KUNGL TEKNISKA HÖGSKOLAN
Royal Institute of Technology

Synchronisation Strategies in STDMA Tactical Radio Area Networks

XAVIER GELABERT DORAN

Master Thesis

October 2003

TRITA—S3—RST—XXXX
ISSN XXXX—XXXX
ISRN KTH/RST/R–XX/XX–SE

RADIO COMMUNICATION SYSTEMS LABORATORY
DEPARTMENT OF SIGNALS, SENSORS AND SYSTEMS

to my family and friends

Abstract

Future military tactical communication networks must be highly mobile, survivable and reconfigurable. Distributed multi-hop packet radio architectures have been adopted, combining robustness and autonomous operation. In order to meet the requirements for Quality of Service (QoS) and delay guarantees, Spatial Time Division Multiple Access (STDMA) schemes have been proposed. Key problems in STDMA are: to find efficient distributed algorithms for STDMA scheduling, slot synchronisation and to handle mobility. In this thesis focus is on slot synchronisation.

Time synchronisation is a critical piece of infrastructure for any distributed system. Moreover, slotted TDMA schemes are of special interest because of the natural mechanism it provides for refereeing the access to the medium: time. To consider timing inaccuracies and propagation delay effects a guard band is usually introduced and, therefore, perfect synchronisation is not required. Hence, good time synchronisation is important not only because it enables Time Division Multiple Access to the data link, but because it shortens the guard time allowing bigger packets to be sent.

In this thesis the performance in terms of synchronisation convergence and timing accuracy will be evaluated for a STDMA scheme in Tactical Radio Area Networks (TRAN). A network set-up environment will be considered and a description of how the synchronisation algorithm fits in the initialization process will be made. We also investigate some parameters related to the synchronisation algorithm and the effects when different topology configurations are used.

Acknowledgments

This thesis exists thanks to the support of many people to whom I am most grateful. I first wish to thank all the people at the Radio Communication Systems Laboratory at the Royal Institute of Technology for their help and support. I am most thankful to my examiner Professor Jens Zander for giving me the opportunity to carry out my thesis at RST. Especially acknowledged is my advisor Assistant Professor Tim Giles for valuable reviews and comments. I would also like to express my sincere gratitude to André Ebner from the Technische Universität Hamburg-Harburg for his help and encouragement.

My study period at the Royal Institute of Technology was made possible thanks to the help of different institutions and grants. I am particularly thankful to Elisa Sayrol and Patricia Duarte from the Universitat Politècnica de Catalunya (UPC) and the rest of people that made my stay in Sweden possible. The financial support received from the Erasmus student exchange program and from The Swedish-Spanish Foundation for the Promotion of Education and Studies are gratefully acknowledged.

Many others have helped me in different ways and it would be impossible to cite all of them here. Many thanks to the people I met during my stay in Sweden and for those I met while studying in Barcelona. Also my friends from Menorca because, although I am far away, they are always close to me.

Finally, I am indebted to my parents, Lea and Miquel, without their help and support any of this would have been possible. Their patience and encouragement during all these years has been outstanding. I can only say *mai vos ho podré pagar, gràcies.*

Contents

List of Figures

List of Tables

List of Abbreviations

CCN(s)	Communication Control Node(s)
CN	Core Network
CSMA	Carrier Sense Multiple Access
FIFO	First In First Out
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GW	Gateway
IEEE	Institute of Electrical and Electronic Engineering
km.	Kilometer
MAC	Medium Access Control
MHA	Minimum Hop Algorithm
MHPRN	Multi-Hop Packet Radio Network
MAI	Mutual Access Interference
PPS	Pulse Per Second
PRN	Packet Radio Network
QoS	Quality of Service
S-ALOHA	Slotted ALOHA
SINR	Signal-to-Interference plus Noise Ratio
STDMA	Spatial Time Division Multiple Access
SN(s)	Standard Node(s)
TCS	Traffic Controlled Schedule
TDMA	Time Division Multiple Access
TDD	Time Division Duplex
TRAN	Tactical Radio Area Network
UMTS	Universal Mobile Telecommunications System
UTRA	UMTS Terrestrial Radio Access
WLAN	Wireless Local Area Network

List of Notations

A	Signal's amplitude
$A_i(.)$	Antenna pattern used by node i
B	Effective noise bandwidth
$C(t)$	Time function
c	Speed of light ($c = 3 \cdot 10^8 \mathrm{m/s}$)
D	Linear fractional drift rate (ageing rate)
d_{ij}	Distance between node i and node j
$E[\cdot]$	Statistical expectation
$E[hops]$	Mean number of hops
$E[neighbours]$	Mean number of neighbours
F_{sys}	Receiver's noise figure
$\mathbf{G}$	Path gain matrix
G_{ij}	Path gain on link (i, j)
h_{ij}	Number of hops from node i to node j
i	Node identification number
j	Node identification number
K	Sample length of correlation sequence
k	Boltzmann's constant ($k = 1.38 \cdot 10^{-23}$)
L_{ij}	Radio propagation loss on link (i, j)
$\mathcal{L}_n$	Set of links assigned to transmit during slot n
L_0	Threshold propagation path loss for connectivity
$m[\cdot]$	Correlation sequence
N	Number of nodes that form the network
N_f	Number of slots in a schedule frame
N_i	Node i's identifier
$\mathcal{N}_i$	Number of neighbours to node i
n	Slot number
n_0	Fixed position of the correlation sequence within the time slot
P_i	Transmission power level used by node i
P_{Noise}	Background noise power level at the receiver
$P_{received}$	Received power at node j
$P_{transmitted}$	Transmitted power by node i
$\mathbf{R}$	Routing matrix
r_{ij}	One-hop sequence to go from node i to node j
$\mathbf{S}$	STDMA schedule

$s[\cdot]$	Received signal
$s(t)$	Timing signal at output of a clock
$\mathbf{T}$	Relative traffic load matrix
T_f	Duration of a frame
T_{ij}	Relative traffic load on link (i,j)
T_s	Duration of a time slot
T_{sample}	Duration of a sample
T_0	Reference temperature ($T_0 = 290K$)
T_{offset}	Initial time offset
t	Time scaled axis label
t_i	Time instant i
$x(t)$	Random time deviation
x_{kl}	Random variable if node k has packet to send to node l
w	Weight factor for slot timing adaptation
α	Distance dependent path loss coefficient
Γ_{ij}	Signal-to-interference plus noise ratio in link *(i,j)*
γ_0	Minimum SINR required for correct reception
$\mathbf{\Delta t}$	Slot timing vector
$\mathbf{\Delta t^{(n)}}$	Slot timing vector at time slot n
$\mathbf{\Delta t^{(0)}}$	Initial slot timing vector
$\overline{\mathbf{\Delta t}}$	Mean of slot timing vector
$\widetilde{\Delta t_{ij}^{(n)}}$	Time offset estimation between node i and node j at time slot n
$\Delta t_j^{(n)}$	Slot timing of node j at time slot n
Δt_j	Slot timing of node j
Δt_{ij}	Time offset between node i and node j
$\Delta t_{j,new}$	New slot timing of node j
$\Delta t_{j,old}$	Old slot timing of node j
$\Delta\nu$	Frequency offset from value ν_{nom}
$\epsilon(t)$	Random process modeling nodes' clock drift
θ_{ij}	Angle to node j as seen from node i
λ	Average total external traffic load
λ_i	Average external traffic load on node i
λ_{ij}	Average traffic load through link (i,j)
$\nu(t)$	Instantaneous frequency
ν_{nom}	Nominal frequency
$\sigma^2_{\Delta t}(t)$	Variance of slot timing at time t
$\overline{\sigma^2_{\Delta t}(t)}$	Estimate of the expectation of variance of slot timing at time t
$\sigma_x(\tau)$	Allan deviation of process $x(t)$ with values equally spaced τsec.
$\Phi(t)$	Total instantaneous phase
$\varphi(t)$	Random phase deviation
Ψ_l	Correlation function at sample l
(i,j)	Unidirectional link from node i to node j
(S,D)	Packet from source S to destination D

Chapter 1

Introduction

1.1 General Background

Tactical communication should support the operational units in the field, and must therefore reflect the strategy of the forces. A flexible threat reaction demands very mobile units which may be spread over a large geographical area. If the forces are to operate under a centralized management and at the same time retain their mobility, heavy demands are put on the communication system. These demands will be in the form of security, survivability and protection against electronic warfare [1]. It seems plausible that only a distributed approach will fulfill the requirements of these kind of networks.

In the future battlefield the amount of information that is being transported and processed will be tremendous. The combat information will flow from the lowest combat units to the highest rank in the military hierarchy, guiding vehicle commanders and even individual soldiers in a much more detailed way than before [2]. It is expected that these requirements in the near future must carry digitized voice and various forms of formatted as well as unstructured data with widely differing data rates, traffic characteristics, and *real time* demands.

The classical solutions for these types of scenarios are to use distributed multi-hop packet radio architectures. This approach combines robustness and autonomous operation. Traffic can be quickly re-routed to handle topological changes, and high reliability can be achieved by sending packets on several paths simultaneously.

In order to meet the requirements in our system, STDMA schemes have been proposed, where transmissions in individual links are scheduled to avoid interference [2, 3].

A solution where some Communication Control Nodes (CCN) are in control of the link schedule generation and distribution can be studied. When a network is being deployed, say after a parachute drop into unknown territory, two types of problems can be found

1. How many CCNs are required and how does a node recognize that it should be a CCN?
2. How can (approximate) slot synchronism be achieved in the network with several CCNs?

In this thesis the focus is on the latter.

1.2 Motivation

Why do *ad hoc* networks need to be synchronised?

In recent years, interest has grown in *ad hoc* networking, which allows devices to establish communication, anytime and anywhere without the aid of a central infrastructure. So far, Packet Radio Networks[1] (PRN) have mainly been considered for military applications, where a decentralized network configuration is an operative advantage or even a necessity.

Time synchronisation is a critical piece of infrastructure for any distributed system, and *ad hoc* networks are not an exception. Obviously, a distributed network will also require a decentralized slot synchronisation algorithm to be consistent with the network's architecture.

We define slot synchronisation, as it will be used in this thesis, as the process in which a station aligns itself in time, with other stations, in order to enable Time Division Multiple Access to the shared channel.

Medium Access Control (MAC) protocols define rules for an orderly access to the jointly shared wireless medium. In slotted TDMA schemes, the channel is structured in time slots of fixed length and data packet transmissions always start at the beginning of a new slot. Of course, these schemes are of special interest because of the natural mechanism it provides for refereeing the access to the medium: time. To consider timing inaccuracies and different propagation delays a guard period is usually introduced between two adjacent time slots and therefore perfect slot synchronisation is not necessary [4]. MAC protocols are frequently designed to achieve high channel utilization and low message latency, for instance, STDMA as an evolution of TDMA. Good time synchronisation is therefore important because it shortens the guard time enabling bigger packet sizes to be transmitted, but also easy because each received packet imparts information about the sender's clock. This information can be used to frequently re-synchronise a node's clock with those of its neighbours [5].

In *ad hoc* networks, existing time synchronisation methods will need to be extended and perhaps combined in new ways in order to provide service that meets the needs of the TRAN environment.

Our proposed work is to evaluate slot synchronisation in STDMA tactical radio networks and explore the performance when varying different system parameters.

1.3 Problem Definition

In any time-slot-based MAC protocol, such as TDMA or S-ALOHA, synchronisation is an important and sometimes critical aspect to take into consideration. Moreover, the deployment and set-up of an *ad hoc* network is also a challenging process. It is believed that the synchronisation phase should come in an early stage when deploying a network of these characteristics. Some approaches in the literature assume a fully operating network when performing synchronisation tasks. In this thesis a much more realistic scenario is proposed considering a fully distributed network in its set-up phase. For this purpose, an existing synchronisation scheme will be adopted and evaluated in this environment.

Ad hoc networks must be highly mobile, survivable and versatile. Moreover, topological changes have to be handled by the network. It seems plausible that the initial distribution of nodes over a certain area and the feasible links that they form will affect the network set-up process and consequently the synchronisation procedure. The behaviour of the synchronisation

---

[1] When developing IEEE 802.11 — a standard for wireless local area networks (WLAN)— the Institute of Electrical and Electronic Engineering (IEEE) replaced the term Packet Radio Network with *ad hoc* network. PRN had come to be associated with military or rescue operations, and by adopting a new name, the IEEE hoped to indicate an entirely new deployment scenario [6]. This thesis will, however, use both terms indifferently.

algorithm when topology factors are changed will be studied and analysis of the algorithm's parameters will be done in order to ensure certain stability and convergence speed.

1.4 Previous Work

Only few approaches for time synchronisation of *ad hoc* networks have been published so far. Moreover, the terms synchronisation strategies, STDMA and TRAN are rarely exposed in a same document, but are covered, specially the two last ones, in a large number of separate scientific papers. Therefore, in order to meet the requirements of this thesis, many sources relating these individual topics have been consulted to come up with a global knowledge of the matter of concern.

Regarding tactical radio networks and future military systems, reports from the Swedish Defence Research Agency (FOI) and the Defence Research Establishment (FOA) [7, 8, 9, 10] were particularly useful so as to understand tactical radio access networks, their possibilities and problems. In [2], the requirements for future wireless tactical networks were addressed. Studies showed that distributed control, short range, multi-hop store-and-forward architectures have definite advantages both with respect to reliability, capacity and power consumption.

When investigating the MAC protocol STDMA, a reference paper is [11], where the STDMA concept was introduced. The broadcast channel was defined and methods for determining slot allocations were developed. Some approximate solutions were given for determining the assignment capacity for the links of the network that minimized the average delay of messages in the system.

Many documents in the literature refer to the performance of STDMA in different multihop networks. Moreover, one of the key problems in STDMA, to find efficient distributed algorithms for STDMA scheduling, is also the topic of many papers and theses. Numerous papers have been studied in order to get an insight in STDMA-based networks, but perhaps the most relevant for the work here presented were [12], [13] and [14]. In [12], Grönkvist compares node-assigned versus link-assigned schedules, and proposes a novel assignment strategy called LET. Somarriba, compares in [14], the performance for two MAC protocols, Slotted ALOHA and STDMA, in PRN in rough terrain. This scenario, rough terrain, can be considered also for tactical environments, so parameters and system models will be adopted from this work. Finally, in [13], smart antennas are introduced and evaluated for different MAC protocols. Moreover, interesting routing and scheduling strategies are also described and were also adopted in the thesis.

As for slot synchronisation, in [15], it is suggested to exploit the existence of a globally known time information coming from the Global Positioning System (GPS) for slot synchronisation. A mutual decentralized slot synchronisation scheme is proposed in [16]. There, synchronisation is achieved by measuring the power of pulses of other nodes as well as the time difference between other pulses and its own pulse. By using this information, the node can shift its transmission pulse to-wards a weighted average of other transmission pulses. Another decentralized slot synchronisation scheme is presented in [4] and will be the one used in our simulations. This synchronisation scheme deals with two groups of individually synchronised vehicles that merge together on a highway scenario. Our purpose is to prove that the same scheme can be applied in a network set-up environment. Moreover, some parameters related to the synchronisation algorithm will be evaluated and some conclusions will be made upon them.

1.5 Thesis Outline

This thesis is composed of five chapters and one appendix. In chapter 2 the system models are described. In chapter 3 some notions about synchronisation and synchronisation strategies are elucidated. The performance measure and simulation results are presented in chapter 4. Finally, chapter 5 deals with the conclusions of this thesis and future work to be done.

Chapter 2

System Models

This chapter introduces the models that will be used for the performance analysis. Section 2.1 gives a short insight in tactical environments and the requirements that a network should fulfill in order to be used in such scenario. Some models are evaluated in the following sections to provide an easy way to analyse results in our simulations.

2.1 Environment Description

Tactical requirements demand that the TRAN should be flexible to different types of command and control structures. Furthermore, a TRAN must be able to be autonomous, that is, it can be deployed in a remote area where no core network is available. To secure robust command and control, the network administration should be distributed as much as possible. If all infrastructure is destroyed (or simply not available) communication between units within good radio coverage should be possible. A flexible network should not only support point-to-point communication but also efficient broadcasting and multi-casting [8].

A TRAN could embrace everything from a small group of soldiers to some joint brigades, deployed in areas of 5 × 5 km. for the first case and 100 × 100 km. for the latter. The communication demands will also vary considerably and the TRAN must be able to handle applications with different QoS requirements. The requirements include real-time transmission for voice and video, and non real-time transmission for large data files. In many presumed combat situations one can foresee the need for extremely reliable and fast communications [9].

To fulfill all these requirements a distributed multi-hop network is suggested. Such a network can work without (or at least with a minimum of) network planning. Network management functions can be distributed and are therefore robust against hostile attacks. Traffic can quickly be re-routed to handle topological changes. With the multi-hop function we get a well-connected network with short communication links, such that the power radiated as well as the power consumption is kept low. Urgent messages can be sent on several paths simultaneously to ensure high reliability [8].

All these features make Multi-Hop Packet Radio Networks (MHPRN) or *ad hoc* networks, a strong candidate for its utilization in TRAN, and will therefore be used in this thesis.

2.2 Path Loss Model

It is well known that radio signals are attenuated when they propagate from a transmitter station to the receiver station. The main variation of the signal strength due to distance factors is described by a *path loss* term. The path loss states the relation between the emitted power and the received signal strength for a given separation of the transmitter and the receiver. The prediction of the path loss can be made using well-known radio propagation models and therefore it is possible to estimate the propagation loss between each pair of links (i, j). Lets introduce here the notation of propagation loss, L_{ij}, as the ratio between the power of the emitted signal $P_{transmitted}$ at node i and the power of the received signal $P_{received}$ at node j. If we express it in decibels we have:

$$(L_{ij})_{dB} = 10 \cdot \log_{10} L_{ij} = 10 \cdot \log_{10} P_{transmitted} - 10 \cdot \log_{10} P_{received} \tag{2.1}$$

We can further define the inverse of the propagation loss in link (i, j) as the power gain G_{ij} expressed in terms of L_{ij} as [13]:

$$G_{ij} = (L_{ij})^{-1} \tag{2.2}$$

Once G_{ij} is defined it is convenient to express the power gain between each pair of nodes in the network by means of the so-called gain matrix $\mathbf{G}$. We can describe it as $\mathbf{G} = \{G_{ij}\}$, where, as we know, G_{ij} is the power gain between node i and node j. Although the path loss can be expressed as a contribution of different losses, such as free-space loss, plane-earth propagation loss, etc. in this thesis we will consider a simple distance-dependent path loss model expressed in dB as:

$$(L_{ij})_{dB} = 10 \cdot \log_{10} L_{ij} = 10 \cdot \log_{10}(d_{ij}^{\alpha}) \tag{2.3}$$

Where d_{ij} is the distance between node i and node j, and α is the distance dependent coefficient. A value of $\alpha = 3$ that may correspond to a rural scenario will be used in our simulations [17, page 329].

2.3 Connectivity Model

We will consider a network with N nodes distributed over a 100×100 km. area. Nodes will be either *connected* or *disconnected* depending on the propagation loss between the nodes. A very simple connectivity model is assumed. If the propagation loss in a given link (i, j) is larger than some *threshold path loss,* L_0, communication cannot be sustained. The *threshold pathloss* is then defined as the maximum path loss for which a communication link may be established in the absence of all interference.

An important parameter when analyzing multi-hop ad hoc networks is the *network connectivity.* In this thesis, we consider that the networks are connected, i.e. there is always a finite path with finite number of hops between any pair of nodes in the network. There are two important parameters that indicate the level of connectivity of a network: The average number of neighbours in the network and the average number of hops to reach every other node.

We say that a node i is a neighbour of node j if a feasible link exists between them or that node i is at a single hop from node j. Thus, if we denote the total number of neighbours to node i as $\mathcal{N}_i$, the average number of neighbours is given by

$$E[neighbours] = \frac{1}{N}\sum_{i=1}^{N}\mathcal{N}_i \tag{2.4}$$

In a fully connected network, i.e. every node is connected to each other node, we have

$$E[neighbours] = N - 1$$

On the other hand, the average number of hops is also an important parameter as an indicator of the number of relying nodes that a packet has to go through to reach to its destination. If the minimum number of hops needed to reach node j from node i is h_{ij}, then the average number of hops is given by [13]

$$E[hops] = \frac{1}{N(N-1)}\sum_{i=1}^{N}\sum_{j=1}^{N} h_{ij} \ . \tag{2.5}$$

2.4 Multiple Access Interference Model

In our simulations, it is assumed that a packet arrives at its destination if the Signal-to-Interference plus Noise Ratio (SINR) during the transmission period of the packet is above a specified threshold γ_0 as defined by [13]

$$\Gamma_{ij} = \frac{P_i G_{ij} A_i(\theta_{ij}) A_j(\theta_{ji})}{\sum\limits_{\forall\, link\,(k,l)\neq link\,(j,i)} P_k G_{kj} A_k(\theta_{kj}) A_j(\theta_{jk}) x_{kl} + P_{Noise}} > \gamma_0 \tag{2.6}$$

where

Γ_{ij} is the SINR for a packet sent from node i to node j.

P_i is the transmission power level at node i.

G_{ij} is the power gain between node i and node j.

$A_i(\theta_{ij})$ and $A_j(\theta_{ji})$ are horizontal antenna patterns used by node i and node j respectively.

$P_{Noise} = kT_0BF_{sys}$ is the background noise power level at node j.

$k = 1.38 \cdot 10^{-23}$ J/K is the Boltzmann constant.

$T_0 = 290$ K.

F_{sys} is the receiver's Noise Factor

B is the receiver's equivalent noise bandwidth

x_{kl} is a random variable defined as

$$x_{kl} = \begin{cases} 1 & if\, k\, transmit\, to\, node\, l \\ 0 & otherwise \end{cases} \tag{2.7}$$

The random variable x_{kl} depends on the MAC protocol being used and if whether or not a node k has a packet to transmit to node l. When using STDMA as a MAC protocol (see section 2.5) a *link transmission schedule* is created in advance. The time domain is divided into slots, each long enough to contain one data packet. The link schedule is created by adding one link at a time to test if all links assigned to transmit in a given slot satisfy (2.6) with $x_{kl} = 1$, i.e. considering the worst case when a node k has always a packet to send to node l. We define $\mathcal{L}_n$ as the set of links (i, j) being tested to transmit during slot n (2.8)

$$\Gamma_{ij} = \frac{P_i G_{ij}}{\displaystyle\sum_{\substack{\forall\, link\,(k,l) \in \mathcal{L}_n \\ link\,(k,l) \neq link\,(i,j)}} P_k G_{kj} + P_{Noise}} > \gamma_0 \tag{2.8}$$

where we have assumed omni-directional antennas [13].

2.5 Medium Access Control

MAC protocols are another important issue to take into consideration when designing a system. How to avoid or resolve conflicts due to simultaneously transmitting radio units is treated by the MAC.

Traditionally, MAC protocols for *ad hoc* networks are based on contention-based access methods, i.e. a user attempts to access the channel only when it has a packet to send. Thus, the user has no specific reservation of a channel [12]. This has clear advantages when the traffic is unpredictable. An example of these contention-based access methods are Carrier Sense Multiple Access (CSMA) protocols [18] and ALOHA protocols [19].

ALOHA is the simplest contention MAC protocol where a node transmits randomly without verifying if the channel is either idle or busy. Due to this completely uncoordinated functionality, packet collisions may occur frequently under moderate traffic loads resulting in a poor channel utilization. An improved version of ALOHA, called Slotted-ALOHA (S-ALOHA), where the time space is divided into slots and nodes randomly decide whether to transmit or not during each slot was proposed. By this mechanism the vulnerability period of a packet to be successfully received is reduced to a single slot, doubling the capacity with respect to ALOHA [13].

In CSMA, each user monitors the channel to see if it is used, and only if it is not, the user will be able to transmit. However, this is done at the transmitter while collisions appear at the receiver. This can lead to the so-called hidden terminal problem, i.e. the lack of a node hearing all other nodes in the network [13].

Bursty data traffic is better handled with contention-based schemes, such as ALOHA or CSMA, but, offer limited functionality for handling priorities and QoS requirements [9]. One of the most important QoS parameters in many applications that are specifically sensitive to the MAC are the delay guarantees.

One approach where delay bounds can be guaranteed is TDMA, i.e. time is divided into time slots which are assigned to a user. Unfortunately, in sparsely connected networks, which is the case we are dealing with, this is usually inefficient. Due to the large path losses between distant users, time slots can often be shared by more than one user without conflicts [12].

An alternative for achieving both high capacity and delay guarantees is to use STDMA [11], where the capacity is increased by spatial reuse of the time slots and delay bounds can be guaranteed. Therefore, in this thesis the use of STDMA as the MAC protocol will be considered.

2.5.1 Spatial Time Division Multiple Access

In order to avoid collisions, deterministic transmission schedules such as STDMA [11] have been proposed. In these schemes, transmission schedules are coordinated in such a way that no conflicts occur. STDMA defines a repeating transmission schedule, a *frame*, which contains a fixed number of slots, with each slot being assigned to a unique set of non-conflicting links. In STDMA, radio terminals are allowed to use the same time slot when interferences caused are small, i.e. a time slot can be shared by radio units geographically separated so that small interference is obtained.

Key problems in STDMA are: to find efficient distributed algorithms for STDMA scheduling, slot synchronisation and to handle mobility.

2.5.2 Routing Algorithms

In a multi-hop PRN, the function of the routing algorithm is to guide the packets through the radio network to their proper destination. Connections are made in such a way that between any pair of nodes, there is at least one path available. This way, a packet might need to be relayed from node to node to reach its final recipient. Moreover, if a node has more than one outgoing link, for each outgoing packet, it must be able to make *routing decisions* for the packet to be relayed [14]. A few terrain adaptive routing schemes such as the Minimum Hop Algorithm (MHA), Minimum Maximum Path Loss Algorithm and the Minimum Interference Algorithm were defined and investigated in [20]. The studies carried out revealed that the expected number of hops increase with the roughness of the terrain. The Minimum Hop and the Minimum Interference algorithms exhibited a similar network delay, both being better than the Minimum Maximum Path Loss Algorithm. The MHA uses static table-based routing where a node consults a table to select the outgoing link on which the packet is to be sent. The routing table can be expressed as an $N \times N$ matrix, $\mathbf{R}$, with elements r_{ij} each one indicating the next hop node, predetermined by the MHA, in order to go from node i to node j [14].

2.5.3 Scheduling

STDMA is a MAC protocol where conflicts are avoided by assigning node transmissions into a repetitive pattern of slots of finite length called *schedule*. Therefore, the schedule can be represented as a frame containing the assigned slots for each node or link transmission. We can now introduce the concept of *frame duration*, T_f, as

$$T_f = N_f \cdot T_s$$

where N_f is the number of slots in a frame and T_s is the duration of a time slot.

The following figure (figure 2.1) illustrates these terms.

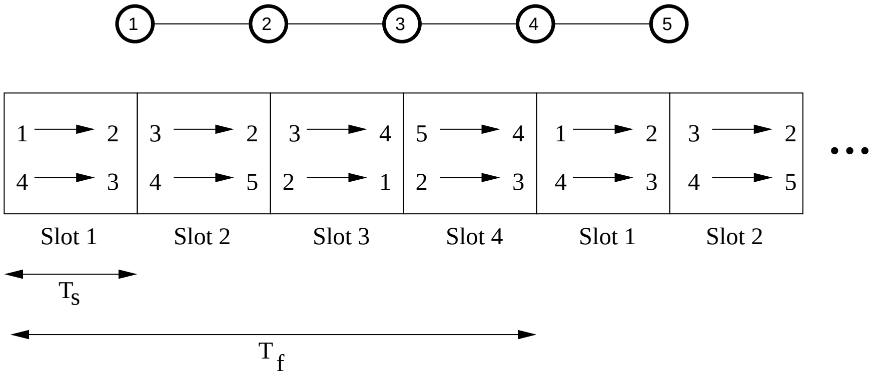

Figure 2.1: Tandem network with $N = 5$ nodes and representation of the schedule for $N_f = 4$ slots.

There are two commonly used slot assignment methods in STDMA: *Link Assignment* and *Node Assignment* [12]. In *Node Assignment* a given node i is allowed to transmit on a slot n to any of its neighbour nodes. Other nodes will be able to transmit in the same time slot n as long as they do not produce mutual interference with other transmitting nodes. This makes *Node Assignment* strategies more suitable for multicast traffic (i.e. from one source to many destinations) when using omni-directional antennas [12].

On the contrary, *Link Assignment* is a link-orientated method where links are assigned to transmit on a given slot within the schedule. For example, if a given link *(i,j)* is assigned to transmit on slot n, it means that node i is allowed to transmit only to node j in time slot n. Therefore, *Link Assignment* seems more suitable for unicast traffic (i.e. from one source to a single destination). Figure 2.1 depicts a clear example of how a link assignment strategy works.

Avoiding interference with other sending nodes is not the only problem that STDMA scheduling techniques deal with. When traffic is unevenly distributed, due to limited connectivity or poor route selection, the performance of the network could be seriously affected. Thus, schedule algorithms that take traffic distribution into account are also studied. These algorithms are commonly referred as Traffic Controlled or Traffic Sensitive algorithms [14, 12, 21].

Therefore a Minimum Hop Algorithm plus Traffic Controlled Scheduling (MHA + TCS) seems a suitable election for a TRAN environment. By defining $\mathcal{L}_n$ as the set of links allowed to transmit in slot n, the schedule $\mathbf{S}$ is defined as a set of $\mathcal{L}_n$, for $n = 1, 2, \ldots, N_f$.

To provide an "optimum" solution for the traffic controlled schedule algorithm is an NP-complete problem [12], therefore heuristic schedule algorithms are commonly used.

2.6 Traffic Model

The multi-hop *ad hoc* network conveys information by means of short bit strings called packets. If a message generated by an information source attached to a node is larger than a packet size, this is broken into several packets to be transmitted through the network as individual entities. Here we consider that packets arrive to the network as a random process and that these messages are of equal size. For simplicity, it is assumed that packets arrive according to a Poisson process with total external traffic load of λ packets per packet duration, being one slot the duration of a packet. Furthermore, it is assumed that the traffic is evenly distributed.

$$\lambda_i = \lambda / N \quad i \in \{1, 2, \cdots, N\} \tag{2.9}$$

where λ_i is the external traffic load on node i.

The initial source (S) and final destination (D) of a packet is denoted by an (S,D) pair. Due to the store-and-forward mechanism, packets between (S,D) pairs may travel through intermediate nodes. Therefore, the average traffic load λ_{ij} going through a link (i, j) is the result of external and internal traffic [13].

$$\lambda_{ij} = \sum_{\substack{\forall (S,D)\, routed \\ through\, link\, (i,j)}} \frac{\lambda}{N\,(N-1)} = \frac{\lambda}{N\,(N-1)} T_{ij} \tag{2.10}$$

where T_{ij} are the elements of the *Relative Traffic Load* matrix **T** given by

$$T_{ij} = \sum (S, D)\, routed\, through\, link\, (i,\, j). \tag{2.11}$$

The relative traffic load matrix is particularly useful for the calculation of traffic-sensitive schedules [13].

Chapter 3

Synchronisation Strategies

3.1 Introduction

In this chapter some notions regarding synchronisation and synchronisation techniques are presented. First, definition of slot synchronisation and synchronisation metrics are showed. In section 3.2, the synchronisation algorithm to be used in the simulations is depicted. Section 3.3 introduces the concept of Communication Control Nodes (CCN) in a distributed network and some of their applications in synchronisation schemes are elucidated.

3.1.1 Slot Synchronisation

As mentioned previously, the STDMA system shares the data link by dividing time into *slots* (Time Division Multiple Access). A set of slots is grouped in a *frame.* We can define now the term *synchronisation*, as it will be used in this thesis, as:

> *The process, in which a station aligns itself in time, with other stations, in order to enable Time Division Multiple Access to the data link.*

A very accurate source for slot synchronisation is the 1 Pulse Per Second (PPS) pulse from a Global Navigation Satellite System (GNSS) receiver. Other sources can also be used, such as *semaphore* stations which provide timing information at some specific intervals [22]. Slot synchronisation should not be confused with other types of synchronisation such as, e.g. bit synchronisation needed in the demodulation/decoding phase of a digital transmission.

The main objective of a slot synchronisation algorithm should be to achieve *one* common slot timing for *all* nodes within the network. However, sometimes it is desirable to achieve a *locally* common slot timing of nodes within their respective range of influence instead of aiming at a globally common slot timing of the whole network [4]. To consider timing inaccuracies and different propagation delays of the individual stations, a guard period is usually introduced between two adjacent time slots and therefore *perfect time synchronisation is not necessary.* Therefore, the goal of the proposed synchronisation system is to establish a common slot timing for all nodes with an acceptable error. The guard time is also necessary to allow clock jitter in transmitters and receivers.

Table 3.1: Guard band for different communication systems [23, 24].

System	Guard Band Between Transmissions (% of slot duration)
GSM	5%
IS-54	2%
UTRA TDD	3.75%

3.1.2 Synchronisation Metrics

A set of five metrics that are useful for characterizing time synchronisation in ad hoc networks are presented. These metrics are [5]:

- **Maximum Error:** Either the dispersion among a group of peers (or nodes), or a maximum deviation of the members from an external standard.
- **Lifetime:** Which can range from persistent synchronisation, i.e. the synchronisation algorithm is operative as long as the network is up, to nearly instantaneous, that is, synchronisation tasks are performed when single events occur.
- **Scope and Availability:** The geographic span of nodes that are synchronised, and the completeness of coverage within that region.
- **Efficiency:** The time and energy expenditures needed to achieve synchronisation.

The services provided by different time synchronisation methods fall into many different points in this parameter space. All of them make trade-offs; no single method is optimal along all axes [5].

In this thesis we will focus mainly on the first criteria, maximum error. In our simulations, maximum error is in the form of *slot timing variance* among all the nodes in the network. Lifetime metric will be persistent, that is, synchronisation will be done as long as the network remains operative. The scope and availability measure and the computational and time cost for algorithms are left for future work.

3.2 Mutual Individual Slot Timing Adaptation Algorithm

The basic idea of the decentralized synchronisation scheme proposed in [4] is to achieve a common slot timing by a mutual adaptation of the individual slot timing. The synchronisation procedure consists of two steps: First, the slot timing of a received burst is acquired in form of a *one-shot synchronisation* [15]. In the second step, the own slot timing is adapted according to the observed time difference to the node that transmitted the respective burst.

3.2.1 Slot Timing Acquisition Phase

The slot timing acquisition procedure is normally independent of what kind of slot adaptation we use after-wards. For this reason, one possible method to retrieve this information is presented in this section and we will consider from now on that acquisition is performed correctly during the the simulations.

In our environment, the TRAN, it is required that each node is able to detect the slot timing of a received data burst without any previous knowledge about the burst timing of the station that transmitted the respective signal. In other words, each station should determine this slot difference based only on local measurements. This procedure is also referred to as *one-shot synchronisation* and here we will adopt a scheme from [15].

For one-shot slot timing acquisition a correlation sequence $m[k]$ of length K is transmitted within each time slot. The sequence is chosen to have noise-like autocorrelation properties and can be placed either at the beginning or in the middle of the burst.

During every time slot without own transmission, each node j correlates its received signal $s[l]$ with the commonly known correlation sequence $m[k]$:

$$\Psi_l = \sum_{k=0}^{K-1} s[l+k] \cdot m^*[k] \tag{3.1}$$

By finding the argument l that maximizes the absolute value of Ψ_l within the own slot boundaries, the observed time offset Δt_{ij} at node j from an arbitrary node i that transmitted during the current time slot can be estimated by

$$\Delta t_{ij} = T_{sample} \cdot (\arg\max_l |\Psi_l| - n_0) \tag{3.2}$$

where n_0 is the fixed position of the correlation sequence within the time slot and T_{sample} is the duration of a sample.

In the case that more than one node is transmitting a burst within the synchronisation range of node j, simultaneously, the maximum correlation value in equation (3.1) corresponds to the *strongest* (closest) node i within the synchronisation range of node j.

3.2.2 Slot Timing Adaptation

At the end of a slot timing acquisition phase described in the last section, each node adapts its own slot timing Δt_j according to the used algorithm. The selection of one or other slot synchronisation algorithm will affect the performance of the system in terms of speed of convergence and timing accuracy. A synchronisation algorithm proposed in [4] will be used for testing the synchronisation performance of our system.

At the end of a slot timing acquisition phase, each node j adapts its own slot timing Δt_j according to

$$\Delta t_{j,new} = \Delta t_{j,old} + w \cdot \Delta t_{ij} \tag{3.3}$$

where the parameter w denotes a weighting factor that considers propagation delays and oscillator drifts of nodes' clocks. By choosing an appropriate value of $w < 1$, a stable synchronisation can be achieved as we will see later on.

A slot timing adaptation of node j is shown in figure 3.1.

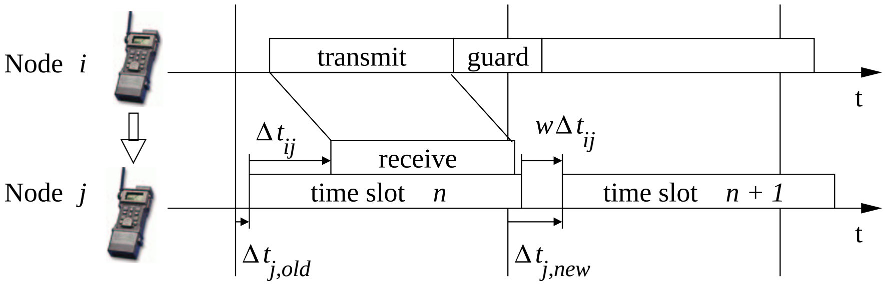

Figure 3.1: Slot timing adaptation for node j.

Within slot n, node j measures a time offset of Δt_{ij} with respect to node i. According to equation (3.3), node j shifts the start position of the next time slot, $n+1$, by $w \cdot \Delta t_{ij}$. Since all nodes take part in the mutual synchronisation procedure, a common slot timing can be achieved.

Next figure (figure 3.2) shows a simple example with a 3-node network:

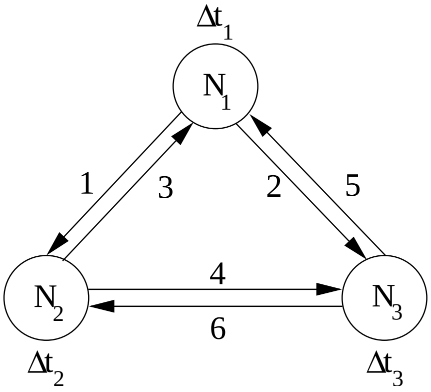

Figure 3.2: Example of a 3-node network.

Where nodes are denoted by N_1, N_2, N_3 and links are numbered from 1 to 6, where we consider link transmissions in a half-duplex fashion. The individual timing of each node i is contained in variable Δt_i . The slot timing is a parameter that indicates the drift of each node with respect to a perfectly synchronised slot scheme. Thus, we define $\mathbf{\Delta t}$ as the slot timing vector

$$\mathbf{\Delta t} = \{\Delta t_1,\, \Delta t_2 \ldots \Delta t_N\}\,.$$

As a performance measure, we use the variance of vector $\mathbf{\Delta t}$ at time t, defined as

$$\sigma^2_{\Delta t}(t) = \frac{\sum_{i=1}^{N}(\Delta t_i - \overline{\mathbf{\Delta t}})^2}{N-1} \tag{3.4}$$

with

$$\overline{\mathbf{\Delta t}} = \frac{\sum_{i=1}^{N} \Delta t_i}{N} \tag{3.5}$$

the mean of vector $\mathbf{\Delta t}$. Throughout the thesis, the estimate of the expectation of $\sigma^2_{\Delta t}(t)$, denoted as $\overline{\sigma^2_{\Delta t}(t)}$, normalized by the initial timing variance will be used as performance measure.

Figure 3.3 shows the flow diagram of the algorithm for a given node j.

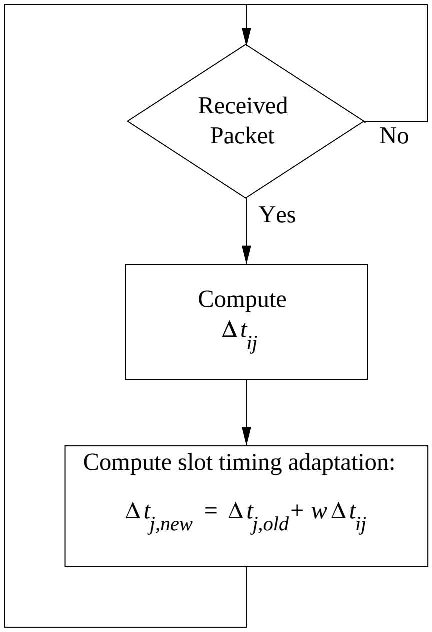

Figure 3.3: Flow diagram for slot timing adaptation of node j.

3.3 Communication Control Nodes and Synchronisation

As already mentioned in chapter 2, schedule generation, distribution and maintenance are crucial activities in an STDMA-based mobile TRAN. A solution where some Communication Control Nodes[1] are in control of the link schedule and routing table distribution is proposed in [8]. Therefore, we can assume two types of radio units: Standard Nodes (SNs) and CCNs. The CCNs maintain and distribute scheduling and routing information, while the actual traffic can be relayed through all nodes, both CCNs and SNs. A CCN may be a vehicle. A soldier with a small radio or a sensor may be a SN. However, it is desirable that any node can act as a CCN. Furthermore, it is also desirable that any node can act as a Gateway (GW), for connection to a Core Network (CN), for instance a cellular UTRAN [8].

In figure 3.4 an application example containing all these network components is illustrated.

---

[1] Sometimes referred as Connection Control Nodes in the literature.

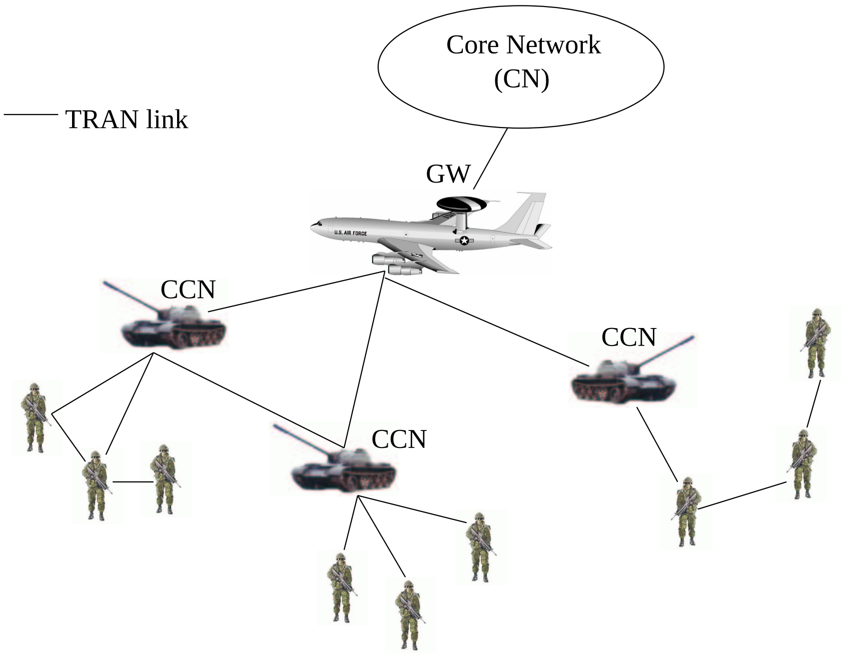

Figure 3.4: Some possible network components in a TRAN

The main reason for using two types of nodes, or hierarchical layers, is that the fully distributed management of network functions may be very difficult and cause a lot of overhead traffic in large networks [8]. For large networks (say more than 100 nodes), the delays can be important. Furthermore, a lot of processing is required in every node to handle routing, mobility management, and channel resource allocation. Thus, the amount of processing is reduced, in the Standard Nodes, by having this second layer, i.e., CCNs.

When a network is being deployed, say after a parachute drop into unknown territory, two types of problems are encountered:

- How many CCNs are required and how does a node recognise that it should be a CCN?
- How can, approximate, slot synchronism be achieved in the network with several CCNs?

How to select a CCN based on connectivity, computational capacity, low mobility etc, is still an open question [8]. The number of existing CCNs in the network should be chosen taking into account factors such as overhead traffic, connectivity, network topology, cost, etc.

From the point of view of synchronisation, it seems plausible that CCNs should play a more active role than SNs. This could imply having CCNs equipped with GPS transceivers in order to transmit *correct* timing information to the rest of the network. Just after the network being deployed, and after connectivity information being exchanged, the CCN (or CCNs) will be in charge of distributing the link schedule and routing tables to the rest of the nodes. They will also, within each packet transmission, send synchronisation information to the other nodes. If a *correct* time reference is available to each of the CCNs, the synchronisation adaptation procedure, described in section 3.2.2, can be more easily done than in a fully distributed non-synchronised network. Furthermore, packets from CCNs could be marked as high priority, this way we give more importance to these packets imparting *correct* timing than the packets generated by SNs. These or other strategies can be used taking advantage of the network hierarchy in order to obtain a higher synchronisation performance.

Chapter 4

Simulation Method and Results

In this chapter simulations are carried out in order to analyse the performance of the synchronisation algorithm described in section 3.2. First, some comments and definitions about the performance measure and the simulation scenario will be made. In section 4.2, simulations results are shown and preliminary conclusions are made .

4.1 Performance Measure and Simulation Setup

4.1.1 Considered Scenario: Network Set-up

A self-organizing network based on radio communications will create its own connections, topology, transmission schedules, and routing capabilities in a distributed manner. It will also change its configuration as the mobile stations change their position and leave and enter the area of coverage. The self-organization process for a number of mobile stations begins with the exchange of connectivity information between nearby (within radio range) stations.

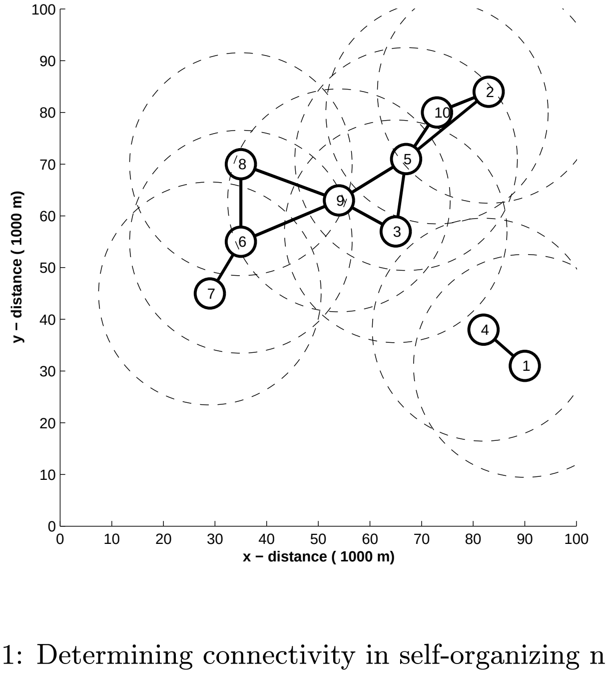

Figure 4.1: Determining connectivity in self-organizing networks.

Figure 4.1 shows a possible scenario where nodes retrieve connectivity information by broadcasting to nodes within their radio range (neighbour nodes).

Organizing starts with an exchange of local connectivity information, either throughout a subset of nodes or the whole network. Here we will consider the latter. Each station has a unique identifying number, which allows it to be paired with individual slots. Stations broadcast in their slot the identifying numbers of stations they have heard during previous slots. Partial connectivity information is communicated during first transmissions until each station knows which stations are within hearing distance [25].

We must notice, however, that this procedure assumes that an accurate time reference system is available to each station. This is, in many cases, not true. For the case that we are dealing with, it is assumed that no common reference time is available and the access to the channel is random. Thus implying that a synchronisation procedure must be carried out at this point, before or during the connectivity discovery phase. Once the network is synchronised, further operations such as transmission schedule generation and routing patterns can be implemented.

The next subsection introduces some basic assumptions that will be used in our simulations.

4.1.1.1 Network Set-up Assumptions

The following statements will be assumed unless otherwise mentioned:

- Although in our network set-up mechanism, explained extensively in [25], it is not necessary to know neither the number of nodes that form the network nor the location of such nodes, it is assumed that our network is composed of N nodes randomly scattered over a $100 \times 100\,km$ area. Each node has a unique identifying number from the set $\{1...N\}$.

- Only *connected* networks are assumed. This means that when the network is up and going, each node should be able to communicate with each other node in the network either by a direct link or by intermediate nodes in a multi-hop fashion. Notice that figure 4.1 represents a *non connected* network, since nodes labeled 1 and 4 cannot reach any other node in the network than themselves.

- A distance-dependent propagation model is assumed, indicating that only stations within a certain transmitting range (neighbour stations) will successfully receive transmissions. This transmission range is depicted in figure 4.1 by dashed circles centered at transmitting nodes.

- Since our synchronisation scheme works on a *synchronise-upon-received-packet* basis, we consider that packets transmitted during this synchronisation phase are actually packets transmitting connectivity information to neighbour nodes. In other words, the synchronisation and connectivity discovery of the network are performed simultaneously.

- Each node broadcasts the connectivity information, and also synchronising information, to each of its neighbours.

- An initially unsynchronized network is assumed. The initial timing of each node, $\Delta t_i^{(0)}$, is defined as the time between the node's first transmission attempt and our time reference and it is actually the measure of how synchronised or unsynchronised is our network (see figure 4.2).

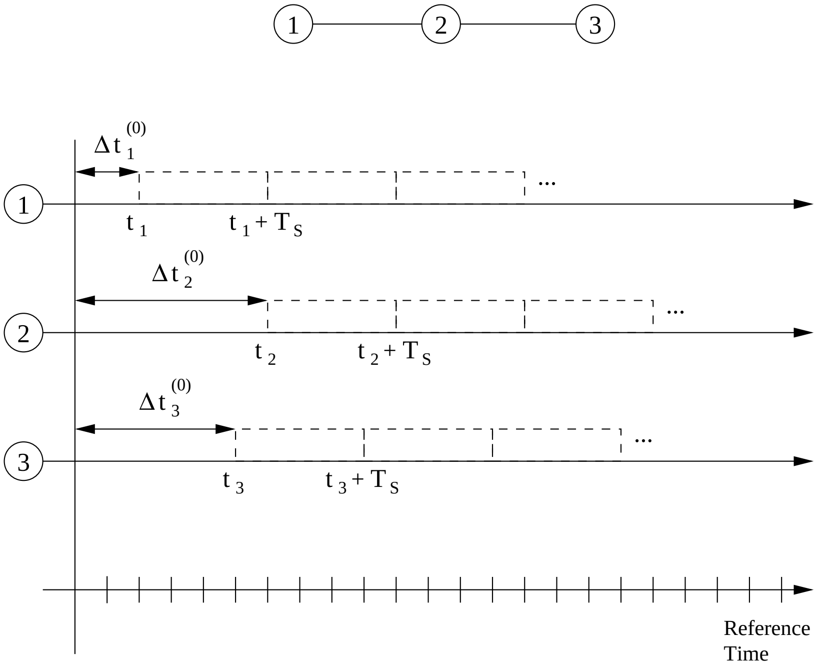

Figure 4.2: Initial timing.

Each node i assumes that the transmitting schedule starts at time t_i with transmitting slots of length T_s . A node will switch on its transmitter/receiver and wait until the first packet is ready to be broadcasted. Each node is aware of the size of a packet, T_S, and therefore, will schedule successive transmissions according to this value and to its own slot boundaries.

- It is assumed that no schedule is available at this time. Therefore, a node will transmit whenever it has a packet to transmit, and it will do so at the beginning of a slot considering its own timing and slot boundaries.

- No routing capabilities are present at this stage. This means that transmissions are done on a single-hop manner. Packets are generated at i and sinked at first-hop node j.

- In a network with no synchronisation between nodes and with no schedule either, collisions may occur. A station can only receive one transmission at a time. Moreover, it cannot receive and transmit at the same time. To solve this problem the following assumptions will be made:
  - All nodes continuously monitor the channel.
  - During the monitoring period, each node j listens for incoming packets and determines the slot offset between transmitter and receiver as explained in section **??**.
  - Timing adjustments at receiving node j will be done according to (3.3). It is important to notice that it is not necessary to have completely received the whole packet in order to perform the timing adaptation. Once the receiving node has correlated its received signal with the known correlation sequence, it is able to adapt its timing according to the measured offset. Of course, the time in which the node will be able to adapt its own timing will be dependent on the length of the correlation sequence and where this

correlation sequence is placed. For simplicity, in this thesis we will assume that this is done just after the packet is being received, meaning one clock after the packet arrives to the destination.
- When the node is ready to transmit in the current slot, it turns off its receiver during the transmission. Therefore, packets received at the node while transmitting will be ignored in both the connectivity and synchronisation process.
A description of conflicts that may occur are listed in table 4.1 together with the pertinent actions that will be considered during the simulations.

Table 4.1: Conflict table.

Conflict	**Action**
1. A station receives more than one packet at the same time through different links.	Only the packet from the closest station is considered for synchronisation. If received packets come from equidistant nodes, only consider one of them by random election.
2. Two stations send packets to each other at the same time.	Packets overlap completely and no synchronisation is performed.
3. A station is transmitting and receiving a packet (or several packets) at the same time.	The incoming packets are not considered for synchronising. The station turns-off its receiver.
4. A station receives a packet (or several packets) while already being transmitting.	The incoming packet(s) are not considered.
5. A station intends to transmit a packet while receiving.	The transmission of the packet is postponed to the next time slot.

- Nodes are assumed to remain statically in the same place while the network is in set-up mode. That is, no mobility is assumed during set-up.

4.1.1.2 Simulation Pseudo-Code

The simulation procedure used in the network set-up is explained through the following pseudo-code:

```
while clock< simulation_time

  clock=clock+1;
  for each feasible link (i,j)

      if i is ready to transmit
          if i has packets to send to j
              %mark i -> j as attempting transmission
          end
      else
          %update transmitting info
      end
```

```
    end
    %solve conflicts, transform attempts into active transmissions
    for each active transmission $i$ -> $j$ at time clock
        %send
        %receive
        %compute offset between $i$ and $j$
        %update receiver's timing
        %update receiver's next transmission time
    end
end
```

Simulations were carried out using MATLAB® and one of its compiler platforms, MEX-Files. The simulations are performed using confidence intervals of 95% for the estimation of the mean.

4.2 Simulations Results

4.2.1 Synchronisation Performance when varying Parameter w

In section 3.2.2 we introduced the slot timing adaptation scheme. Equation (3.3) gives us the timing update procedure that each node performs when it receives a packet. The weighting factor w was also introduced, this value considers propagation delays and oscillator drifts of clocks. w enables a stable synchronisation if an appropriate value is chosen, but what value should it take?

Actually, the value of w depends on the considered scenario. If a value of w very close to 1 is chosen, the convergence of the individual node timings, Δt_i, will be increased, but the control system can become unstable. Indeed, the individual nodes do not measure the actual timing offsets but rather a superposition of the timing offsets and the propagation delay between them. Therefore, we can express equation (3.3) as

$$\Delta t_j^{(n)} = \Delta t_j^{(n-1)} + w \cdot \widetilde{\Delta t}_{ij}^{(n-1)} \tag{4.1}$$

Where $\widetilde{\Delta t}_{ij}^{(n-1)}$ is the time offset estimation between node i and node j at time $n-1$. We can express $\widetilde{\Delta t}_{ij}^{(n)}$ at time slot n as

$$\widetilde{\Delta t}_{ij}^{(n)} = \Delta t_{ij}^{(n)} + \frac{d_{ij}}{c} + \epsilon(n) \tag{4.2}$$

with

$\Delta t_{ij}^{(n)} = \Delta t_i^{(n)} - \Delta t_j^{(n)}$ the *real* time offset between node i and node j.

d_{ij} the distance between node i and node j.

$c = 3 \cdot 10^8 m/sec$ the speed of light.

$\epsilon(t)$ the random process modeling node's clock drifts. (see Appendix A)

So if we increase the factor w and we have large distances, the individual node timings drift away quite quickly, specially if we have partly meshed networks where one node is part of a group with large distances and also part of a group with small distances, this may lead to some problems in convergence. In this case, if the factor w is small, an adaptation to a node with large observed time offset does not necessarily lead to a loss of synchronisation to synchronised nodes within the own group.

If the factor w is large and a time offset (maybe from far away) is received, the node will adapt its own timing and therefore loses synchronisation to nodes within the own group (those nodes may not have received the sync info from the first node, for some reason).

So, to find a good compromise for w, the respective scenario (velocity, transmission range, node density, etc.) should be considered carefully.

To illustrate this behaviour we consider the following scenario shown in figure 4.3.

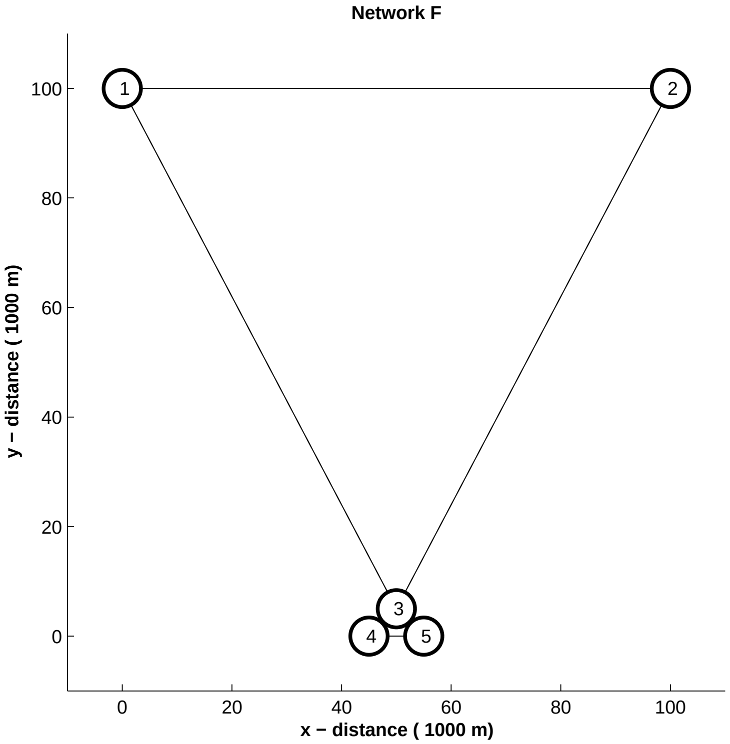

Figure 4.3: Partly meshed network with large distance cluster and small distance cluster.

This network contains a long-distance cluster formed by nodes 1, 2 and 3; and a small-distance cluster formed by nodes 3, 4 and 5.

If we plot one realization of the synchronisation process for values of $w = 0.5$ and $w = 1$ we obtain the following curves representing the normalized timing variance as a function of system time (see figure 4.4)

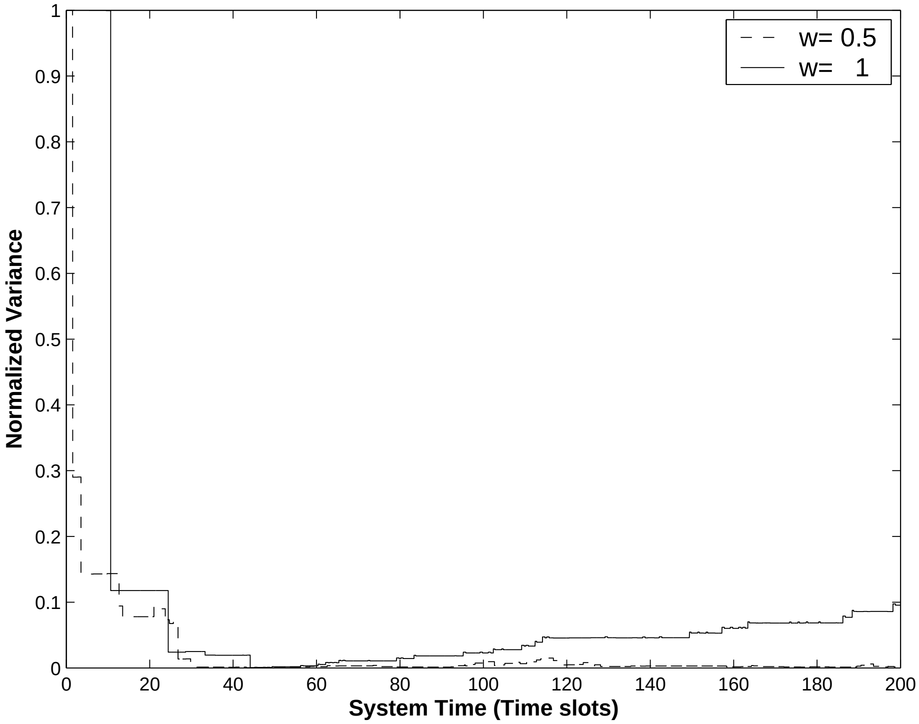

Figure 4.4: Normalized timing variance as a function of system time in one realization of the synchronisation process using network depicted in figure 4.3.

Notice how for $w = 1$, the synchronisation algorithm does not converge completely to zero, on the contrary, the normalized timing variance increases at one point due to the influence of propagation delays from the large-distance cluster.

Some relevant simulation parameters used for the simulation were:

Table 4.2: Used simulation parameters.

Parameter	**Value**
Antenna Radiation Pattern	Omni-directional
Mobility model	*none*
Distance dependent path loss coefficient α	3
Total traffic load λ	0.6 packets/packet duration
Minimum path loss threshold L_0	130 dB
Simulation time	200 time slots
Packet Size	666,67$\mu sec.$
Clock resolution	100 clock steps/slot
FIFO Buffer Length for each Link	100

In [4] and [16] values of $w = 0.1$ and $w = 0.5$ are used respectively with no further justification. To choose an "optimal" value for w seems a quite difficult task. However, at this point we can say that the value of w is highly dependent on the topology, and it is trade-off between convergence speed and stability to chose its value. If the distribution of nodes over a certain area is unknown a priori, a conservative value of w, around 0.5, could be a good starting value in a network set-up scenario. Once the network acquires positioning knowledge, the value of w could be increased or decreased depending on the situation. An implementation of the adaptation algorithm with a different value of w for each node is proposed for future work.

4.2.2 Synchronisation as a Function of Network Connectivity

An important parameter when analyzing multi-hop ad hoc networks is the *network connectivity.* In this thesis, we consider that the networks are connected, i.e. there is always a finite path with finite number of hops between any pair of nodes in the network. We can characterize the network connectivity by means of the number of neighbours to each node.

We say that a node i is a neighbor of node j if a feasible link exists between them or that node i is at a single hop from node j. Thus, if we denote the total number of neighbors to node i as $\mathcal{N}_i$, the average number of neighbors is given by

$$E[neighbours] = \frac{1}{N}\sum_{i=1}^{N}\mathcal{N}_i \tag{4.3}$$

We have already mentioned that the neighbour-discovery process in a network set-up scenario will run parallel to the synchronisation procedure. Therefore, a situation where the network has a high level of connectivity will be beneficial in terms of synchronisation since nodes will exchange information in fewer slots than if a sparse network is considered. Moreover, sparsely distributed networks with low connectivity will be more susceptible to large drifts if high values of w are considered, this may occur, for instance, to nodes which are poorly connected to the rest of the network and therefore receiving packets at scattered non-frequent intervals. These nodes will then update their timing in a rough manner, causing large timing drifts with respect to other *well-connected* nodes.

Simulations are driven considering 10-node networks, each of them with a different $E[neighbours]$ value. In order to evaluate the effect of connectivity, we assume that the links connecting the nodes have the same length, this way we neglect the effect of propagation delays.

Figure 4.5 depicts the synchronisation performance, in terms of *probability(normalized timing variance< threshold_value),* as a function of the mean number of neighbours. This is done for different values of weighting factor w.

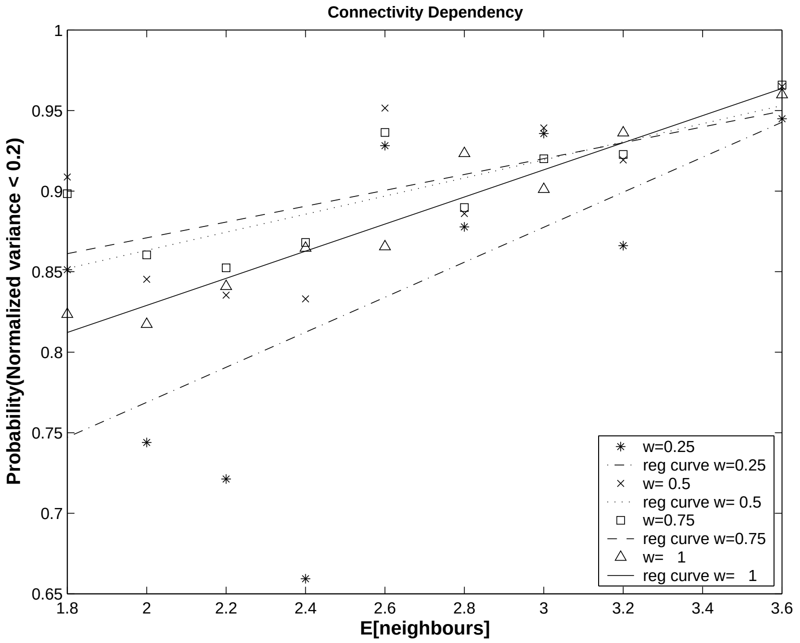

Figure 4.5: Synchronisation performance as a function of network connectivity.

The graph shows how for low connected networks, i.e. low $E[neighbours]$, the election of the value of w will affect the performance of synchronisation. In this case, $w = 0.75$ and $w = 0.5$ produce better results than the other two proposed values for w. On the other side, if we consider networks with high $E[neighbours]$, we observe that the election of w makes almost no difference on the synchronisation performance, being in this last case higher than for low connected networks. The error intervals for the linear fitting of figure 4.5 are shown in the next table (table 4.3):

Table 4.3: Error bars for a 95% confidence interval of regression curves.

	$w = 0.25$	$w = 0.5$	$w = 0.75$	$w = 1$
$E[neighb.] = 1.8$	0.7472 ± 0.2037	0.8520 ± 0.0963	0.8612 ± 0.0613	0.8123 ± 0.0613
$E[neighb.] = 2$	0.7689 ± 0.1956	0.8633 ± 0.0925	0.8710 ± 0.0588	0.8291 ± 0.0613
$E[neighb.] = 2.2$	0.7907 ± 0.1896	0.8745 ± 0.0896	0.8809 ± 0.0570	0.8459 ± 0.0613
$E[neighb.] = 2.4$	0.8124 ± 0.1858	0.8857 ± 0.0878	0.8907 ± 0.0559	0.8628 ± 0.0613
$E[neighb.] = 2.6$	0.8342 ± 0.1843	0.8970 ± 0.0872	0.9005 ± 0.0554	0.8796 ± 0.0613
$E[neighb.] = 2.8$	0.8559 ± 0.1853	0.9082 ± 0.0876	0.9104 ± 0.0557	0.8964 ± 0.0613
$E[neighb.] = 3$	0.8776 ± 0.1886	0.9194 ± 0.0892	0.9202 ± 0.0567	0.9133 ± 0.0613
$E[neighb.] = 3.2$	0.8994 ± 0.1941	0.0918 ± 0.0584	0.9301 ± 0.0584	0.9301 ± 0.0332
$E[neighb.] = 3.6$	0.9428 ± 0.2112	0.0998 ± 0.0998	0.9638 ± 0.0344	0.9638 ± 0.0344

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this thesis we have studied the slot synchronisation performance using STDMA MAC protocol in a TRAN environment. The synchronisation algorithm used in this thesis was chosen based on resource efficient and stable mechanism factors that make them suitable in the considered environment. Resource efficient because there is no need of additional signaling effort and stable because the synchronisation is performed on the basis of user data transmission.

Although the proposed synchronisation scheme was intended for a inter-vehicle communication scenario, we have seen that it is also suitable for an *ad hoc* network set-up environment. Indeed, the deployment of a distributed network is particularly critical in the first steps where there is little information on how many nodes form the network, their location and the feasible links that their form. It has been shown that synchronisation is needed at this early stage for a correct network functionality. A network set-up procedure has been presented, where our synchronisation scheme interacts with the connectivity discovery process which is the first step when deploying a distributed ad hoc network. The synchronisation scheme adds no further complexity to the system due to the *synchronise-upon-received-packet* adaptation model.

Among the main contributions of this thesis we can find a study on the value of weighting factor w, which was not included in previous works. We found a tight dependency of w with topology distribution. Simulations revealed unstable behaviour of the synchronisation algorithm for values of w close to 1 when partly meshed networks were under test. Conservative values of w (those around 0.5) on the contrary, showed a better stability behaviour at the cost of lower speed of convergence. Network connectivity has also proved to be an important indicator in order to chose an appropriate value for w. It was shown that, for low connected networks, choosing an appropriate value for *w can* make a difference. Indeed, a poorly connected network needs to update its timing progressively in order to keep large timing drifts as low as possible. This means that conservative values of w must be used in order to improve the synchronisation performance of the network.

5.2 Future Work

In the thesis we have used the synchronisation scheme in a network set-up environment. This is perhaps the most critical scenario. However, when the network is up-and-running synchronisation techniques are also required. This is because node clocks can drift away in time. Moreover, nodes can suddenly lose synchronisation and therefore misinform other nodes in the network, or new unsynchronised nodes can join the network at some stage. To evaluate the proposed synchronisation algorithm in these scenarios is proposed as future work. Moreover, considering a synchronisation algorithm with different values of w for each node, depending on its location and connectivity, seems also an interesting challenge.

Bibliography

[1] J.E. Rustad, R. Skaug, A. Aasen "New Radio Networks for Tactical Communication". *IEEE Journal on Selected Areas In Communications,* Vol. 8, No. 5, June 1990.

[2] L. Ahlin, E. Englund, C. Jönsson, I. Söderquist, J. Zander, G.Q. Maguire "Autonomous Tactical Communications Possibilities and Problems". Proc MILCOM 97.

[3] E. Englund, J. Grönkvist, J. Nilsson, M. Sköld, U. Sterner, I. Söderquist, J. Zander "Tactical Radio Access Networks based on Future Civilian Communication Architectures". NATO Symposium on Tactical Mobile Communications, Lillehammer, Norway, June 1999.

[4] A. Ebner, H. Rohling, M. Lott, R. Halfmann "Decentralized Slot Synchronization in Highly Dynamic Ad Hoc Networks". Wireless Personal Multimedia Communications, 2002. The 5th INternational Symposium on , Volume: 2, Oct. 27-30,2002. Page(s): 494-498.

[5] J. Elson "Time Synchronization Services for Wireless Sensor Networks". Dissertation Proposal, Department of Computer Science, University of California, Los Angeles, CA, 90095. April 10, 2001.

[6] M. Frodig, P. Johansson, P. Larsson "Wireless *ad hoc* networking — The art of networking without a network". *Ericsson Review*, No. 4, 2000.

[7] J. Grönkvist, U. Sterner, A. Hansson, C. Jönsson, I. Söderquist "Modelling Tactical Mobile Radio Networks Based on Iterference Environment",FOA-R–99-01155-504–SE, June 1999.

[8] J. Nilsson, A. Hansson, U. Sterner, E. Englund, J. Grönkvist, M. Sköld, I. Söderquist, J. Zander "Tactical Radio Access Networks - A Concept Study; Report 2", FOA -R–99-01358-504–SE, December 1999.

[9] A. Hansson, J. Nilsson, M. Sköld, U. Sterner, "Tactical Radio Access Networks - A Comparison of Ad Hoc and Cellular Network Concepts", FOI-R–0086–SE, March 2001.

[10] M. Sköld, U. Sterner "Evaluation of Tactical Radio Networks – Simulation methods and results", FOI Memo 01-1486/L, June 2001.

[11] R. Nelson, L. Kleinrock "Spatial-TDMA, A collision-free Multihop Channel Access Protocol", *IEEE Transactions on Communications*, COM-33(9), Sept. 1985.

[12] Grönkvist, J., "Assignment Strategies for Spatial Reuse STDMA". Licentiate Thesis, Radio Communication Systems, Department of S3. Royal Institute of Technology, Stockholm, Sweden, April 2002.

[13] M. Sánchez G. "Multiple Access Protocols with Smart Antennas in Multihop Ad Hoc Rural-Area Networks". Licentiate Thesis, Radio Communication Systems, Department of S3. Royal Institute of Technology, Stockholm, Sweden, June 2002.

[14] O. Somarriba "Multihop Packet Radio Systems in Rough Terrain". Licentiate Thesis, Radio Communication Systems, Department of S3. Royal Institute of Technology, Stockholm, Sweden, October 1995.

[15] A. Ebner, H. Rohling, R. Halfmann, M. Lott "Synchronization in Ad Hoc Networks based on UTRA TDD" *Proc. PIMRC 2002, Lisbon, Portugal* (2002)

[16] E. Sourour, M. Nakagawa "Mutual Decentralized Synchronization for Intervehicle Communications", *IEEE Trans. On Vehicular Technology* 48, 6 (Nov. 1999), 2015-2027.

[17] J. Zander, S. Kim "Radio Resource Management for Wireless Networks". Artech House, Inc., 2001.

[18] L. Kleinrock, F. Tobagi "Packet Switching in Radio Channels part I - Carrier Sense Multiple-access Modes and their Troughput-delay Characteristics", *IEEE Trans. Commun.*, vol. 23, pp. 1400-1416, Dec. 1975.

[19] L.G. Roberts "ALOHA Packet System with or without Slots and Capture", Comput. Commun. Rev., vol. 5, 1975.

[20] C. Roobol "Multihop Radio Networks in Random Terrains: Connectivity and Some Terrain Adaptive Routing Algorithms", Military Communications Conference, 1993. MILCOM'93. Conference Record. 'Communications on the Move'., IEEE, Volume: 2, 11-14 Oct 1993, Page(s): 428-431 vol.2

[21] T. Robertazzi, J. Shor "Traffic Sensitive Algorithms and Performance Measures for the generation of Self-Organizing Radio Network Schedule", *IEEE Transactions on Communications,* COM-41(1), January 1993.

[22] R. Kjellberg "Capacity and Throughput using a Self Organized Time Division Multiple Access VHF Data Link in Surveillance Applications". Master Thesis, Department of Computer and System Sciences, University of Stockholm and The Royal Institute of Technology, Stockholm, Sweden, GP&C Sweden AB, April 1998.

[23] 3GPP. Physical channels and mapping of transport channels onto physical channels (TDD). Technical Specification Group Radio Access Network, 3G TS 25.221 V4.2.0, Sept. 2001.

[24] 3GPP. Multiplexing and Multiple access on the radio path (GSM). Technical Specification Group GERAN, 3G TS 05.02 V8.7.0, Nov. 2000

[25] T.G. Robertazzi and P.E. Sarachik "Self-organizing Communication Networks" IEEE Communications Magazine, 24(1), pages 28-33, January 1986.

[26] S. Bregni, "Measurement of Time and Frequency Stability in SDH Equipment Clocks: Methodology and Experimental Results" WSEAS Transactions on Communications, Issue 1, Vol. 1, 2002. ISSN 1109-2742.

[27] D. Mills, “Network Time Protocol Version 3 - Specification, Implementation and Analysis”, RFC 1305, UDEL, March 1992.

[28] M. A. Lombardi, “The Measurement, Instrumentation and Sensors Handbook”, Chapter 19, “Frequency Measurement”. CRC Press LLC 1999 ISBN 0849383471.

Appendix A

Clock Model

In telecommunications a clock is a device able to supply a timing signal, ideally periodic, usable for the control of telecommunication systems. In our case, the clock is needed to synchronise the access to the medium in a ordered fashion.

A general expression describing a pseudo-periodic waveform which models the timing signal $s(t)$ at the output of a clock is given by

$$s(t) = A\sin(\Phi(t)) \tag{A.1}$$

where A is the signal's amplitude and $\Phi(t)$ is the *total instantaneous phase*.

A well-known definition of *instantaneous frequency* can be expressed as

$$\nu(t) = \frac{1}{2\pi}\frac{d\Phi(t)}{dt} \tag{A.2}$$

A common model used to characterize $\nu(t)$ is given by [26]

$$\nu(t) = \nu_{nom} + \Delta\nu + D\nu_{nom}t + \frac{1}{2\pi}\frac{d\varphi(t)}{dt} \tag{A.3}$$

where

ν_{nom} is the *nominal frequency.*

$\Delta\nu$ represents the *frequency offset* from value ν_{nom}.

D is the *linear fractional drift rate*, also called *ageing rate.*

$\varphi(t)$ is the *random phase deviation.*

By using (A.2) and (A.3) the total instantaneous phase $\Phi(t)$ can be expressed as

$$\Phi(t) = 2\pi\int_{t_0}^{t}\nu(\tau)d\tau = 2\pi\int_{t_0}^{t}[\nu_{nom} + \Delta\nu + D\nu_{nom}\tau + \frac{1}{2\pi}\frac{d\varphi(\tau)}{d\tau}]d\tau$$

yielding

$$\Phi(t) = 2\pi[\nu_{nom} + \Delta\nu]t + \pi D\nu_{nom}t^2 + \varphi(t) + \Phi_0 \tag{A.4}$$

where Φ_0 is the initial phase offset.

Defining the *time function* $C(t)$ generated by the clock as

$$C(t) = \frac{\Phi(t)}{2\pi\nu_{nom}} \tag{A.5}$$

and substituting (A.4) into (A.5) we obtain

$$C(t) = t + \Delta\nu t + \frac{D}{2}t^2 + x(t) + T_{offset} \tag{A.6}$$

where

$$x(t) = \frac{\varphi(t)}{2\pi\nu_{nom}} \tag{A.7}$$

is called the *random time deviation* and is a factor that models oscillator intrinsic phase noises sources; and T_{offset} is the *initial time offset*.

In the usual stationary model $\Delta\nu$ and D can be assumed constant or changing slowly with time. The random nature of the clock is characterized by $x(t)$. In the usual analysis second-order term D is ignored and the noise term $x(t)$ modeled as a normal distribution with predictable spectral density or autocorrelation function [27]. We also neglect T_{offset}, so equation A.6 becomes

$$C(t) = t + \Delta\nu t + x(t) \tag{A.8}$$

Table A.1 illustrates different oscillator types and some parameters that will be useful to our model of $C(t)$.

Table A.1: Summary of Oscillator types [28].

Oscillator type	Quartz (TCX)	Quartz (MCXO)	Quartz (OCXO)	Rubidium	Cesium	Hydrogen maser
Aging/ year (D)	5×10^{-7}	3×10^{-8}	5×10^{-9}	2×10^{-10}	None	$\cong 1 \times 10^{-13}$
Frequency offset ($\Delta\nu$)	1×10^{-6}	1×10^{-8}	$1 \times 10^{-8} - 1 \times 10^{-10}$	$5 \times 10^{-10} - 5 \times 10^{-12}$	$5 \times 10^{-12} - 1 \times 10^{-14}$	$1 \times 10^{-12} - 1 \times 10^{-13}$
Stability ($\sigma_x(\tau)$, $\tau = 1\,s$)	1×10^{-9}	3×10^{-10}	1×10^{-12}	$5 \times 10^{-11} - 5 \times 10^{-12}$	$5 \times 10^{-11} - 5 \times 10^{-12}$	$\cong 1 \times 10^{-12}$
Cost	$100	$1000	$2000	$3000-$8000	$30,000-$80,000	$200,000-$300,000

In table A.1, $\sigma_x(\tau)$ is the Allan deviation defined as

$$\sigma_x(\tau) = \sqrt{\frac{1}{2(M-1)} \sum_{i=1}^{M-1} (\overline{x_{i+1}} - \overline{x_i})^2} \tag{A.9}$$

where M is the number of values in the x_i series of random process $x(t)$, and the data is equally spaced in segments of τ seconds. The Allan deviation is a standard specification for stability recommended by the IEEE. However if we assume that x_i values are stationary and time independent, for instance if $x(t)$ is normally distributed, then, the Allan deviation becomes the classical statistics deviation.

Therefore, in this thesis we will assume that $x(t)$ is a normally distributed random process with zero mean and deviation σ_x.