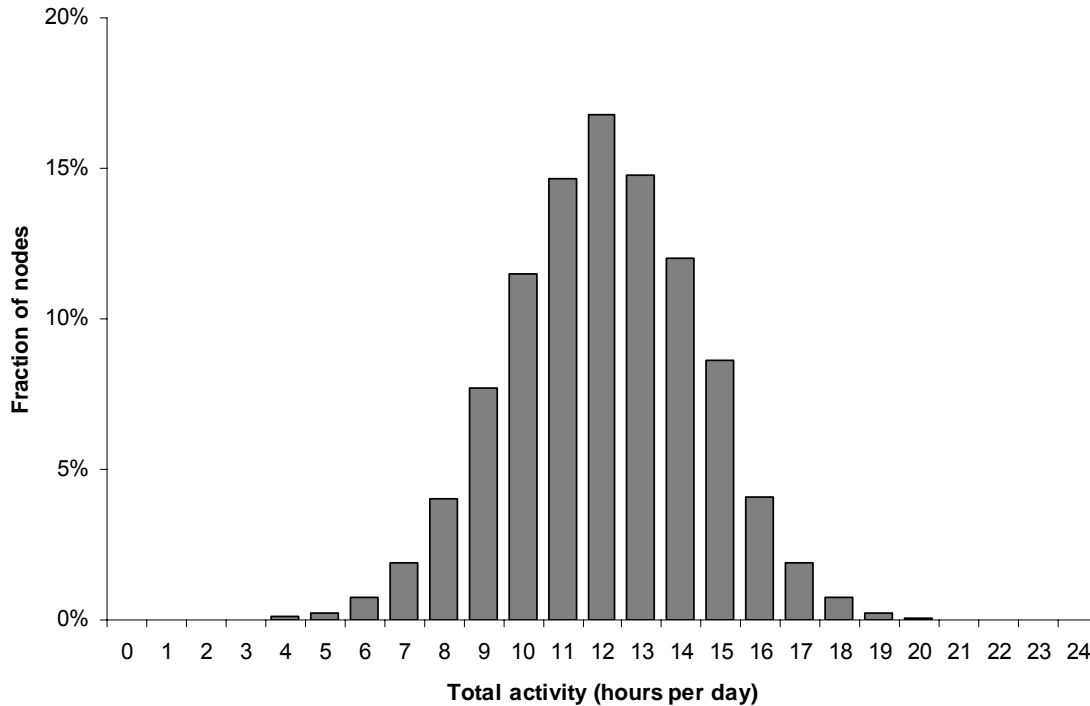
For instance, let us use a binary notation where “1” indicates that a node is active and “0” that it isn’t, and consider the trivial case in which every node in the system needs at least one backup at any one time. It is not enough for node *X*, whose activity pattern is “111000” (meaning that *X* is active during the first half of the day, discretised in 6 slots of 4 hours each), to have a link to a node *Y* whose signature is “101010”. *X* needs at least one additional neighbour whose daily pattern is “\*1\*\*\*\*” (“\*” designating either value). In this scenario, node *Y* is not satisfied with *X* as its sole neighbour either: it needs at least one additional backup whose daily pattern is “\*\*\*\*1\*”. Note that any node *Z*, whose activity pattern is “\*1\*010” (the two “0” corresponding to the two slots when neither *X* nor *Y* would be available as backup) can close the *X*-*Y*-*Z* triangle, forming an autonomous, self-sufficient virtual team according to these particularly simple requirements.

The problem becomes a lot more subtle when the target number of backups becomes higher, the granularity of the pattern becomes lower (longer bit-string), and/or the total size of the population increases. In order to obtain a first approximation, we generated a complete graph (every node connected to every other node) and pruned it until every remaining link is vital to at least one of the two nodes it connects (i.e. removing it would result in the available backup dropping below target during at least one slot of the full cycle).

Evidently, for any given level of granularity, the number and distribution of active slots is of paramount importance. The most realistic case would probably involve “blocks” or “aggregated” activity periods corresponding to, e.g., business hours (0900-1700). However, as a benchmark for comparison purposes (and because of its possible scientific relevance) we first explored the more generic case in which individual signatures are strictly random,



i.e. every bit in the string is independently set to “1” or “0” with a 50% chance, meaning that we can expect nodes to be normally distributed around the average of  $T/2$  active slots per day, with  $T$ =total number of slots (cf. fig. 3.1).

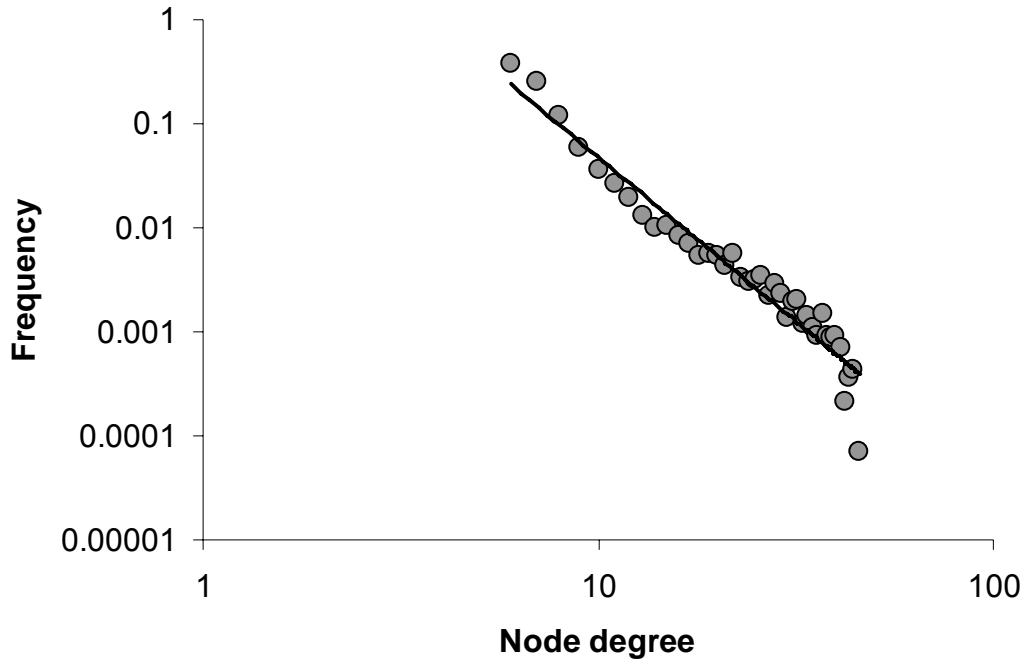


**Fig. 3.1: Frequency distribution of nodes as a function of their total activity over a day (number of active slots, or “1” in a string of 24 bits). This data is for 100 randomly generated populations of 100 nodes each.**

Because the active periods of every node are also randomly distributed over the  $T=24$  slots, the average size of the active population during any one hour period is  $N/T$ , with  $N$  indicating the total number of nodes. There will however be strong random fluctuations, unless  $N \gg T$ . The question we are trying to answer is therefore: in this completely randomised scenario, for a given population size and target backup (assumed identical for all nodes) can we identify some characteristic properties of the minimum compliant overlay?

Figure 3.2a shows the degree distribution of the stabilised collaborative overlay for the case in which  $N=120$  nodes are seeking to maintain just enough links to guarantee  $b=4$  active backups during every time-slot in which they are themselves active (the day is divided into  $T=24$  one-hour slots). The pruning is done by removing the least useful link of every node and repeating this operation until no more connections can be severed. The least useful link for a particular node is defined as the link to the neighbour with which it shares the least active slots, i.e. for which applying the AND operator to the two bit-strings returns the lowest number of “1” (when there is a tie, one link is cut at random). Note that the “least useful” relationship is not necessarily symmetrical (i.e.  $X$  can be the least useful neighbour of  $Y$  and  $Z$  the least useful neighbour of  $X$ ), which is part of the reason why some links cannot be removed even though they are useless to one of the two neighbours.





**Fig. 3.2a: Distribution of node degree in the stabilised overlay configuration. The first point regroups all nodes with degree equal or below 6 the most commonly observed degree for these parameter values ( $N=120$ ,  $T=24$ ,  $b=4$ )**

Figure 3.2b shows that degree distribution is only marginally affected by population size, although there is an understandable trend towards wider distribution for larger systems. Indeed a smaller sample (lower  $N$  value) implies that finding a good match can be more difficult for some nodes, which then need to maintain more connections to reach their target backup (hence the shift of the most commonly observed degree from 6 to 7 for populations of size inferior to 120). Conversely, for higher values of  $N$ , some nodes can be expected to retain a higher number of neighbours, due to the combined effect of higher initial connectivity and increased statistical variability.

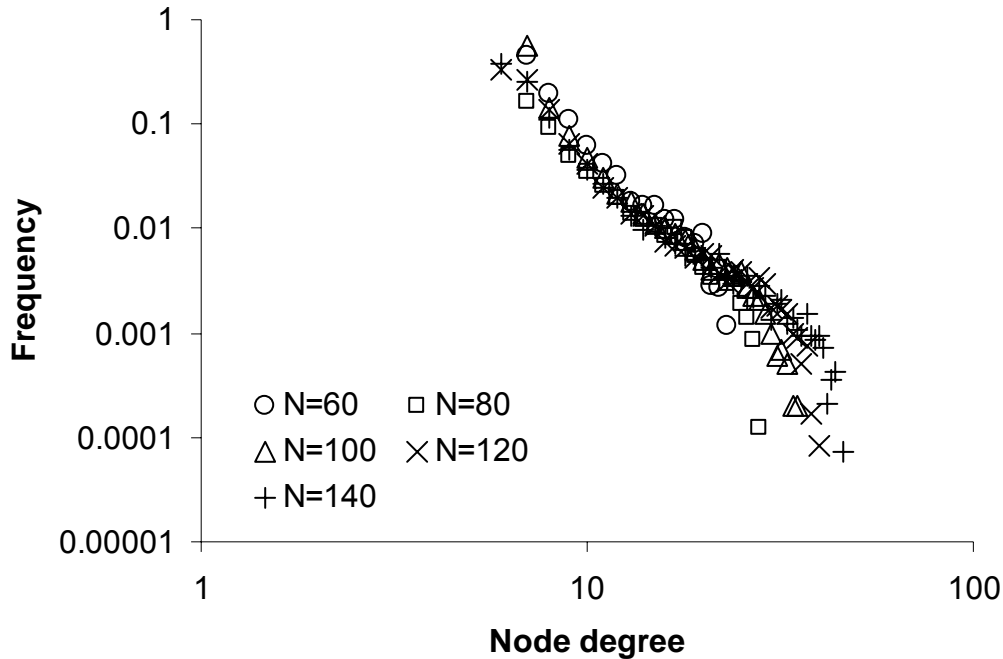


Fig. 3.2b: Comparison of node degree distribution for variable network size (other parameter values are unchanged, i.e.  $T=24$ ,  $b=4$ )

### 3.3 Realistic overlay creation/maintenance

#### 3.3.1 Random activity pattern

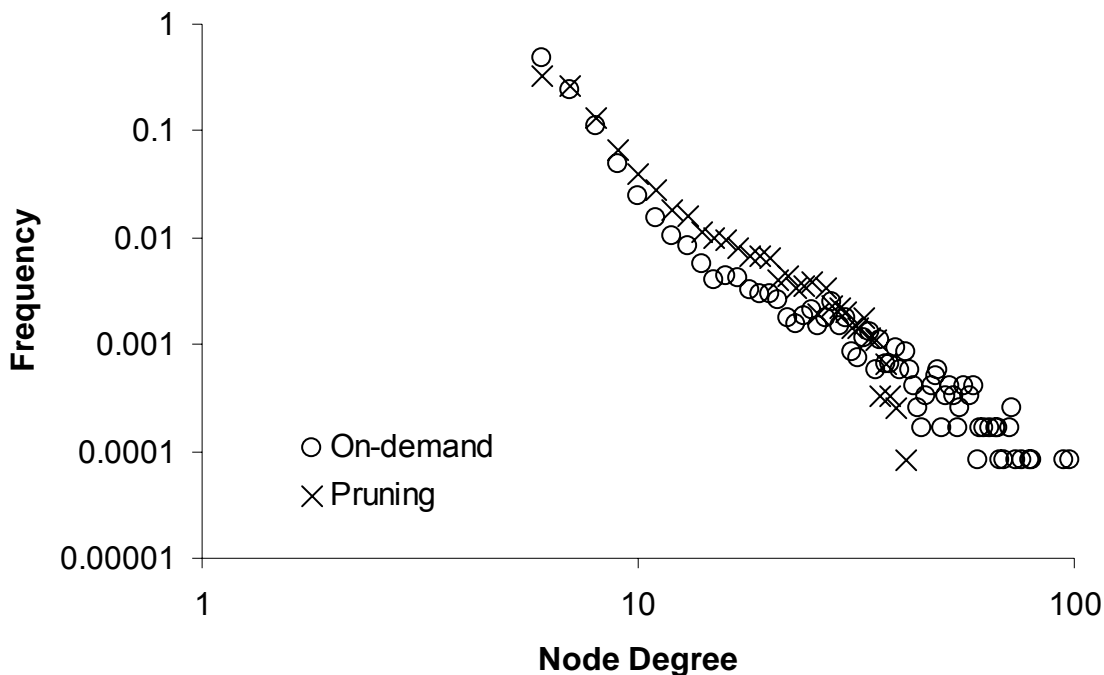
Clearly, identifying the minimum compliant overlay through pruning of a complete graph is only useful to study the key characteristics of its topology. It can certainly not be expected that the initial state of any network comprised of more than a handful of nodes will be anything like a complete graph of  $N(N-1)/2$  edges. It is therefore necessary to identify a set of local rules capable of supporting the “growth” of a satisfactory collaborative network starting from initial conditions less favourable to the discovery of the minimum compliant overlay. The measure of success of the underlying rewiring algorithm will be its ability to approximate the characteristics of this ideal target configuration.

We defined initial conditions as involving a random graph of arbitrarily chosen average degree (number of links). We then used a variant of the “on-demand” clustering algorithm to support a form of rewiring that only involves interaction between first neighbours (and no flooding, see D3.1 and D3.2 for details). Basically, instead of the “neighbour request” message specifying a desired type (or “colour”), the matchmaker uses the activity patterns of its neighbours to rank them by order of decreasing compatibility with the initiator’s. This list is then examined sequentially (and the initiator informed of the corresponding candidate identity) until a successful handshake occurs.



We relaxed the rule on link conservation, which is potentially incompatible with the requirements of synchronization (e.g. there may not be enough connections in the initial configuration to allow all participants to reach their target backup). It was replaced by a “local pruning” rule, designed to prevent link proliferation, the precise logic of which is as follows. Every time that a rewiring attempt has taken place (independently of its outcome), the initiator also examines its neighbourhood, in search for links that have become obsolete. A link is declared obsolete if nodes at both ends would still reach their target backup for all time-slots in which they are active if this link was severed. Whenever this is the case, the connection is effectively terminated. Note that the order in which links are checked impacts on which can be severed: at present, they are examined in order of increasing “usefulness” to the initiator (increasing overlap between activity patterns).

Figure 3.3a shows a comparison between the node degree distributions resulting, at steady state, from the benchmark “complete graph pruning” scenario and from the more realistic “random graph rewiring” scenario (average degree of 8 in initial conditions, i.e. 480 links for 120 nodes). In the latter, simulation was stopped after 100000 rewiring attempts, which was found to far exceed the typical number required for convergence in a system of this size.



**Fig. 3.3a: Node degree distribution for the realistic, “on-demand” clustering algorithm, compared to the benchmark “pruning” scenario ( $N=120$ ,  $T=24$ ,  $b=4$ )**

Although there is substantial overlap between the two distributions (which indicates that the local rules are capable of successfully approximating the minimum compliant overlay), the realistic scenario results in a small fraction of nodes acquiring a much higher number of neighbours. We interpret this effect as resulting from the difficulty for “hyperactive” nodes to discover partners with similarly high activity levels, now that they no longer start with full visibility over the whole system. This explanation is supported by the shape of the



correlation curves between node degree and average daily activity, which appear to diverge very clearly for activity values higher than ~16 hours/day (see Fig. 3.3b)

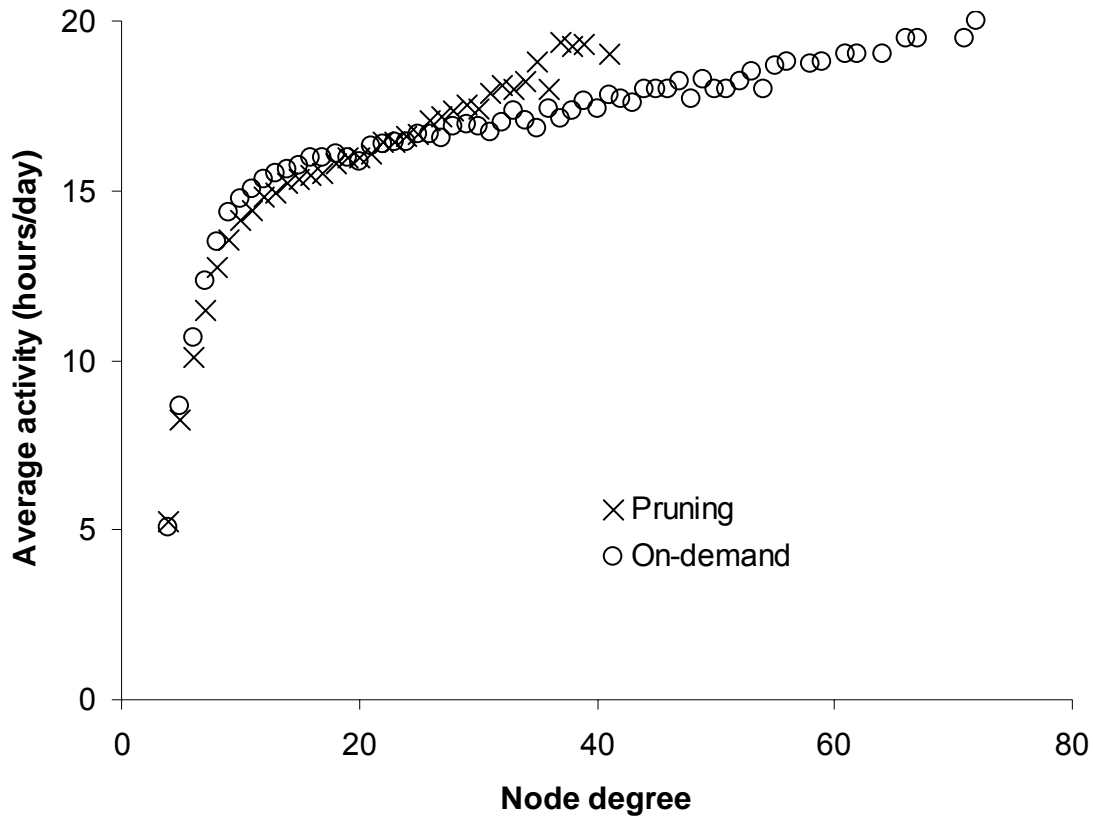


Fig. 3.3b: Correlation between average daily activity and node degree. Comparison between the realistic, “on-demand” clustering algorithm and the benchmark “pruning” scenario ( $N=120$ ,  $T=24$ ,  $b=4$ )

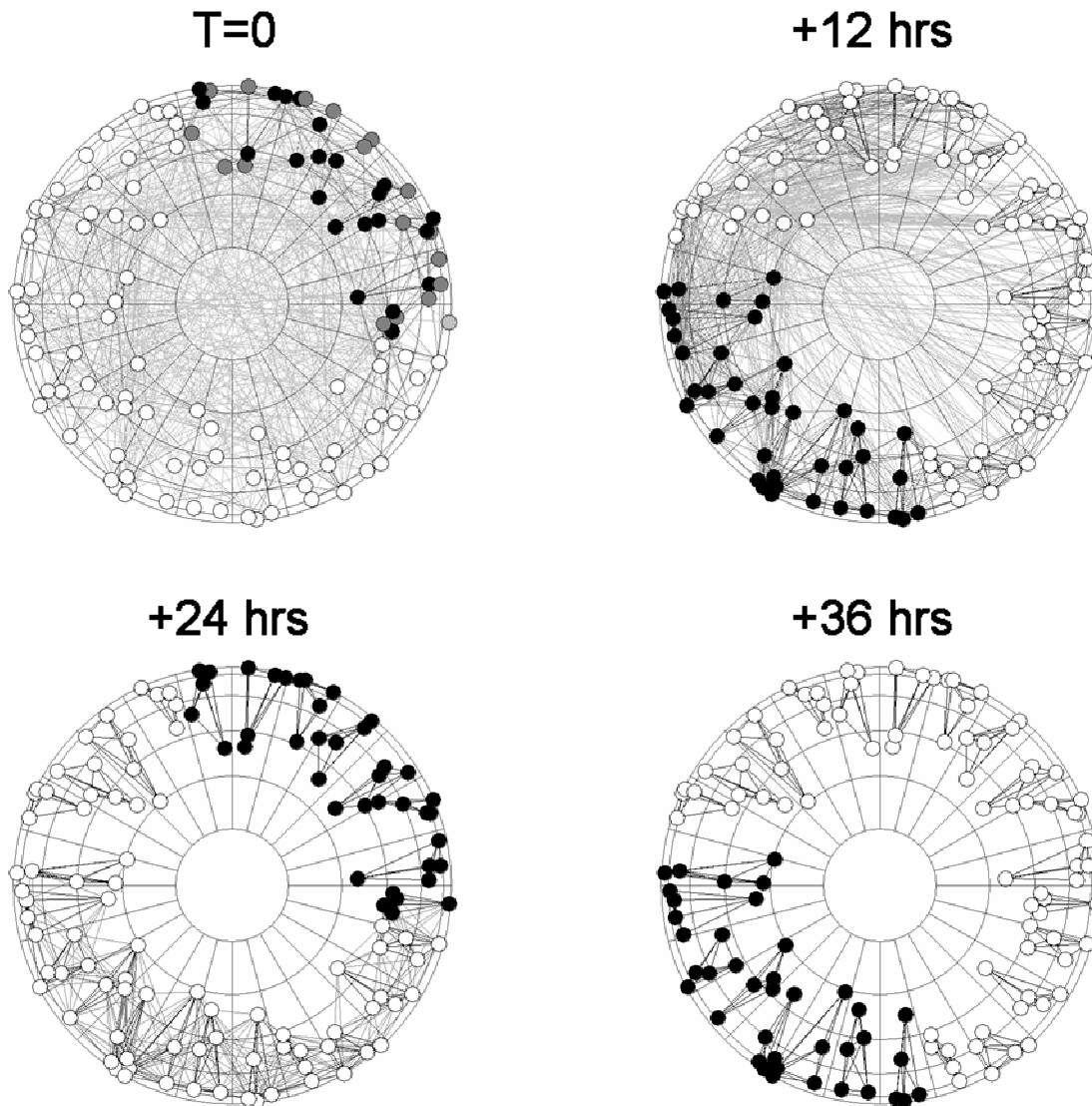
### 3.3.2 Time zone-defined activity pattern

For most applications, time of day will play a critical part in the activity pattern of individual nodes, hosts or components. For instance, one can easily imagine a device that is switched on (and can therefore join a collaborative team) only during business hours, say between 0900 and 1700. In this situation, starting from an initially random graph, the rewiring algorithm should be able to converge towards a situation in which first neighbours tend to belong to the same (or nearby) time zones, even though neither the time of the day nor the geographical location of the host is explicit in the “on-demand” scenario.

The simplest possible case involves a population of static locations, evenly distributed around the Earth, where hosts switch on and off at exactly the same time within each zone. In this case, if there are enough nodes in total for each one of them to find sufficient backup within its time zone, i.e. if  $N \geq T(b+1)$ , the random graph should “split” into region-specific collaborative teams, as nodes gradually establish preferential relationships with

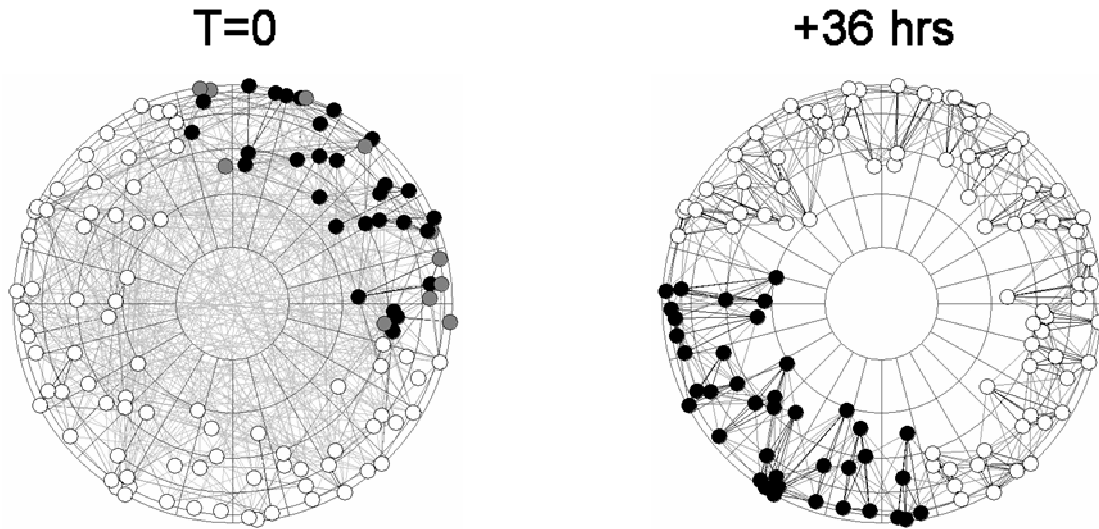


their geographical neighbours (with which compatibility of activity patterns is maximum). For  $N=120$ , there are exactly  $b+1=5=N/T$  nodes per time zone, and so each collaborative team should be an isolated complete graph. This is indeed what is observed, as shown in figure 3.4a.



**Fig. 3.4a: Time series showing the evolution of the collaborative overlay in the time zone-defined activity pattern scenario ( $N=120$ ,  $T=24$ ,  $b=4$ ). Greyscale (“darker=better”) indicates satisfaction (nodes) or compatibility (links). The representation is a top-down view from the pole (i.e. “wedge-shaped” regions are time zones, concentric circles are parallels).**

Conversely, if the total backup available within a specific time zone is not sufficient to meet individual requirements, then connections to additional hosts in other regions will spontaneously be maintained to compensate for the local shortage, as shown on figure 4b.



**Fig. 3.4b: Evolution of the collaborative overlay in the time zone-defined activity pattern scenario ( $N=120$ ,  $T=24$ ,  $b=6$ ).**

Clearly, in reality, fluctuations will affect the time at which individual units switch on and off, and/or the actual activity pattern during business hours. Such fluctuations can be categorised either as regular, i.e. nodes that are not entirely in synchrony with the majority of others located in the same time zone (e.g. whose “working hours” are 0800-1600) but repeat the same pattern every day, and irregular, i.e. the same node changes its activity pattern from one day to the next. In section 3.3.3, we investigate the influence of regular “discrepancies”.

### **3.3.3 Regular discrepancies in time zone-defined activity patterns**

We first assume that switch on/off times of nodes can be randomly offset by a given amplitude  $a$ , expressed in terms of the chosen discrete unit (e.g., in the scenario where the day is divided into  $T=24$  slots of one hour,  $a=2$  means that switch on time can vary between 07:00 and 11:00, local time). However, from that point, every node remains active for the same duration of 8 hours, aggregated into a single, uninterrupted block.

Results suggest that the actual value of  $a$  does not significantly impact on the self-organisation process: nodes that are “out-of-sync” with their geographical neighbours simply form partnerships with other time zones (the only difference is that, the higher  $a$ , the longer the “distance”). Although somewhat counter-intuitive, this result can actually be anticipated by considering that the initial overlay is a random graph and that the nodes have no explicit notion of geographical neighbourhood. Therefore, the problem they face is exactly the same whatever the value of  $a$ , even though the noise level is increased (which accounts for the higher average node degrees).

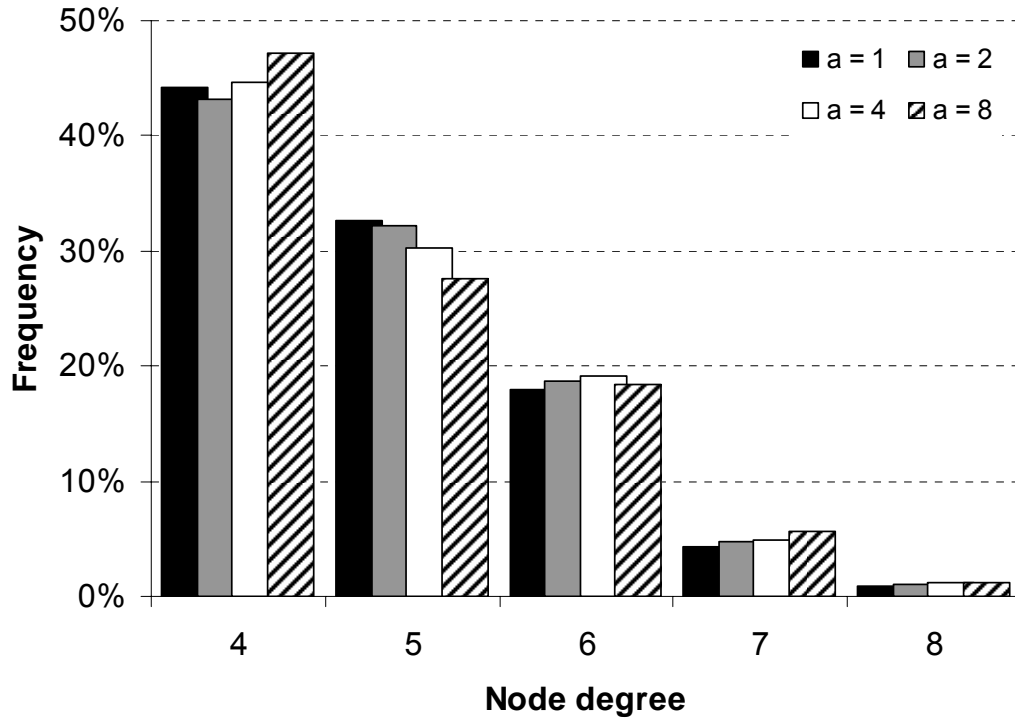


Fig. 3.5a: Distribution of node degree for variable maximum values of the offset parameter  $a$  ( $N=120$ ,  $T=24$ ,  $b=4$ , see text for details).

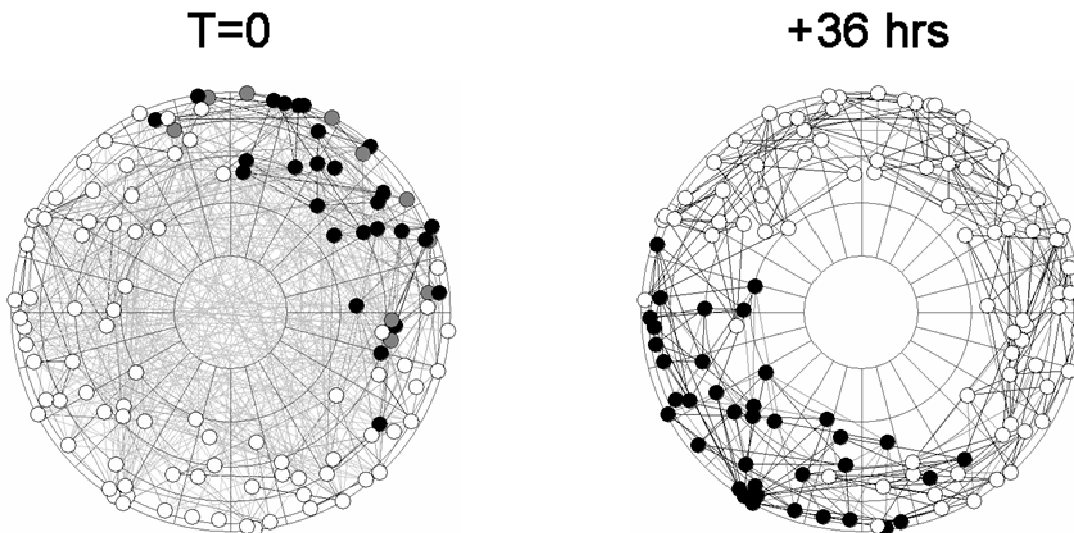


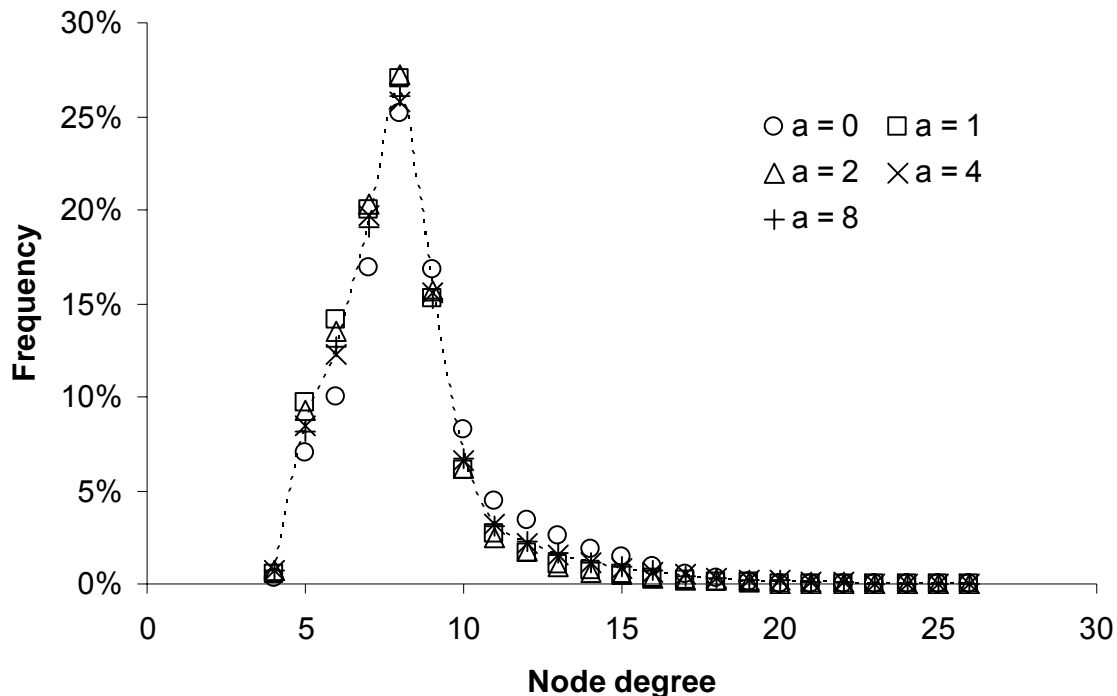
Fig. 3.5b: Evolution of the collaborative overlay in the time zone-defined activity pattern scenario with regular discrepancies ( $N=120$ ,  $T=24$ ,  $b=4$ ,  $a=2$ ). Note that the most noticeable difference with fig. 4 is the presence of “long-range” perfect matches (black links).

The next step is to introduce noise in the activity pattern itself, i.e. interruptions and/or completely “out-of-hours” active slots. In order to avoid any other “inequality” between



nodes, (which would add an additional source of variability similar to that investigated in section 3.3.1), we maintained the total number of hours that every node spends online constant. This effectively means that, for every one-hour interruption during the “normal” business hours, one bit which would normally correspond to an “out-of-hours” slot is set to one.

Because we did not have time to systematically explore the whole parameter space, we followed the following procedure: for each time-slot during which the node is supposed to be active (whatever the value of the offset parameter  $a$ ), we perform an additional random test by drawing a (uniformly distributed) real number comprised between zero and one. If it is lower than the noise level  $l$ , the slot is deemed to be an “interruption” of the normal activity cycle and a (randomly selected) “out-of-hours” slot is set to “active” instead.



**Fig. 3.6a: Node degree distribution for  $l=0.125$  and variable values of the offset parameter  $a$ . Other parameter values are unchanged ( $N=120$ ,  $T=24$ ,  $b=4$ ).**

Results (cf. fig. 3.6a and 3.6b) confirm intuition that the higher  $l$ , the higher the average number of neighbours required in order to secure adequate backup. They also indicate that, even when combined with extra noise affecting the “internal structure” of the activity pattern, the maximum offset period  $a$  does not appear to have a strong effect, apart perhaps on the shape of the distribution. Indeed, the slightly bimodal distribution for  $l = 0.25$ , which suggests that, even though all nodes have rigorously identical total activity over a 24 hours cycle, as well as statistically identical noise levels, they effectively split into two distinct subpopulations, appears to disappear for higher values of  $a$  (see fig. 3.6b). We currently have no satisfactory explanation for this phenomenon.



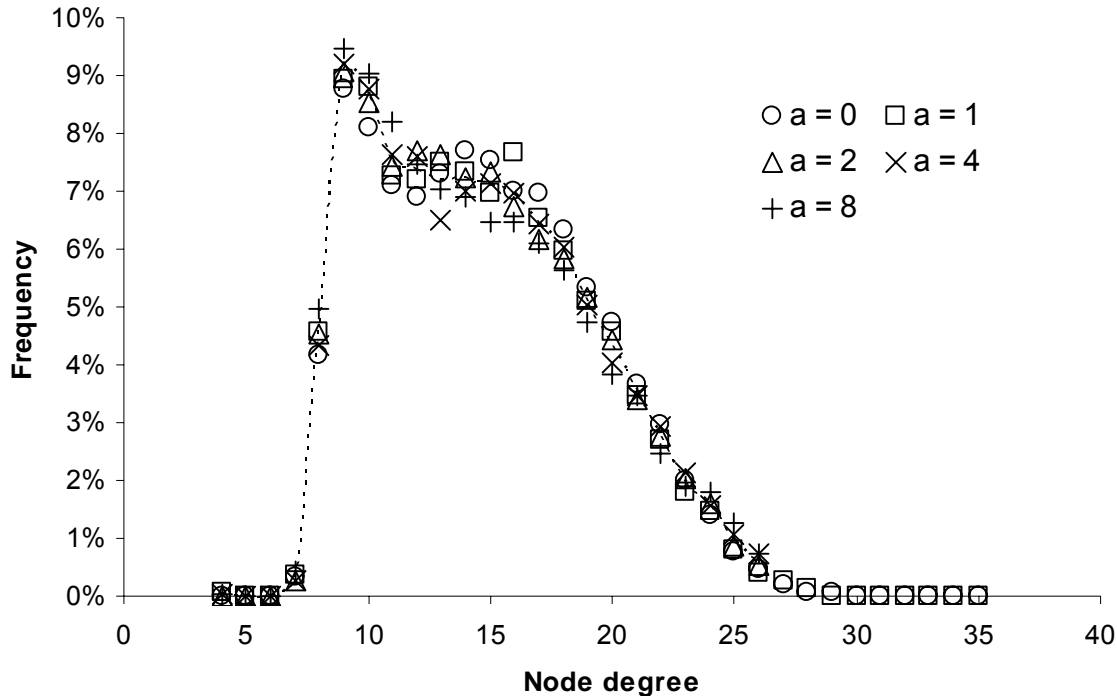
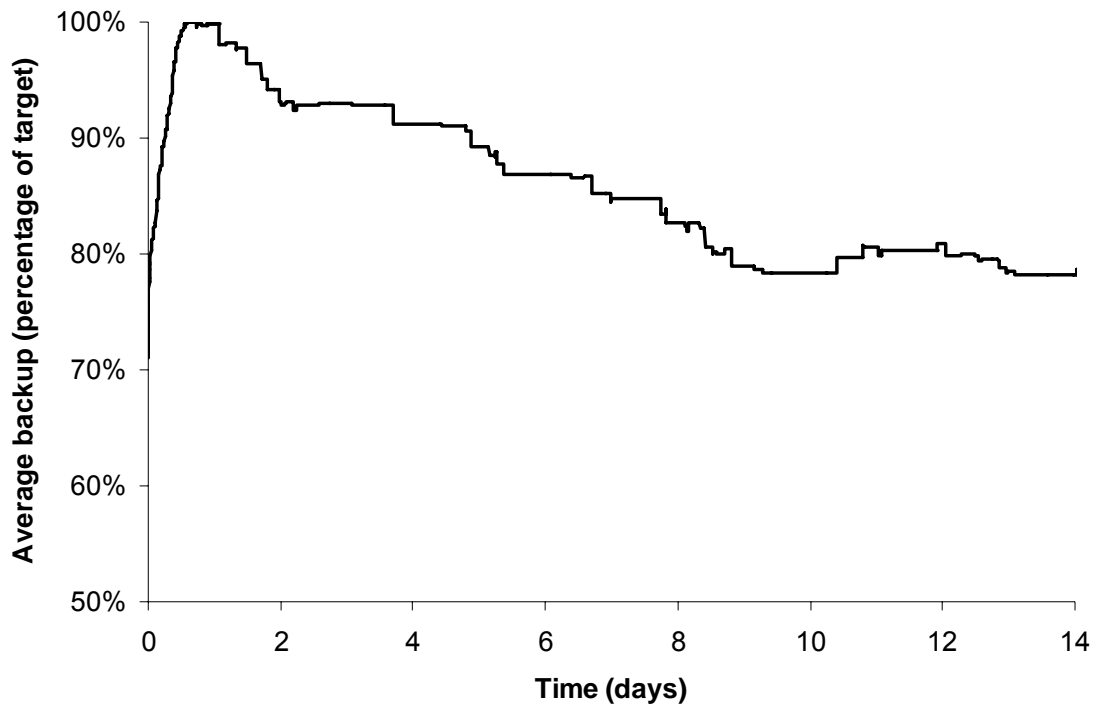


Fig. 3.6b: Node degree distribution for  $l=0.25$  and variable values of the offset parameter  $a$ . Other parameter values are unchanged ( $N=120$ ,  $T=24$ ,  $b=4$ ).

### 3.3.4 Movement between time zones

Our last set of experiments investigated the influence of nodes travelling between time zones, adopting local time (and changing their activity pattern accordingly) when reaching their destination. At every time-step, every node has a small probability of undertaking such a journey, which will take it to a randomly selected location on the planet (travel time is 12 hours, independently of the distance travelled). While moving, the node is inactive. Upon arrival, it reactivates its existing links (which will now be more or less out-of-sync, depending on how many time zones have been crossed), using them to “bootstrap” a new neighbour search that, ideally, should result in the “traveller” eventually establishing new collaborative relationships with “residents” of the time zone to which it has just relocated.

Early results are very thought-provoking in that they emphasise the positive role of factors that could intuitively be expected to impact negatively on the successful self-organisation of the collaborative overlay. Fig. 3.7a for example shows that the “noiseless” scenario in which activity periods are deterministic and there are enough nodes in every time zone to reach target backup without “long-range” links (illustrated in fig. 3.4a) is utterly crippled by movement.



**Fig. 3.7a: Evolution of average “satisfaction” over time when travel is permitted ( $N=120$ ,  $T=24$ ,  $b=4$ ,  $a=0$ ). Single simulation trace. See text for details.**

This phenomenon is caused by the fact that, once split into  $T$  complete sub-graphs, the system has effectively become very brittle: any node leaving its initial time zone is effectively incapable of reorganising its collaborative relationships since it can only “see” residents of its former time zone, which are themselves isolated from the rest of the population. The result is the slow but irreversible transition of the overlay from a quickly organised to an increasingly random state, as relocation results in cumulative noise that cannot be compensated using local rewiring.

In stark contrast, the scenario in which the population is insufficient to meet backup requirements by establishing collaborative relationships exclusively within each time-zone is extremely robust to node relocation (see fig.3.7b). This is because the tougher requirements effectively force the graph to remain globally connected, allowing for the rewiring of the overlay in response to perturbations. These results provide an excellent illustration of the importance of noise and what could be considered “sub-optimality” (e.g. imperfect match between collaborators’ activity patterns) to support adaptive features and robustness.

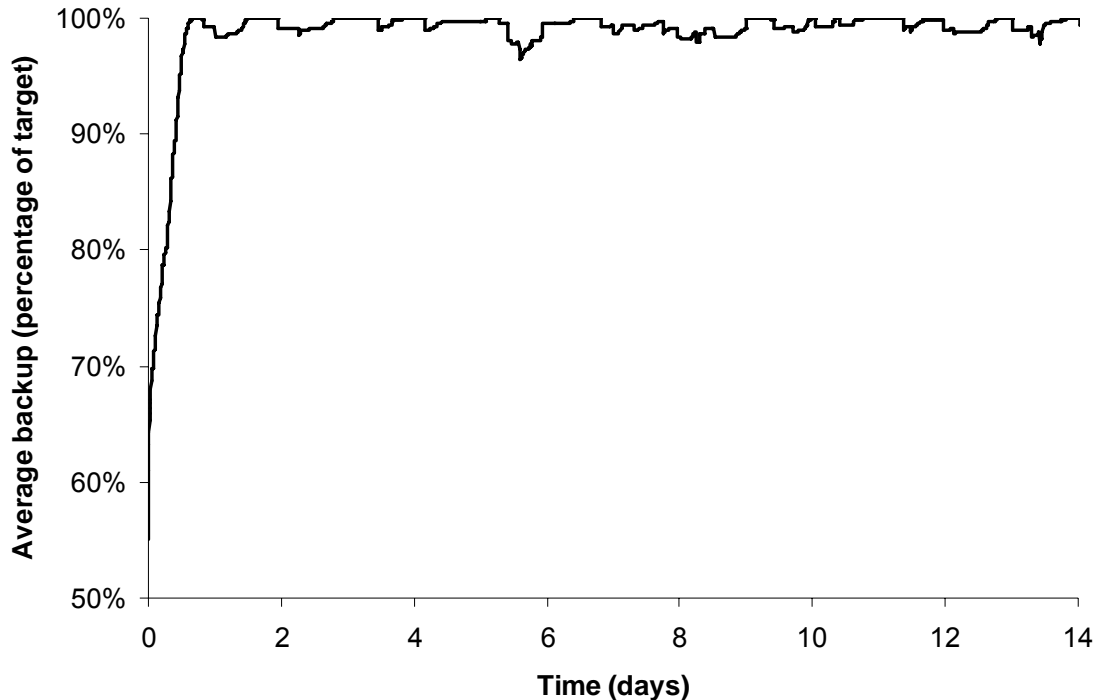


Fig. 3.7b: Evolution of average “satisfaction” over time when travel is permitted ( $N=120$ ,  $T=24$ ,  $b=6$ ,  $a=0$ ). Single simulation trace. See text for details.

## 4 Synchronization through individual adjustments (“type 2”)

In this section we present models for what we have called Synchronization of type II. The model describes populations of individuals that have individual clocks. If they are isolated, the individuals present the same rhythm and phase. This is a new class of synchronization model existing in the literature. We consider a class of model where individuals have a probability to change state that depends on the phase but independently of the time spend in this state.

### 4.1 Minimal model based on imitation

Individuals can exist in two states: Active or Inactive. They can switch from one state to the other. The model considers a double effect: (i) individuals in one state are stimulated by the number of individuals in the other state; (ii) individuals are inhibited to change state according to the number of individuals in their state.

Now we define the model and its parameters.

$N$ : total population;  $A$  ( $I$ ): total population of active (Inactive) individuals.



$$N = A + I$$

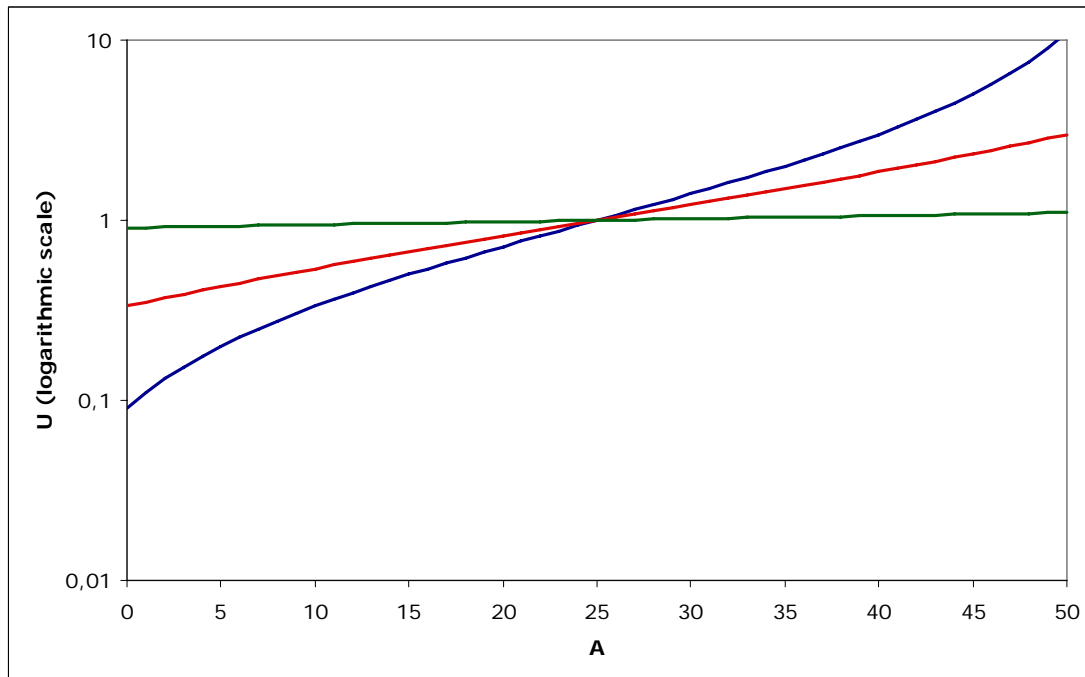
$$A \xrightarrow{\rho U} I$$

$$A' \xrightarrow{\rho U'} I'$$

$$U = \frac{\Delta + I}{\Delta + A}; U' = \frac{\Delta + A}{\Delta + I} = \frac{1}{U}$$

$U$ ,  $U'$  are functions that take into account the interactions between individuals. The larger  $\Delta$  the less function  $U$  is sensitive to the number of individuals  $A$  and  $I$ .

For  $\Delta$  tending towards infinity, the social interaction tends to zero.



**Figure 4.1.** Function  $U$  as a function of  $A$  for  $\Delta=5$  (blue line), 25 (red line) and 500 (Green) for  $N=50$ .

The time evolution of fraction of individuals in state  $A$  and  $I$  is given by the following ordinary differential equations:

$$\frac{dA}{dt} = -\rho U(I, A)A + \rho' U'(I, A)I \quad (4.1)$$

$$\frac{dI}{dt} = -\frac{dA}{dt} = \rho U(I, A)A - \rho' U'(I, A)I \quad (4.2)$$

Changing variables:



$$a = \frac{A}{N}; i = \frac{I}{N}; \delta = \frac{\Delta}{N}; a + i = 1 \quad (4.3)$$

$$\frac{da}{dt} = -\rho U(i, a)a + \rho' U'(i, a)i \quad (4.4)$$

Solving equations 4.3 and 4.4 give the following steady states:

$$-\rho U(i, a)a + \rho' U'(i, a)i = 0 \quad (4.5)$$

$$-\frac{\rho}{\rho'} U^2(i, a)a + i = 0$$

$$r = \frac{\rho}{\rho'} = \frac{i\delta^2 + 2\delta ia + ia^2}{a\delta^2 + 2\delta ia + i^2 a} = \frac{(1-a)(\delta+a)^2}{a(\delta+1-a)^2} \quad (4.6)$$

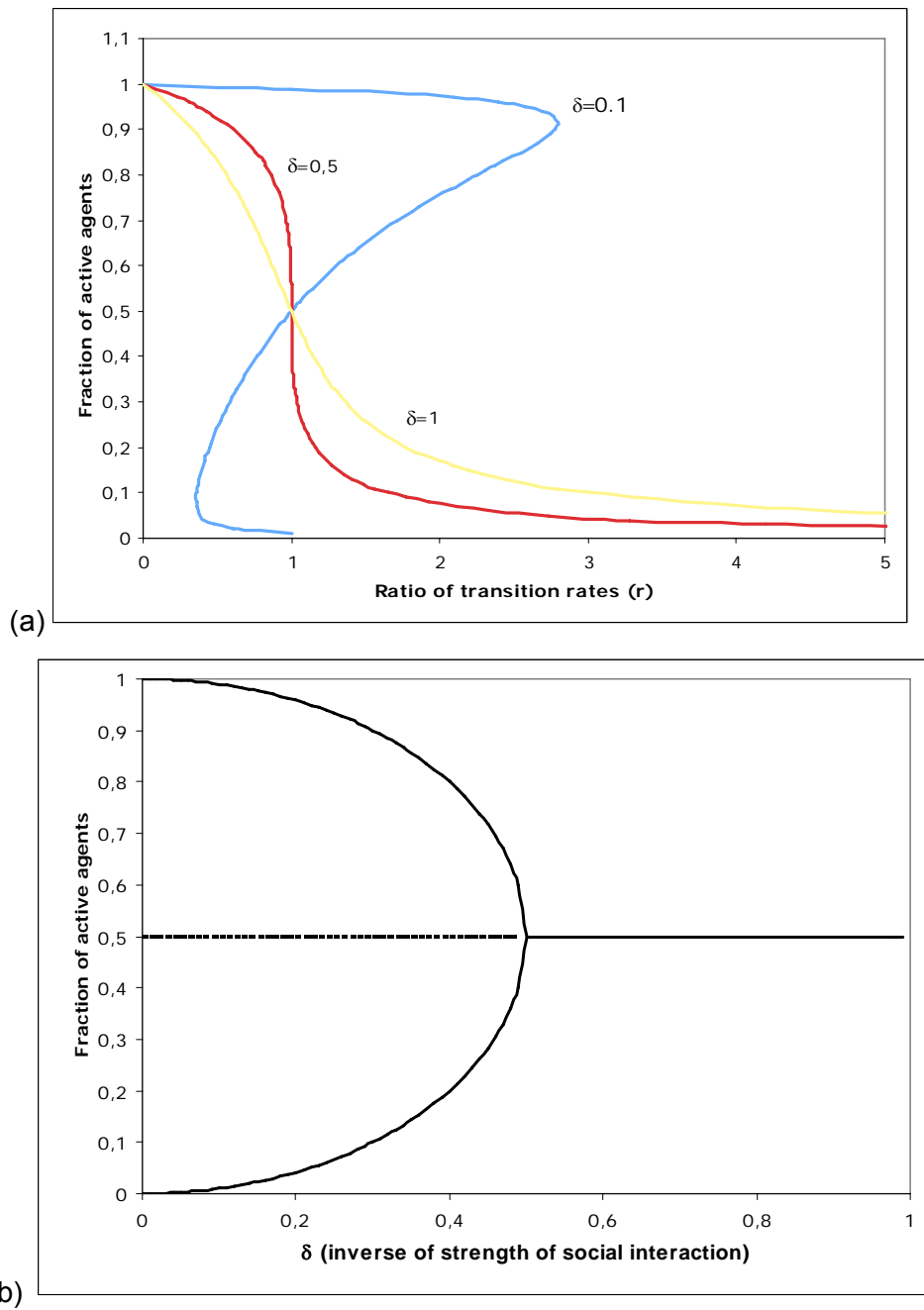
With  $r = \frac{\rho}{\rho'}$

The model presents one or three steady states as a function of  $r$  and  $\delta$ . The system only presents multiples states when  $\delta < 0.5$  (see Fig 4.2). In the case of existence of three steady states one is unstable and two are stable, this corresponds to a classical bistable system.

In the case  $r=1$ , corresponding to a symmetrical states, the systems present a classical pitchfork bifurcation (Fig 4.2). The steady states are given by:

$$a = 0.5 \quad (6, a)$$

$$a = 0.5(1 \pm \sqrt{1 - 4\delta^2}) \quad (4.7)$$



**Figure 4.2.** (a) Steady states of the systems as a function of  $r$  for  $\delta=0.1, 0.5$  et  $1$ . (b) Steady states as a function of  $\delta$  for  $\rho = 1$ , dotted line denotes unstable state

## 4.2 Rhythm within a single population

This version of the model is the same as in equation (1) but here the functions  $\rho$  and  $\rho'$  are periodic functions. We consider a period of  $2T$ , within a half-period ( $0$  to  $T$ )  $\rho' < \rho$ , which means that the inactive state is favored. It also means that in the same social environment,



active individuals tend to remain active. During the second half-period ( $T$  to  $2T$ )  $\rho' > \rho$ , in this case the active state is favored.

$$\frac{da}{dt} = -\rho(t)U(i,a)a + \rho'(t)U'(i,a)i \quad (4.8)$$

$$\rho = Kg(t) \quad (4.9)$$

$$\rho' = K(1 - g(t)) \quad (4.10)$$

Where  $K$  is a constant and  $g(t)$  is a function of period  $2T$ . We consider here the limit case of a step function with  $g = 1$  between  $0 < t < T$  and  $g = 0$   $T < t < 2T$ .

The model is characterised by three parameters  $K$ ,  $\delta$  et  $T$ . In the following we keep  $T$  constant noticing that changing the value of the period ( $2T$ ) is equivalent as changing the value of  $K$ .

We define three functions to describe the dynamical properties of the system. The first one is the average fraction of individuals in the active state over one period:

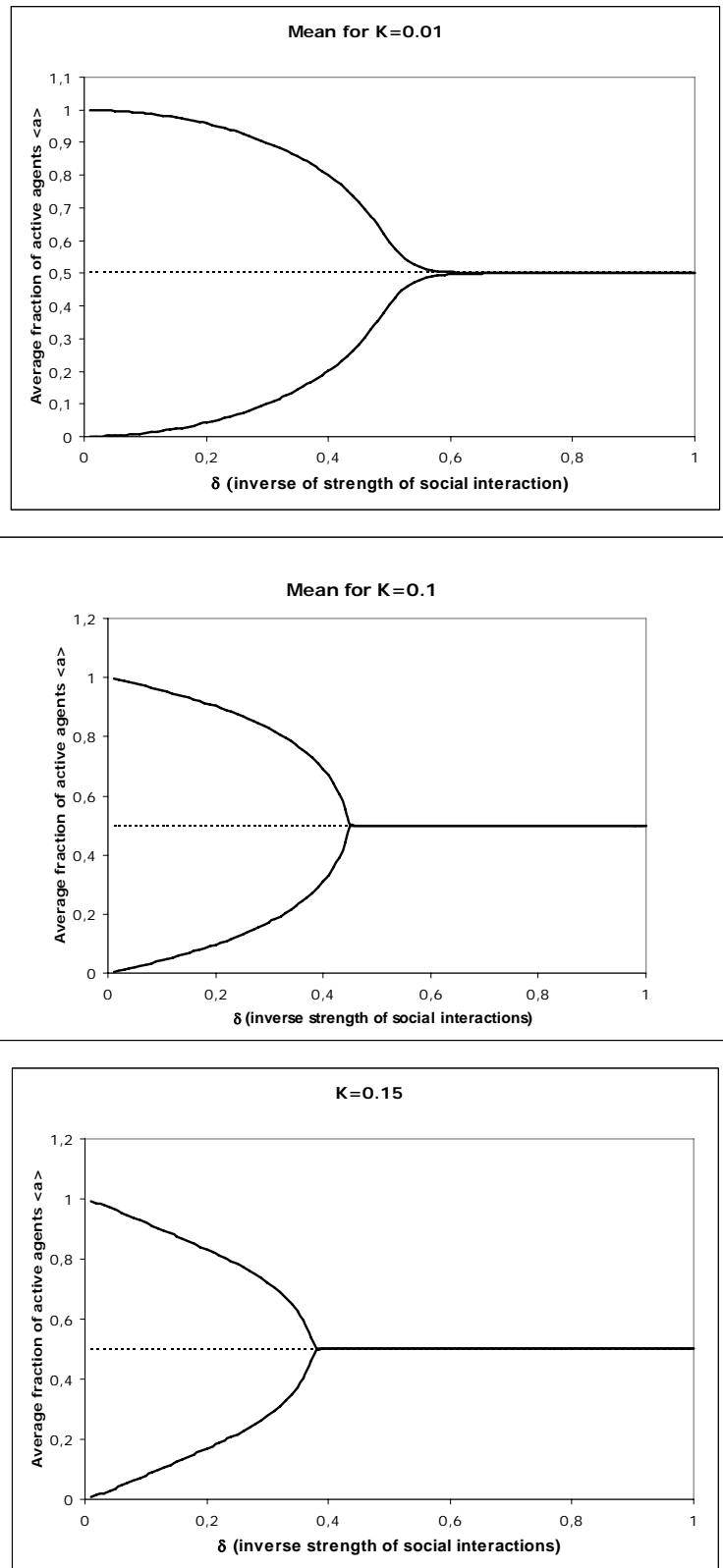
$$\langle a \rangle = \frac{0}{2T} \int_0^{2T} a(\tau) d\tau \quad (4.11)$$

The second one is the distance to the mean:

$$EM = \frac{0}{2T} \int_0^{2T} |\bar{a} - a(\tau)| d\tau \quad (4.12)$$

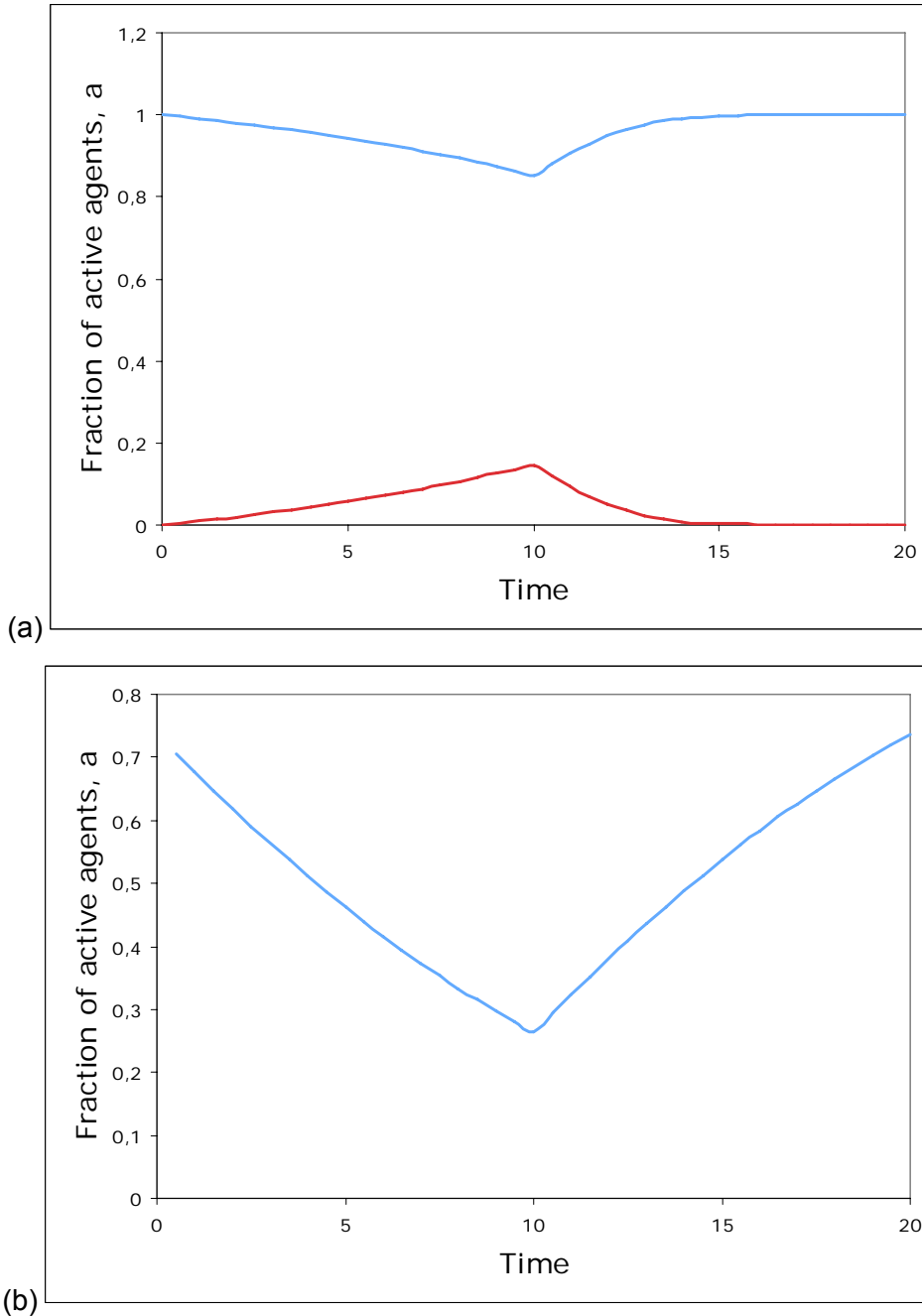
And finally the third one that are the maximal values of  $I$  (fraction of inactive agents) and  $A$  (fraction of active agents) that are reached at time  $T$  and  $2T$ .

Figure 4.3 shows the mean value of the fraction of inactive agents ( $\langle a \rangle$ ) as a function of  $\delta$  for different values of  $K$ . Again we find a pitchfork bifurcation leading to multiple coexisting states. For small values of  $\delta$ , two coexisting regimes can be observed: (i) small amplitude oscillations around a high value for  $\langle a \rangle$  and (ii) small amplitude oscillations around a low value of  $\langle a \rangle$  (correlatively corresponding to a high value of  $\langle i \rangle$  i.e. average fraction of inactive agents). The model also present unstable oscillations around  $\langle a \rangle = 0.5$ . Beyond a critical value ( $\delta_c$ ) this state becomes stable and oscillation around  $\langle a \rangle = 0.5$  are observed.



**Figure 4.3.** Steady state value of mean fraction of active agents ( $\langle a \rangle$ ) as a function of  $\delta$  for  $K= 0.01, 0.1$  and  $0.15$ . The half-period is set to  $T= 10$  time unit.





**Figure 4.3.** Oscillations of small or large amplitude with a half-period  $T$  of 10 time units. (a) For  $\delta=0.1$  the system presents small amplitude oscillations around a high level of activity. (b) For  $\delta = 2$  the systems presents large amplitude oscillations around a fraction of 0.5 of active agents.



### 4.3 The case of two populations out of phase

In the case of two populations of agents with a different phase of oscillations equations 4.8 becomes:

$$\frac{da_i}{dt} = -\rho_i(t)U(i_t, a_t)a_i + \rho_i'(t)U^{-1}(i_t, a_t)i_1 \quad i=1,2 \quad (4.13)$$

$$i_t = i_1 + i_2$$

$$a_t = a_1 + a_2$$

$$\rho_2(t) = \rho_1(t - \theta)$$

$$a_1 + i_1 = 0.5$$

$\rho_i(t)$  is the periodic function presents in equations 4.9 and  $\theta$  is the difference in phase between the two populations. To characterize the systems two indicators are defined: (i) the average total population (equation 4.14) and (ii) the absolute value of the difference between the two populations (equation 4.15).

$$\langle a_t \rangle = \frac{\int_0^{2T} (a_1(t) + a_2(t))dt}{2T} \quad (4.14)$$

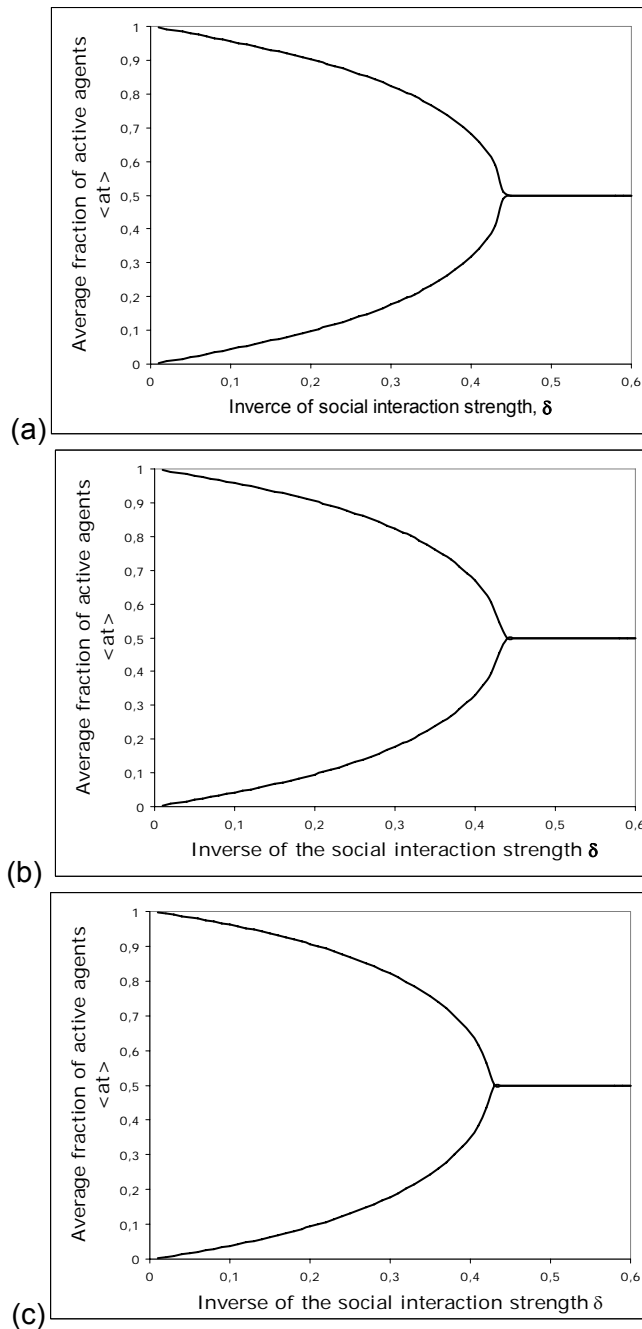
$$\langle |a_1 - a_2| \rangle = \frac{\int_0^{2T} abs(a_1(t) - a_2(t))dt}{2T} \quad (4.15)$$

In this case, again, the system presents a pitchfork bifurcations leading to the coexistence of two stable states.

The first type of states corresponds to two populations presenting both of them a high (or low) value of the average fraction of active agents. This phenomenon is observed despite the initial phase difference present between the two populations. In this case the absolute difference in the mean is small.

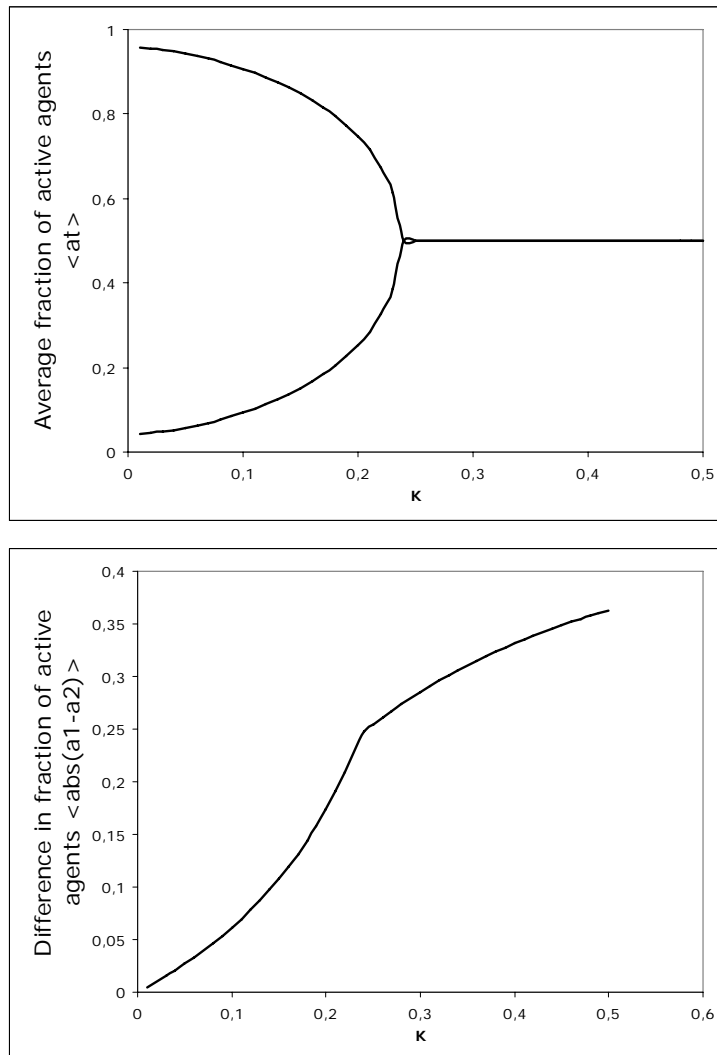
The second type of state corresponds to a phase difference between the two populations. The mean value of the fraction of active agents over a period is 0.5 and each population contributes to this mean by 0.25. In this case the absolute difference in mean is important.

The influence of parameter  $\delta$  is similar to the previous case of a single population. For  $\delta < 0.5$ , the asymmetrical solutions are stable and for  $\delta > 0.5$ , only the symmetrical solution is stable (Fig.4.4).



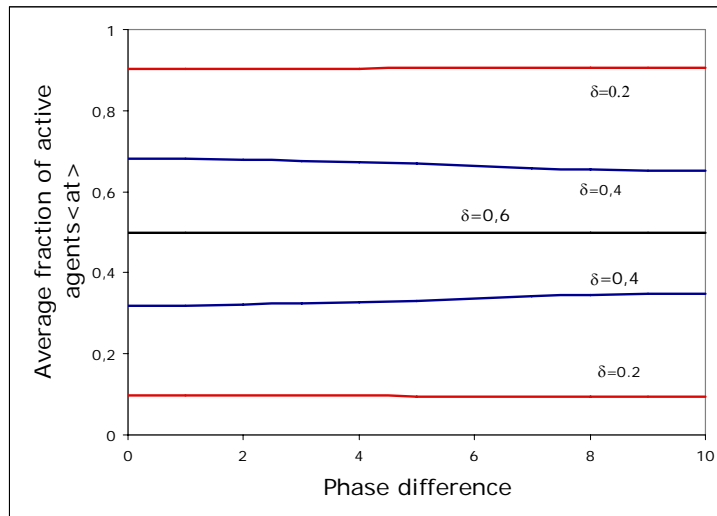
**Figure 4.4.** Bifurcation diagram as a function of parameter  $\delta$  for three values of phase differences (a) 0; (b) 5 and (c) 10. The value of  $K$  is 0.1 and the half-period is 10 units of time. For low values of  $\delta$  two states exist, for higher value only one stable state exists.

The effect of parameter  $K$  is the same as in the previous case (Fig. 4.5).



**Figure 4.5.** (a) Bifurcation diagram as a function of  $K$ . For large values of  $K$  only one stable state exists. (b) Difference in the fraction of active agents as a function of  $K$ .

This model shows that the global properties of the case of two populations are very similar to the case of only one population. More importantly, the model shows that the phase difference plays a small role in the kind of stable states that can exist in the system (Fig.4.6). This is unexpected as one would consider that the more out of phase the agents are, the more difficult it is to synchronize the system. It shows that social interactions can play a very important role when considering synchronization of agents.



**Figure 4.6.** Average fraction of active agents as a function of the phase difference for different values of  $\delta$ . The figure shows that regarding activity synchronization the important factor is the strength of social interaction more than the phase difference between the agents.

#### 4.4 Summary

In this section we have analyzed a model for synchronization of agent populations that takes into account social interactions. We consider agents existing in two states, active or inactive, which can switch from one state to the other. This switch can be a periodic function with periods of activity and rest. Each agent has its own preference but their behaviour is influenced by the number of agents in other states. Our results concern the influence of non-linear mimetic effect between agents. The fact that agents take into account what others are doing in a non-linear way is poorly discussed in the scientific literature. We have seen that simple models describing limit cases already present the possibility of coexisting states.

The first case presented of a single population shows already that the system can be blocked around states that correspond to steady states of the system without oscillations. Even in these very simple cases, we have also show that such system can present bifurcations leading to multiple coexisting states as a function of the social strength of interaction and of cooperative effects.

The model demonstrates that oscillations between active and inactive states can utterly disappear provided the agents are strongly influenced by each other's state. This alone constitute evidence that dynamical interactions between agents can produce complex behaviour that could lead to pathological states in terms of resource management.



## 5 General discussion and conclusions

It is relatively easy to find ad-hoc solutions to synchronization problems when neglecting their influence on the general characteristics of the overlay (for type 1 synchronization) or the fundamental properties (e.g. stability) of the resulting dynamical system (for type 2 synchronization). However, in-line with the objectives of WP3, which revolve around producing well-documented design patterns, we followed a principled scientific approach and attempted to explore the role of parameters as systematically as possible (within the limits imposed by time constraints).

In the case of synchronization through rewiring, which amounts to a form of pattern-matching, we found that random time signatures (in the form of randomly generated bit-strings of arbitrary length) result in the formation of a scale-free overlay when pruning from a complete graph and trying to secure a target number of active neighbours at any one time. We also identified distributed heuristics, based on the “on-demand” clustering algorithm, capable of approximating the near-optimal configuration obtained by pruning (see deliverable D3.4 for details).

When using more realistic time zone-defined activity patterns, we found that node degree becomes more homogeneous and that geographically co-located components successfully self-aggregate when starting from random conditions (as expected). We also discovered that noise and “sub-optimality” can play a crucial role in maintaining plasticity in a dynamic environment (e.g. when components occasionally migrate between time zones, requiring a local reorganisation of the overlay).

In the case of synchronization through individual adjustments, we chose to perform a thorough analytical study of a simple model rather than multiply examples. Once again, the quantitative properties of what appeared like the most basic logic turned out to be quite complex, some parameter values resulting in pathological stable states (e.g. locking the entire population in an “always on” or “always off” state). The finer study also revealed that, somewhat contrary to intuition, closer “preferred” activity periods did not necessarily translate into better synchronization between sub-populations.

The overall conclusion is that, despite the impressive body of knowledge available from the literature, the fundamental properties of “type 2” synchronization are far from sufficiently documented. Further research is required before the corresponding models and algorithms can be confidently used to enhance collaboration in a decentralised environment, under penalty of risking macroscopic pathological behaviour.

## 6 References

- [1] Rosenblum MG, Pikovsky AS & Kurths J 1996 Phase synchronization of chaotic oscillators. *Phys Rev Lett* 76(11), 1804-1807.
- [2] <http://en.wikipedia.org/wiki/Amplitude>
- [3] Murray JM 1993 *Mathematical Biology*, 2<sup>nd</sup> edn. Springer.
- [4] Ermentrout GB & Troy WC 1986 Phase-locking in a reaction-diffusion system with a linear frequency gradient. *SIAM J Appl Math* 46(3), 359-363.



- [5] Winfree AT 1967 Biological rhythms and the behaviour of populations of coupled oscillators. *J Theor Biol* 16, 15-42.
- [6] Koppell M, Ermentrout GB, Whittington MA & Traub RD 2000 Gamma rhythms and beta rhythms have different synchronization properties. *Proc Natl Acad Sci USA* 97(4), 1867-1872.
- [7] Nagano S 2000 Modeling the model organism *Dictyostelium discoideum*. *Develop Growth Differ* 42, 541-550.
- [8] Krause J & Ruxton GD 2002 *Living in Groups* Oxford University Press.
- [9] Couzin I & Krause J 2003 Self-organization and collective behaviour in vertebrates. *Adv Study of Behav* 32, 1-76.
- [10] Hoppensteadt FC & Keller JB 1967 Synchronization of periodic cicada emergence. *Science* 194, 335-337.
- [11] Fanshawe JH & FitzGibbon CD 1993 Factors affecting the hunting success of an African wild dog pack. *Anim Behav* 45, 479-490.
- [12] Levy O Appelbaum L Leggat W Gothlif Y et al. 2007 Light-Responsive Cryptochroes from a Simple Multicellular Animal, the Coral *Acropora millepora*. *Science* 318, 467-470.
- [13] Maynard Smith J & Harper D 2003 *Animal Signals* Oxford University Press.
- [14] Sutherland WJ 1996 *From Individual Behaviour to Population Ecology*. Oxford University Press.
- [15] Kokko, H., Mackenzie, A., Reynolds, J.D., Lindström, J. & Sutherland, W.J. 1999 Measures of inequality are not equal. *Am Nat* 154, 358-382.
- [16] Mirrollo RE & Strogatz SH 1990 Synchronization of pulse-coupled biological oscillators. *SIAM J Appl Math* 50(6), 1645-1662.
- [17] Buck J & Buck E 1976 Synchronous fireflies. *Sci Am* 234, 74-85.
- [18] Goel & Ermentrout GB 2002 Synchrony, stability and firing patterns in pulse-coupled oscillators. *Physica D* 163, 191-216.
- [19] Babalogu O, Binci T, Jelasity M & Montresor A 2007 Firefly-inspired heartbeat synchronization in overlay networks. *1<sup>st</sup> Intl Conf on Self-Adaptive and Self-Organizing Systems*, Cambridge, MA, July 2007, pp. 77-86.
- [20] Hu A-S & Servetto SD 2006 On the scalability of cooperative time synchronization in pulse-connected networks. *IEEE Trans Information Theory* 52, 2725-2748.
- [21] Iwanicki K, van Steen M, Voulgaris S 2006 Gossip-based clock synchronization for large decentralized systems. In: Keller A & Martin-Flatin, JP (eds.), *Self-Managed Networks, Systems and Services*, LNCS 3996, Springer.