Czech Technical University in Prague
Faculty of Electrical Engineering

Diploma thesis

Distributed topology control in MANETs

Tomáš Meiser

Thesis supervisor: Ing. Milan Rollo, Ph.D.

Study programme: Open Informatics

Specialization: Artificial Intelligence

July 2012

Prohlášení autora práce

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 28. 7. 2012

Podpis autora práce

Acknowledgements

I would like to thank all the people who helped me in any way with this master thesis. Especially, I would like to thank to my supervisor Ing. Milan Rollo Ph.D. for his time and patience.

Abstrakt

Cílem této práce je prozkoumat možnosti použití distribuovaných algoritmů pro řízení topologie v bezdrátových mobilních sítích s ohledem na implementaci takového systému na reálné hardwarové platformě. Navržený systém řízení topologie má být implementován jako modul pro projekty využívající multi-agentní platformu Aglobe; dále jako podpora pro provádění simulací s částečným nasazením na reálném hardware (tzv. mixovaná simulace). Součástí práce by mělo být nasazení takto implementovaného systému řízení topologií na reálnou komunikační platformu Xbee spolu s nasazením uvnitř komunikačního protokolu, kde by měla sloužit k zlepšení vlastností přenosového média. V neposlední řadě je cílem této práce otestovat finální řešení na sadě experimentů, které odpovídají budoucímu využití pro potřeby výzkumu bezpilotních prostředků.

Abstract

Aim of this work is to investigate available distributed topology control algorithms in mobile ad hoc networks (MANETs), with respect to their deployment on real hardware platform. Designed system for topology control should be implemented as plug-in module for projects, which utilize multi-agent simulation software Aglobe and in the same time as a part of mixed simulations. In these simulations, where the part of system is deployed on real hardware, topology control module will be deployed on Xbee communication platform as part of communication stack. Topology control methods implemented in this work should improve the properties of communication platform, which is used in several ongoing research projects at our department. All the results and system parts will be tested in mixed simulation to prove their correctness and suitability for future use in domains with unmanned aerial vehicles.

Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Cybernetics

DIPLOMA THESIS ASSIGNMENT

Student: Bc. Tomáš Meiser

Study programme: Open Informatics

Specialisation: Artificial Intelligence

Title of Diploma Thesis: Distributed Topology Control in MANETs

Guidelines:

1. Familiarize yourself with problem of topology control in mobile ad hoc networks.
2. Formalize problem of finding optimal transmission power for heterogeneous agents to provide defined k-connectivity in wireless network.
3. Design a topology control algorithm which will utilize autonomous agent for connectivity improvement (UAVs, drones in scenarios with other self-interested actors.
4. Implement designed algorithm using an A-globe simulation system.
5. Experimentally evaluate properties of this algorithm using real hardware devices (unattended ground sensors, UAVs) and humans.

Bibliography/Sources: Will be provided by the supervisor.

Diploma Thesis Supervisor: Ing. Milan Rollo, Ph.D.

Valid until: the end of the summer semester of academic year 2012/2013

prof. Ing. Vladimír Mařík, DrSc.
Head of Department

prof. Ing. Pavel Ripka, CSc.
Dean

Prague, January 10, 2012

České vysoké učení technické v Praze
Fakulta elektrotechnická

Katedra kybernetiky

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: Bc. Tomáš Meiser

Studijní program: Otevřená informatika (magisterský)

Obor: Umělá inteligence

Název tématu: Řízení topologie mobilních bezdrátových sítí

Pokyny pro vypracování:

1. Seznamte se s problematikou řízení topologie v mobilních bezdrátových sítích.
2. Formalizujte problém nalezení optimálního vysílacího výkonu pro heterogenní agenty k dosažení definované k-konektivity sítě.
3. Navrhněte algoritmus řízení topologie, který bude využívat autonomních agentů pro zlepšení konektivity sítě (bezpilotní prostředky, UAV), ve scénářích s dalšími nezávislými jednotkami.
4. Implementujte navržený algoritmus za použití agentní platformy A-Globe.
5. Experimentálně ověřte vlastnosti algoritmu pomocí hardwarových zařízení (pozemní senzory, UAV) a lidí.

Seznam odborné literatury: Dodá vedoucí práce.

Vedoucí diplomové práce: Ing. Milan Rollo, Ph.D.

Platnost zadání: do konce letního semestru 2012/2013

L.S.

prof. Ing. Vladimír Mařík, DrSc.
vedoucí katedry

prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 10. 1. 2012

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Wireless Networks

Wireless networking in general is a way to transfer data using signals encoded in electromagnetic waves. This communication medium is an integral part of modern portable digital devices and allows information exchange and source sharing across the network. Its importance grows with increasing numbers of devices like laptops, PDAs, tablets and cell phones equipped with Wi-Fi and Bluetooth technologies.

Wireless data transfer is affected by several phenomenons, which do not appear in wired communication or their impact on this communication is very small. The most important property of wireless communication channel is the fact, that the channel is shared among all devices in the same spatial area, which communicate on the same frequency. Utilization of the shared channel causes the possibility of interference in communication. Communicating device can even interfere with itself, because of multipath signal propagation and its reflection from the obstacles. Devices also interfere with each other, because of broadcast nature of the wireless communication. Finally the electromagnetic waves can interfere with other other electrical devices that behave as sources of electromagnetic radiation. Besides interference weather conditions also have strong impact on the quality of the wireless media. Higher probability of data loss caused by the interfered data corruption together with the low transmission rates, which is closely coupled with the utilized frequency and encoding of data in the signal, cause low bandwidth of the wireless media in comparison to the wired ones.

Wireless network structure and channel properties may change rapidly in time. Communicating devices may move and their mutual positions, distances and surrounding environment (obstacles, terrain) strongly influence the possibility and quality of communication. From above mentioned follows, that the wireless communication channel has much more dynamic nature than the wired one. Portable devices, which mostly utilize this way of communication also have limited computational and energy resources, as they usually run on batteries and have small dimensions. Application development for mobile computers is discussed e.g. in [1].

Widely spread wireless technologies like cellular networks and wireless LANs (which are based on the combination of wired and wireless networks) are considered as well known area of knowledge nowdays. Research in area of wireless networks with backbone infrastructure is still considerable, but in recent years research community started to focus on more challenging area of these networks, where fixed infrastructure is no longer required to dynamically form a network of units equipped with radio transceivers.

Wireless networks where nodes communicate directly with each other without any fixed infrastructure are called Ad hoc networks. Nodes equipped with radio transceivers can communicate directly with other nodes in their neighborhood and also with rest of nodes along multihop paths where other nodes serve as relay points (in case that network is connected). This kind of communication network is very useful in situations, where fixed infrastructure is not available. We can imagine a scenario, where a group of exploration robots is deployed in outlying part of the world or in some unknown region and robots have to form a formation to survey this area automatically. Other possible application of communication platform based on Ad hoc networks is to coordinate rescue teams in the region damaged by the natural disaster. In such region the whole communication infrastructure could be damaged. The standard walkie-talkie may not provide sufficient communication platform, because of limited range of individual transceivers. A particular type of Ad hoc network is a Wireless Sensor Network. Small smart sensors, sensing some environmental properties equipped with limited power source and short-range transceiver, are deployed in area and should deliver measured values to some gateway.

1.2 Motivation

Main motivation for this work is the latest research carried out at Agent Technology Center at Department of Computer Science and Engineering, CTU in Prague, in the area of unmanned aerial vehicles (UAVs) and various scenarios which utilize use of these UAVs.

Recently raised a demand on performing field tests of currently implemented algorithms for UAV control, which are nowadays tested only in simulated scenarios built on top of Aglobe multiagent platform. In projects like Agentfly [17], Tactical Agentfly [19] or Agent Scout [18] appears necessity to prove the results achieved in simulation also in real-world applications. For this purpose two UAV units from Procerus Technology company were acquired recently. These UAVs are capable of controlled flight following a list of GPS waypoints given by a ground station. This level of control is not sufficient to provide complex mission control, therefore second tier of control is employed on-board. For this purpose we installed Gumstix COM in each UAV, which hosts high level control algorithms derived from above mentioned projects. Waypoints necessary for mission accomplishment are computed by this second tier control and passed to the autopilot unit through a serial line. Crucial part of mission planning and execution is also the communication channel to other cooperating UAVs.

Because preliminary experiments shown that the communication platform dedicated for autopilot is not able to provide additional bandwidth required for complex planning and negotiation between UAVs, additional Xbee modules have been deployed onboard and connected to Gumstix COMs. This low level communication platform is able to provide unguaranteed datagram layer, which is well-informed about state of communication medium. To establish guaranteed and controlled communication channel in pure distributed fashion is necessary to provide upper control layers. Topology control (TC) is part of routing and medium access layers, which could strongly enhance the communication capability and throughput of such platform and these properties are really demanded in application scenarios we focus on.

There are two main reasons for careful design of network topology: (i) to save energy, which is utilized for maintaining radio transmission and (ii) to increase the capacity of the wireless medium by spatial reuse of the channel. Because the shared medium is not able to provide concurrent transmission for two units with overlapping regions, topology control mechanism should make these regions as small as possible. Then the set of neighboring units, which experience corruption of their transmission should decrease.

For scenarios which we consider in our research at ATG the first pointed motivation is irrelevant, because for UAVs, which are propelled with electro motors that draw up to 80A, the power consumption of the radio unit is insignificant. But the second mentioned reason for topology control shows to be crucial. The bandwidth of radio modules used on UAVs is low anyway (considering our needs for information exchange) and data corruption caused by interference could turn it to inapplicable platform. Furthermore in many application scenarios (eg. collision avoidance) the most of communication is performed just among the group of closest unit, which negotiate solution of some task. In such a case it is senseless to pollute channel for distant units as well.

In this work we will focus on two specific scenarios:

- Group of autonomous UAVs that are performing a complex mission. We assume that UAVs have to communicate to exchange information required for high level planning, task decomposition, collision avoidance, etc. Structure of network changes rapidly due to the fast movement of these units and their mutual arrangement.

- Other motivation scenario is such, where the group of high dense smart sensors is deployed on ground. Nodes have to communicate with each other, optimally with low transmission power (because of their high density). But sudden presence of the UAV, which has to harvest collected data, should trigger sensors to temporarily increase their transmission power to provide reliable channel for such harvesting. In this case we assume that most of the time the network topology would be static in comparison to the previous case.

From the above mentioned arises main motivation of this work, which is to add a capability of topology control and channel management to communication platform for experimental UAVs based on Xbee modules .

1.3 Contribution

In this work implementation and analysis of topology control algorithms for mobile ad hoc networks with emphasis on deployment on real UAVs is presented. These algorithms were implemented and evaluated using a Tactical Agentfly software platform, which is suitable for pure simulated scenarios as well as mixed scenarios which integrate real hardware components besides the simulated ones. The implemented module was tested on UAVs and ground stations equipped with Xbee communication devices. To solve this problem following steps have to be carry out:

- Analysis of characteristics of available hardware devices and demanded scenarios. Identification of the usable configuration methods, built-in functions and limitations of hardware communication platform, which can influence the opportunity of the topology control algorithm to solve the problem of range assignment
- Identification of the best suitable approach to topology control problem in this particular arrangement of hardware components and scenarios.
- Implementation of the suitable and reliable communication protocol, which will be able to broadcast, unicast and acknowledge messages of arbitrary size and format with built-in routing protocol for multihop communication and media access (MAC) layer. MAC layer should be able to utilize some medium control mechanism to avoid collisions on channel.
- Implementation of the topology control module in the Tactical Agentfly platform. This module should be able to operate on the real HW platforms, but also with a simulated entities in the same manner. For this interoperability proper interface will be designed, which will simulate the original responses of the HW modems (Received Signal Strength Indicator (RSSI) value from distance etc.).
- Finally carry out a set of experiments, which should prove, that the solution is applicable for the given scenarios. Part of the experiments will be performed as a pure simulation and the rest as mixed simulation with real HW ground stations (WSN) and UAVs.

1.4 Thesis Organization

This thesis is organized as follows. In chapter 2 we describe the taxonomy of TC approaches and its pros and cons. Also the formalization of topology control problem is introduced in this chapter. In chapter 3 we discuss parameters and configuration possibilities of Xbee communication modules with emphasis on transmission characteristics and useful information, which could be gathered from this platform for TC decision making. Next we discuss all relevant parameters of used HW platforms and its impact on selection of suitable TC method. In chapter 4 we give a brief overview of the Tactical Agentfly platform and relevant application scenarios and its characteristics. Next the representative TC algorithms from literature are introduced with arguments for their employment in mentioned scenarios. Description of the implemented TC module with all the communication stack layers (message transport, routing, MAC) are presented in chapter 5. Way how simulated entities compute and imitate signals of Xbee platform for the TC module and how the visualization and evaluation of experiments was performed is described in this chapter as well. Arrangement of each experiment together with its results are clearly discussed in chapter 6. Finally, chapter 7 concludes this thesis.

2 Topology control

In this chapter we will introduce formal definition of the topology control problem as proposed by Santi in [14] and provide its brief taxonomy. From the point of view of the agent systems topology control problem is defined as a distributed constraints satisfaction problem (DCSP). Each agent (wireless device participating in ad hoc network), should solve the decision problem of setting its transmit power level in order to satisfy some networkwide property [7]. At the same time agent should try to minimize its own power consumption and therefore the interference caused by the transmission in its neighborhood .

2.1 TC problem formulation

From above mentioned definition follows, that the optimal solution is such, where each agent has its transmit power level set as low as possible to keep the specified networkwide connectivity property satisfied.

More formally, assume there is a set N of nodes with $|N| = n$ deployed in certain bounded region R. In our case we could assume that region is three dimensional cube $R = [0, 1]^3$. Than we can say that position of any node $u \in N$ is denoted as *L(u)*. For our case of mobile nodes, should be definition of position extended by time factor t. Wider definition of position function is $L : N \times T \to R$ and assigns to every node from N and every time $t \in T$ the coordinates of n at time t in region R. Here T is any time from network lifetime. Now it is possible to define mobile ad-hoc network as $M = (N, L)$. With such a defined network we could also define the distance d, where *d(u,v)* for $v \in N$ and $u \in N$ is the standard euclidean distance. Range assignment is for network *M(N,L)* a function *RA(u)*, which for every node u from N gives the value $RA(u) = (0, r_{max}]$. This value represents the transmitting range of node u. Specific value of r_{max} denotes the maximal transmitting range of node and we assume that for environment with homogeneous devices is this value same for each node $u \in N$. Now we could introduce communication graph, which is formed as result of topology control algorithm. It composes of nodes and edges and is defined for given RA, M and $t \in T$ as $G_t(N, E_t)$, where directed edge $(u, v) \in E_t$, with $u, v \in N$, only if $RA(u) \geq d_t(u, v)$. If the opposite directed edge exists (also $RA(v) \geq d_t(u, v)$), than such an edge can be denoted as *bidirectional* or *symmetric.*

In the introduction to this chapter topology control problem was mentioned as a making decision about transmitting power level. This consideration is based on practical way how to set the transmitting range of wireless device and therefore control the topology of underlying communication graph. Now it is time to clarify the relation between transmitting power and transmitting range. First we should explain what does it mean that some node is in transmitting range of some other node in terms of wireless channel. When node u broadcasts the radio signal with power P_t, than other node v receives this signal if, and only if, $P_r \geq \tau$, where P_r is received signal power and τ stands for some sensitivity threshold, which depends on properties of communication platform (utilized devices and channel). It is clear that P_r depends on P_t, but it also depends on path loss, which stands for signal degradation on the way from transmitter to receiver. This path loss may be modeled in many ways, but the log-distance path model is mostly mentioned in literature:

$$P_r(d) = \frac{P_t}{d^\alpha} \tag{2.1}$$

where d is the distance between transmitting and receiving node,
α path loss exponent ($\alpha = 2$ for free space).

Using this path loss model it is possible to show that for every possible transmitting power level P_t^l exists exactly one transmitting range RA^l.

$$RA^l = \sqrt[\alpha]{P_t^l} \tag{2.2}$$

From this equation follows that for defined path loss model are the range assignment and transmit power level assignment problems interchangeable. For more detailed description of wireless channel modeling and radio signal propagation model see e.g. [13].

Communication graph is *k-connected* [11] if and only if it is connected and there does not exist a set of *k-1* nodes whose removal disconnects the graph. For *k=1* simple connectivity of network guaranteed. For *k¿1* the network topology is resistant to *k-1* nodes failure, which means that network stay connected in the case when *k-1* nodes fail to communicate. If the topology control holds this criterion, then if the graph is *k-connected*, when nodes utilize full transmitting power, then it stays *k-connected* after each node sets its own transmitting power level according to result of topology control algorithm.

2.2 Taxonomy

The main level of TC taxonomy is the distribution of TC algorithm. There exist many well-known graph algorithms, which solve the range assignment (RA) problem optimally [9]. Many of these algorithms were proposed in literature in the past, because of clear proof of correctness. The main disadvantage of centralized TC approach is its low robustness and communication overhead. Central unit needs to create a view of the whole network to provide solution of the TC problem and the resulted assignment of transmitting power levels should then be distributed back to individual devices. This approach is not immune to central unit failure and process of acquisition of information is ineffective in dynamic scenarios. But the centralized TC problem solution is good choice for benchmarking of developed distributed algorithms.

Distributed algorithms for TC problem are more suitable approach and will be discussed in the following part of this thesis. Distributed solution is robust against nodes failure and decreases communication overhead, because only the local view of network is required by each node. It is clear that the local view of network enables the node to find only the locally optimal solution, but the well-designed TC algorithm ensures, that local optimality in each part of network ensures at least *close to optimal* solution networkwide.

Finally, numerous algorithms were designed, which combine both previously presented approaches. These methods are mainly cluster based (e.g. [5, 8]) and are suitable for high dense scenarios, where devices naturally form groups. In this case some clustering algorithm is executed and *clusterhead* of each cluster is elected. Then TC is performed in centralized manner by the clusterhead in each cluster and intercluster backbone is established by clusterheads in distributed manner. These approaches are closely tied to selected clustering method. Analysis of impact of clustering algorithm on the performance of TC algorithms is given in [21].

2.3 Homogeneity

TC methods could also be divided into two groups according to the homogeneity of decision making process output. First is the approach based on homogenous range assignment, where all the network nodes have to use the same transmitting range. This case is known as Critical Transmission Range (CTR) problem, which reduces the original TC problem to simpler problem of determining the minimum value of transmitting range such that the networkwide property is satisfied. Finding of CTR is the approach reminding centralized solution of TC problem and therefore it has the same pros and cons. It is simple to prove that the networkwide goal is met, but the decision is based on view of the whole network, whose creation brings high communication overhead. If the information about physical node placement is known in advance than the CTR is length of the longest edge of the Euclidean minimum spanning tree (EMST) [6].

In nonhomogeneous topology control, nodes are allowed to choose different transmitting ranges. Range assignment decision of each node is in this case based on local information about its own neighborhood. Therefore the decision should be at most almost (locally) optimal, which implies that it is harder to prove the satisfaction of networkwide property.

2.4 Node information model

Nonhomogeneous TC could be classified into three main categories arising from the type of information that is used to provide TC problem solution. These categories are presented in ordering according to the accuracy of information about node's neighborhood.

2.4.1 Location based

This approach is based on very accurate information about node positions. It is the same type of information utilized by centralized approach and therefore many of algorithms are only the distributed versions of these approaches, which somehow try to approximate centralized solution in distributed manner. As written before this type of protocols relies on very accurate information, which is assumed to be somehow available to each node. The way to provide this information is to equip every node in network with GPS receiver. This receiver has to provide accurate position estimation with low power demands, but this type of HW equipment could be expensive and implies some extra demands on network nodes, which don't need this equipment for to provide their main functionality. There exist many location estimation techniques introduced in literature. These techniques assume that at least few of network devices (uniformly distributed in the network area) are equipped with GPS receivers (called anchor nodes) and other nodes can estimate their position by exchanging messages with the surrounding anchor nodes. These estimation techniques reduce the cost of HW equipment, but put extra message overhead in place. It has to be pointed that location based techniques are based on position exchange in the neighborhood of each node. This information could be a part of some beacon message, which is broadcasted in certain period of time, but the message overhead increases with dynamism of nodes mobility. To provide the local information about neighboring nodes up to date, each node should send the beacon message with its new position every time, when the position is changed for some E, which is the minimal distance that can affect topology. In higly dynamic scenarios this message overhead may become unacceptable.

2.4.2 Direction based

When precise position is not available to nodes another group of algorithms could be used which relies on the ability of nodes to estimate the relative direction of their neighbors. This information is less accurate than the previously introduced location based information. This problem is known in signal propagation domain as Angle-of-Arrival (AoA) problem. It is typically solved by equipping nodes with more than one directional antenna. So the extra HW on the nodes is needed. When we utilize some distance estimation techniques in addition (for example RSSI based approximation) we can gather some relative distance information, but forming of global position information in this case is expensive in sense of message overhead. The main advantage of this approach in comparison with location based techniques is that the AoA can be accurately estimated in indoor environment (where GPS signal is not available) as well and direction-based TC protocols can produce almost as good solution as in the case of location-based TC.

2.4.3 Neighbor based

Finally we will discuss TC protocols that rely on node's ability to discover neighboring nodes within the maximum transmitting range. For this class of methods is crucial the possibility to determine the number and identity of nodes in full power range and to sort these neighboring nodes in some way. The intuitive principle of nodes sorting method is quality of link to each neighbor. This quality could be expressed in many possible fashions. If the Receive Signal Strength Indicator (RSSI) is available for each incoming message, then its mean or some kind of average value is good choice to estimate quality of link in time. RSSI could also stay as good approximation of relative distance between two nodes as was noted above, but it is reasonable only in open-space scenarios. If the RSSI is not included in received message or HW communication platform is not able to provide this value in some other way, then the packet lose or packet corruption statistics are also possibilities to estimate the link quality. In this case, problem could arise with nodes, which do not transmit for a long period of time or the size of the transmitted messages differs from node to node (in case of message corruption statistics, where the probability of message corruption is closely tied to message size). From the above mentioned follows that in scenarios, where each node regularly communicates with other nodes, is not required any additional message overhead to provide reasonable estimation of sorting criterion value. In other scenarios suitable way to provide this value up to date is introduction of some beacon messages, which are transmitted periodically and contain just minimum of information, which is suitable to distinguish transmitting node. Repeat frequency for beacon message comes out of scenario dynamics.

It is possible to say that this is the critical amount of knowledge about the network required by the nodes to build the view of network topology. If nodes are not able to identify their neighbors, they don't have sufficient knowledge to provide reasonable network model for making decision about transmitting power level. From the fact that information used to solve the TC problem in this case is not directly derived from either the absolute or relative position of nodes and it is possible to maintain this model up to date with minimal demands on massage overhead, follows that the TC techniques based on it are probably the most suitable for application in mobile ad hoc networks.

Basic neighbor-based TC protocols are strongly connected to the k-neighbor graph definition. This graph is for some k, with 0¡k¡n-1, where n is number of network nodes, defined as directed (with possibly asymmetric links) subgraph, where each node is connected only with k closest neighbors. There also exist the symmetric modifications of such graph: symmetric supergraph, where each unidirectional edge is turned to bidirectional, and symmetric subgraph, where each unidirectional edge is dropped. Using this notation we are able to define the worst case connectivity for the network such that we look for smaller k for such the generated k- neighbor graph stays strongly connected. It is clear that for general case the k should be set to n-1 to ensure worst case connectivity. Determining the k automatically for instance is known as k-neighbor connectivity problem. This problem should be solved only with networkwide information about connectivity, which is in real-world application achievable only by networkwide message exchange and therefore practically unachievable. There were many empirical values for critical neighbor number (CNN) introduced in literature, estimated through an extensive simulations, which for given distribution of nodes ensure worst case connectivity with high probability. From previously written follows that worst case connectivity is ensured only if the k is unbounded. In other ways the worst case connectivity could be provided at most with high probability. However bounded k increases network capacity because each node is allowed to communicate only with its k 'closest' neighbors and therefore the physical node degree is also bounded. As the physical degree of each node in network decreases, the spatial reuse of wireless medium increases.

2.5 Topology control with mobility

Mobility is a significant property of nodes in ad hoc networks. As was noted in introduction many of participating nodes could be cell phones, PDAs and other devices where their movement is expected. But the mobility could be quite well covered by stationary network model, when mobility stays low. It is intuitive that it is possible to use TC protocol designed originally for stationary network and perform it periodically. However the topology, which is computed in time t, does not remain correct in some time $t{+}\epsilon$. When nodes change their positions and actual topology becomes obsolete is time to reexecute the TC protocol. It is clear that the lifetime of topology will decrease with increasing mobility in network.

When the node mobility is at least moderate (in comparison with size of its range), we have to consider some extra properties of TC protocol. It is clear that this protocol should be distributed for sure. It is meaningless to try to accumulate data about whole network, because till it is done the situation will change. From the same reason arises that only local information should be utilized. It is also reasonable to provide only the information, which is somehow resilient to nodes mobility. With increasing speed of nodes grows the demand on recomputation of network topology. Therefore the TC protocol designed for mobile instance of network should be fast enough to track the changes in the network. The fact that resulted topology will be probably outdated soon after its establishment is strongly connected with networkwide property, which we are able to ensure. It is pointless to provide topology which preserves connectivity for sure, because it could loose this property quickly by node mobility. Therefore algorithm should just try to ensure connectivity for most of nodes and for most of network life time.

Several widely used mobility models were introduced in the past, which allow both simulation on individual nodes movement and statistical analysis of behavior of whole mobility model. For detailed description see e.g. [3, 2].

3 Hardware platform

Communication plays crucial role in deployment of distributed algorithms on real hardware platforms. Simulated communication layer used during the development and testing phase is much more stable and provides better throughput, than one provided by real long range wireless communication hardware. The messages could be delayed or dropped, because of errors in communication channel. Main problem is related to high interference on shared wireless medium. Signal can interfere with itself because of multipath propagation, devices can interfere with each other because of broadcast nature of the transmission and radio signals can interfere with other electrical devices. Interference can cause higher data loss rates. Devices may move, therefore signal power attenuates with distance, and also signal quality is influenced by the weather and so on. Last but not least comes the problem of limited computing and energy resources of devices that run on battery. From these aspects follows the lower transmission bandwidth ability and robustness of wireless communication, which have to be solved.

3.1 XBee-Pro platform

Employed communication platform is based on XBee-Pro DigiMesh extended range RF modules (see Figure 3.1). These modules operates in 2.4 GHz ISM frequency bands and provide reliable long range delivery of data between remote devices with minimal power consumption and small dimensions of module. The size and weight of communication modules is very relevant in application on Unmanned Aerial Vehicles (UAVs) platform. Utilized RF modules provide communication range up to 1 mile (1.6 km) with 2.0dB dipole antenna. RF data rate 250 kbps is shared among all devices in communication range. RF modules are connected to the computation units (laptop or Gumstix computer-on-module) through UART RS-232 line. No hardware flow-control is used. The RF module modulation and medium access control is provided according IEEE 802.15.4 standard, which specifies use of DSSS (direct sequence spread spectrum) for data modulation and use of CSMA/CA protocol with random exponential back-off to access wireless medium.

3.1.1 Module configuration

Serial line is configured to communicate with RF module with baud rate set to 115 200 bps using 2 stop bits without parity. This configuration is used according to the measured quality of line and interference, which is caused mainly by electric engine on UAV platform. Two stop bits were introduced on he computer side of communication line to overcome the problem of timing. Capabilities of device to provide the right timing for

Figure 3.1: XBee-Pro Wireless RF Module

equipment like UART interface depends on internal clock frequency. The low frequency signal is reached by division of original clock signal with integer divisor. Therefore just the approximation of demanded frequency is reached. In this particular case, where Xbee unit is connected directly to computer the problem appears, because the reached approximation of clock signal differs a lot. Utilizing of two stop bits is the well known technique, which is recommended by manufacturer to solve this problem. RF modules could be configured as a transparent communication channel, which means that any data incoming to the module through serial line would be transmitted without a change and all received RF data are passed to serial line immediately in the same manner. Alternatively modules could be configured in so-called API mode, which is more complex and allows better supervision. MeshGrid layer is disabled on RF modules to ensure that any extra traffic (network exploration, routing or multihop relay) wouldn't be passed through the RF medium. In scenarios, where the physical position of devices could change rapidly, the traffic caused by maintenance of network topology increases. Also the flood broadcast schema for multihop network could overcrowd the network with useless messages. Instead of MeshGrid layer simple ad-hoc topology is introduced. Each message is broadcasted to each node in communication range in point-to-multipoint schema manner. Addressing and acknowledgement sending is provided by transmission control protocol in upper layer. All acknowledgements, consistency checks and retransmission in RF module MAC layer are disabled and substituted with upper layer functionality.

3.1.2 Data handling modes

Xbee modules provide two kinds of data handling modes. First one called transparent mode provides simple data processing and low ability to control module configuration.

Any data which come through serial line are sent through RF interface. Module is provided by 200B income buffer and after buffer overflow the next data are dropped. Data transferring through RF interface start, when there is at least 100B in income buffer, which is the maximal size of data part in RF packets, or when the packetization timer times out. Default packetization timeout is set to 3ms. Configuration of module parameters is done by standard AT commands. Frequent change of parameters especially changing the destination address for unicast communication is very inefficient, because of AT mode guard time, which is time before, after and between each AT command character.

Second possible data handling mode called API mode is using complicated packet structure for communication with Xbee module, but provides much more possibilities to gather extended data from module and allows quick reconfiguration of relevant module parameters. Additional ability of this mode is a possibility of remote reconfiguration of other modules in communication range. Data transfer packets are by its structure able to set the receiver address for each transmitted RF segment, which enables to use the standard 802.15.4 acknowledge mechanism for unicast communication. Each transmitted API packet is acknowledged by module to sender by its ID. This response provides additional status information about data transferring, like failure due to back-off mechanism, acknowledged or unacknowledged successful transmission. Received RF data are packed also in specially structured packet and provide information about sender address and RSSI (Received Signal Strength).

3.1.3 Java serial interfacing

Java applications are not able to perform communication through serial line by default. In our experiments we have tested two different possibilities. First one is communication based on RXTX open source library, which is recommended. This library meets the platform independence, which is crucial for many of Java applications, but experiments show, that it is very ineffective and spends huge part of bounded computational performance of our embedded Gumstix computation unit.

Second possibility for interfacing the serial line from Java applications is our lightweight utility, which uses standard Unix POSIX interface to communicate with serial port and provides data tunneling to the TCP socket pipe. This TCP socket is then interfaced standard way from Java application. Experiments show, that this solution provides better performance on embedded platform with bounded performance, but the utility isn't portable for other than UNIX platforms.

3.1.4 Low-level communication experiments

First set of experiments has been focused on one way point to point communication using the transparent data handling mode, where the RF packets have been sent in multicast manner and addressing performed by upper application layer. This configuration has proven, that there is a possibility to provide reliable communication channel with speed up to 7KBps. Communication between ground station and UAV has been affected by distance, and also by orientation of UAV against the position of ground station. This behavior was identified as interference with UAV's electric engine, when it is situated on the line of sight between UAV dipole antenna and ground station antenna. Worst case packet loss have been measured in situation, when the UAV was approximately in 1km distance from ground station and dipole antenna was in alignment behind the UAV's engine. Experiments show, that this worst case packet loss won't exceed the 20% barrier.

Second set of experiments has been focused on duplex communication with the same configuration as the previous. One communicating side has been sending packets of fixed size with fixed sending period and the second side has been responding with the same data as received one. These experiments show, that the communication is strongly affected by collisions on the wireless medium. We tried to overcome this problem by finding best combination of packet size and sending period. Second possibility was the configuration of back-off exponent in MAC layer of the module. Experiments show, that there is no universal configuration which helps to overcome these problems.

Results of communication tests based on transparent data handling mode, which enable to reuse maximum of currently implemented communication layers, lead to development of the complicated API data handling mode layer, which provides much more important information about the wireless channel and Xbee module status to the upper layer. This information will be used to avoid problem with collision in fully duplex communication, which is crucial in each scenario, where the agents have to negotiate.

3.2 Computational unit

For developed hardware platform Gumstix Overo Fire COM (Computer On Module) unit was utilized (see Figure 3.2). It is based on ARM Cortex-A8 OMAP3530 SOC (Open Multimedia Application System on Chip) core developed by Texas Instruments corporation. This is the most suitable unit, whose computational core with frequency 720 MHz is able to perform up to 1200 Dhrystones MIPS. To other available equipment of the platform belong graphic accelerator for 2D and 3D graphic with high-performance DSP (Digital Signal Processor) core. For communication WiFi 802.11 b/g and Bluetooth interface are available. As extension of incorporated 256 MB ROM and RAM memory is

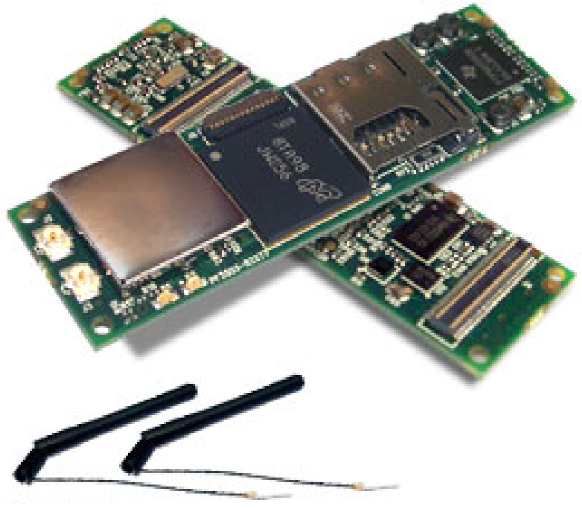


Figure 3.2: Gumstix Overo Fire COM

included interface to connect extended microSD card. For connection of external devices is this module equipped with I2C bus interface, high-speed USB hub and 3 UART RS-232 interfaces, where one is utilized as debug console of operating system, next as interface to autopilot unit and last for interfacing Xbee module. To most significant properties of this module, according to its deployment in UAVs, are low dimensions (17mm x 58mm x 4.2mm) and low power consumption about 5W.

As operation system of this module the Angström linux distribution is utilized, which is developed to provide easy way to operate each incorporated equipment of platform. For better performance the system installed in the built-in NAND flash memory. It is also possible to run OS from extended memory card, but the response is worse in this case. The customized distribution is enhanced with Java language interpreter, RXTX library and some other auxiliary software utilities, which help for better remote configuration of modules.

3.3 UAV Procerus platform

Procerus UAV is the platform for testing and development of unmanned aerial vehicles, which is developed by Procerus Technologies company (see Figure 3.4). It is robust solution based on Kestrel autopilot unit from the same manufacturer, which together with ground station form the core of the system.

The plane itself belongs to category of so-called delta-wings. This construction with the arrow shape with large surface of the wing without the tail pads ensures high stability and performance. The manoeuvres are performed through pair of ailerons, which are situated on trailing edge of each wing part. To perform roll change ailerons are triggered in the

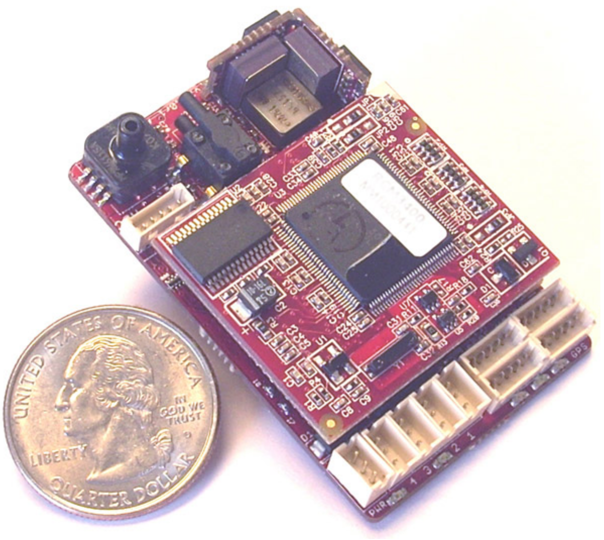


Figure 3.3: Kestrel Autopilot

mutual direction and in opposite direction for pitch control. This type of construction (without tail pads) contributes to minimization of damage chance when something fails, but it is not able to provide the yaw manoeuvre. Yaw maneuver could be performed using a combination of pitch and roll change. With brushless electro motor with maximal draw about 80A the plane is able to reach maximal airspeed about 35 m/s, but normal cruise speed is 20 m/s and minimal for safe manoeuvring is about 12 m/s.

Kestrel autopilot (depicted in Figure 3.3) is the standalone control unit which records and processes sensor data and controls the fly and navigation of UAV. Its IMU is composed of 3-axis rate gyros and accelerometers. Absolute and differential pressure sensors provides barometric pressure and aircraft air speed. Three temperature sensors combined with a 20 point temperature compensation algorithm reduce sensor drift and improve aircraft state measurement and estimation. It is powered by an 8-bit 29 MHz processor. The autopilot interfaces directly to the digital communication link which enables it to send real-time status telemetry to the ground station and receive commands in-flight. The GPS plugs into the autopilot board and provides inertial navigation information to the autopilot. The autopilot controls the aircraft through four standard RC hobby servos. Autopilot unit provides fly control for given flight plan using a set of PID regulation loops, which are configured for specific aircraft indirectly by tuning of flight control variables.

Figure 3.4: Procerus UAV

4 Tactical Agentfly platform

In this chapter we will present some key features of the Tactical Agentfly project, which are fundamental for this work. This project is hosted on A-globe multi-agent platform [16], which is a Java-based system for experimental scenarios testing with high scalability, simulation support, agent mobility and very fast exchange of messages among hosted agents. With relatively lightweight architecture it strongly outperforms other popular multi-agent platforms. The main objective of Tactical Agentfly project is to develop tactical missions control system for unmanned aerial vehicles. UAVs in the system are able to accomplish complex tactical missions as a team. This means, that high-level mission description received from human operator should be automatically distributed among all available UAVs and then fulfilled. Between mission tasks there could be surveillance of unknown areas, patrolling, tracking of multiple targets or similar nontrivial tasks. Relatively small number of UAVs is forced to cooperate in same area which brings high requirements on precise coordination to avoid collisions. From this objective arise demands on negotiating, planning, collision avoidance and therefore also on a reliable communication.

4.1 Architecture

System is designed as modular multi-agent system and is able to provide so-called mixed simulations [10], where part of entities is completely simulated and some other are represented by real hardware units (UAVs, sensors or even humans). Each UAV is represented as an agent entity with set of sensors and actuators. Sensors are modules, which provides agent entity with information about itself and about environment in its surrounding. Modular structure of sensor model, where each module provides specific information through strictly defined interface, is used for simple implementation of the same sensor for simulated and real UAV. In the first case the central simulation coordinating platform is able to provide implementation of simulated sensor. In case of real UAV specific sensor interface is implemented as a view of real state obtained directly from autopilot telemetry. Actuators are modules that represent physical behavior of the UAV. In a same manner as sensors framework is able to implement actuator of simulated UAV as its response back to the simulation. For real UAV the same actuator module uploads new commands to the autopilot. Behind the actuators stands set of planners, which is encapsulated in so/called *aviator module*. Each planner is responsible for specific part of mission execution. On top of planners hierarchy stands the most important one responsible for collision avoidance. Other planners are specific for particular mission tasks.

4.2 Mixed simulation

Main advantage of this approach is its capability to provide robust and reliable mixed simulation. The term 'mixed' means, that simulated as well as real units are able to coexist in the system and the difference between them is recognizable only for system supervisor. This is achieved by splitting of agent, that represents real UAV. The main part of its control agent runs on-board, but small part called *ground-plane*, which provides its counterpart with information from simulation runs on the ground host. This architecture allows to quickly integrate new real UAV into the system and increase fidelity of simulation of complicated missions.

4.3 Communication

Communication in Tactical Agentfly system is performed via standard A-globe message transport mechanism, which is encapsulated into so-called channels. Channel provides message exchange among all entities, which are registered in it. This message transport system is unaffected by existence of real UAVs in system. Above mentioned principle of 'ground-plane' reflects this situation in each aspect. The message, which is transmitted by simulated entity and belongs to real UAV, is delivered to the ground twin of real UAV. If it is possible this message is processed by ground representation of plane or transmitted by dedicated Xbee channel up to the UAV. Other situation arises, when the real UAV decides to transmit message to another UAV. If the target UAV is simulated, than the solution is clear, because message should be delivered down into the simulation platform, where it is correctly processed. In case that target of message is real UAV two different solutions are available. First, that the message is delivered directly in plane-to-plane manner. The second, which is useful e.g. in case that the message should be visualized, is that message will be relied through ground station. In each case, when the message is delivered through the simulation platform, this platform is responsible that no visibility constraint will not be violated (message is delivered only in the case, when units are within their transmitting ranges).

4.4 Scenarios

Standard scenarios in Tactical Agentfly handle situation, where group of UAVs maintain mission in some bounded region. This mission composes of surveillance for given rectangular region or tracking of the ground target. One mission assignment may contain one or more surveillance regions and one or more objects for tracking. Mission could be assigned to one or many UAVs and more than one group of UAVs could be formed with independent mission commands. When the group of UAVs is forced to accomplish some mission, participating units divide the mission goals cooperatively and start to fulfil their private goals. It is necessary to point that one unit can perform more than one mission command (when number of commands is higher than the number of UAVs) and several units can share one mission command (e.g. surveillance area is than split into a subregions). At the same time UAVs should try to avoid collisions with other units. This implies that the mobility model of such scenario is specific for given mission assignment. For benchmarking of proposed TC module two special scenarios were introduced with different goal of communication and different mobility pattern. These scenarios contain stationary Ground Station unit, which is not commonly utilized in Tactical Agentfly. This unit can represent ground sensor or data collecting gateway. Introduction of this static unit allows to form scenarios with specific ratio between mobile and stationary units and therefore control the average mobility. It is crucial to determine behavior of proposed TC protocols in extreme scenarios from mobility ratio point of view, because finally utilized one has to keep the best response for each of them in average.

First introduced scenario composes of huge set of Ground Stations, which are uniformly at random distributed in the inner centralized square subregion of deployment area. Small group of UAVs executes surveillance task for whole deployment region. Goal of this mission is to maintain connectivity in the stationary network and stable links to UAVs at the same time. It is assumed that stationary WSN collects some mission relevant data and that data should be acquired by UAVs. This represents low mobility scenario. Only few units change their positions and because mission contains only one surveillance task, units move only inside one predefined region and along same regular paths.

Second proposed scenario considers only one Ground Station, which is located somewhere in deployment region, and huge group of UAVs, which execute complex mission with both surveillance and tracking tasks. The Ground Station represents mission operation center or gateway for gathering data collected by surveillance and tracking. In both cases of explanation it is crucial for the network operability to maintain connection to the Ground Station for each UAV (if it is possible) or at least for major part. It is clear that this scenario represents opposite mobility pattern than the first one. Almost each unit moves according its mission task and especially for tracking the path random could be random (depending on mobility model of tracked target).

5 Implementation

In this chapter we will present design of implemented modules for topology control. First we will introduce implementation of chosen TC protocols and its integration into the Tactical Agentfly framework. Next we will discuss TC actuator and TC sensor modules, which provide integration platform for TC protocol and are the main elements providing homogeneous mixed simulation. At the end of this chapter the Central unit module, which is responsible for maintenance of mixed simulation will be presented. The whole structure of implemented module is clearly shown in figure 5.1.

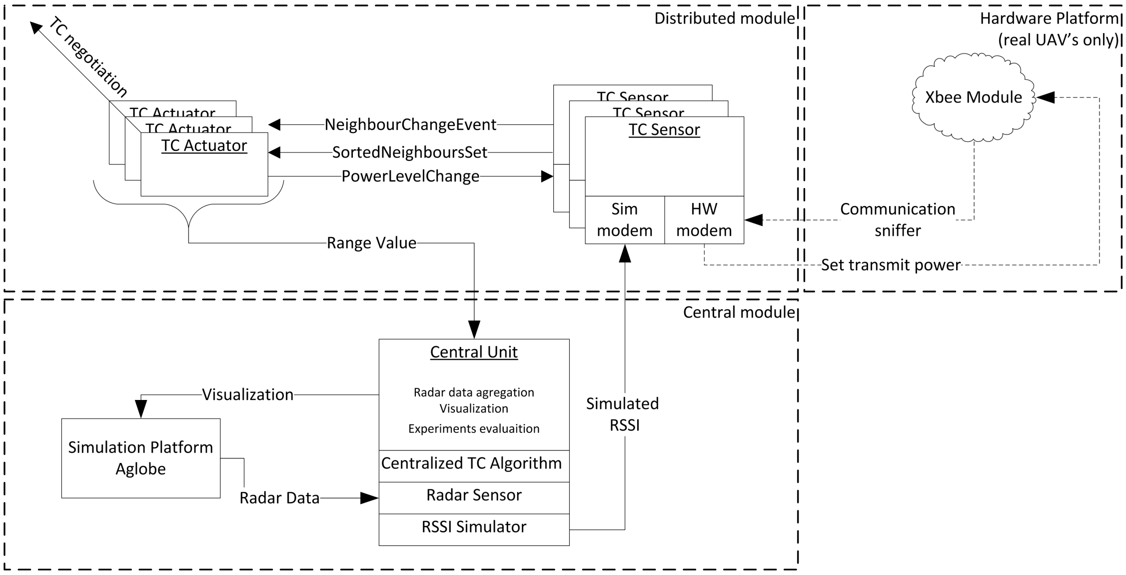


Figure 5.1: Topology control module implementation scheme

5.1 TC protocols

In this section we will present a set of algorithms, which are suitable to perform topology control for proposed scenarios with respect to utilized hardware platform properties and capabilities. As was noted previously, topology control protocol for networks which contain mobile nodes with at least moderate mobility should be fast enough and should utilize information somehow resilient to mobility of nodes. From this follows, that desired protocol should perform some simple heuristic in distributed manner, which will utilize neighborhood based information with reasonable sorting. Therefore next will be present three broadly cited neighbor-based algorithms, that are all based on simple heuristic. First two algorithms - K-NEIGH and XTC were originally developed for stationary networks, but from their properties make them suitable for mobile networks as well.

When the TC protocol for stationary network is to be used in mobile networks, then also its execution scheme has be introduced. There are two possibilities, where first one is proactive periodical execution of protocol. This proactive scheme may lead to useless TC execution in cases when the neighbor model stays the same. Proactive scheme may therefore waste the bandwidth resources. This is closely tied with the need of execution frequency determination. The execution frequency should somehow follow the rate of change in network. Too low frequency leads to topology which does not respect actual state of network and therefore disconnection could occur. When the frequency is unnecessarily high mentioned wasting of resources increases as well. Contrary reactive execution scheme triggers TC protocol only in the situation when the routing protocol experiences inaccessibility of neighboring node. It is clear that this approach causes frequent fails in upper communication layer. In most of scenarios it is inadmissible to wait with almost each message for execution of TC protocol. Solution could be in carefully designed combination of these two approaches. Setting reasonable frequency of execution with respect to mobility rate of concrete scenario with reactive trigger to solve problems caused by underestimation of this frequency.

Third presented algorithm - LINT is also neighbor-based TC protocol, which was designed especially for mobile networks. It's main asset lies in trigger mechanism. Contrary to previously introduced protocols LINT does not handle its neighborhood size equal to predefined number, but in more loosely within interval centralized around this desired number. Therefore it is capable to utilize reactive execution scheme, which is triggered by violating these interval boundaries. It is easy to see, that this situation appears less frequently than increase (decrease) of neighbors number by one, as in previously introduced reactive scheme. This leads to automatic trigger frequency estimation, which corresponds with actual mobility behavior in node's neighborhood.

5.1.1 K-NEIGH protocol

K-NEIGH [4] is the fully distributed, asynchronous and localized TC protocol designed for stationary networks, which tries to keep the number of neighbors for each node equal or a bit lower than given k value. The value of k, which provides connectivity of communication graph with probability of 0.95, was estimated by the authors for cases, where the nodes are uniformly distributed in the unit square. Preferred value of k was shown to be in interval [6-9] for wide range of network sizes (10-500 nodes). K-NEIGH protocol ensures symmetric communication graph with bounded physical degree of each node, and therefore is not able to provide worst case connectivity. Each node is forced to transmit only two messages on full power, where the first one is just beacon message with ID of node and second one contains the first k neighbors from the surrounding. Authors proved that algorithm terminates at each node after $2n$ messages are exchanged. Also the extra local optimization technique was proposed for this TC protocol, which is able to identify and drop redundant links. These links form triangle in communication graph and therefore the energy efficient two hop path should be used instead of longest edge in triangle. This tuning process is performed locally by each node, utilizing currently available information and preserves symmetry of communication graph.

5.1.2 XTC protocol

XTC protocol [20] is an instance of neighbor-based protocol with unbounded resulting physical node degree. This feature makes preserving of worst case connectivity available contrary to previously presented K-NEIGH protocol. This trade-off between worst case connectivity and bounded physical node degree arises from previously presented property of k-neighbors graph. For neighbors ordering XTC utilizes more general 'link quality' criterion. As noted above, this criterion shouldn't be based only on some distance estimation (as RSSI in K-NEIGH), but on some other link quality estimation like packet lose or corruption statistics. Each node is forced to exchange only two messages at maximal transmit power, where the first one is just beacon message (same as in K-NEIGH) but the second includes the ordered list of all its neighbors (not only the first k as in K-NEIGH). Final network topology computation is then done by each node locally. From these facts follows that XTC protocol is lightweight in terms of message exchange overhead and terminates at each node after message exchange. There are no special guaranties on nodes degree in resulting topology, but authors proved that, when the 'link quality' sorting criterion coincide with distance-based criterion, than produced topology has logical node degree at most 6 and its planar.

5.1.3 LINT protocol

LINT (Local Information No Topology) protocol [12] provides fast and simple heuristic, which is able to keep network connected in presence of node mobility. LINT is a neighbor-based protocol, which maintains the number of neighbors in predefined lower and upper bounds. These bounds are centered around desired number of neighbors. The mechanism relies on sensing of actual number of neighbors, which should be outsourced to other layer of communication stack. Authors present, that it is possible to obtain this information from routing protocol. In this special case it is possible to say that LINT requires no message overhead, because it uses only locally available information. This statement is little bit weak, because we do not count the overhead, which produces the routing protocol to identify neighbors. But in scenarios, where regular and continuous traffic is expected, this statements hold. In other cases, where traffic is not so uniformly distributed in time and among nodes, some beaconing mechanism should be introduced, which ensures that estimation of neighbors number is reasonable. Carefully designed beaconing mechanism than will cause message overhead, which has upper bound comparable with overhead produced by periodically executed K-NEIGH or XTC. Finally should be noted, that estimation of proper value of desired neighbors number refers to previously mentioned problem of CNN.

5.2 TC Actuator module

Each of TC algorithms presented in previous section was implemented in as standalone module, because each should be appropriate for different scenario. In chapter 6 we will present results of experiments with different combination of scenarios and TC algorithms, which should clarify the difference in its employment scheme.

The TC actuator module is a runnable software object, which contains reference to appropriate TC sensor. This TC sensor provides required information about node's neighborhood as actual set of neighboring nodes in descending order of link quality value. Each implemented algorithm uses also the above mentioned reactive recomputation scheme and therefore implements method for trigger events sensing. This events are produced by TC sensor module as neighbor state change message. This message contains insert/remove node information and is triggered according to visibility induced by current topology.

Along with TC sensor the TC actuator object belongs into the distributed part of module, which means that for each plane (simulated or real) one such object exists to maintain TC behavior of the entity. The main difference between simulated and real UAV state is that real UAV representation in Tactical Agentfly project is partitioned into two parts. First one is located on-bard. This on-board part provides communication with other on-board HW (e.g. Xbee module), senses the real state of UAV and executes flight plan according to mission control system. Second part, which is located at ground simulation platform serves as simulation representation of the real UAV and provides perception and communication with other simulated entities.

The negotiation among TC actuator objects is realized through TC sensor module. As you can see in 5.2 delivery way differs for real-to-real and real-to-sim UAV channel. The TC message is broadcasted on full power through Xbee module and at the same time is delivered to A-globe communication layer, which should determine visibility of simulated units in the platform and provide correct message delivery to such units.

5.3 TC Sensor module

This module is responsible for providing communication channel to TC Actuator, as was noted above. Also this sensor is able to sense link quality for neighboring nodes. The

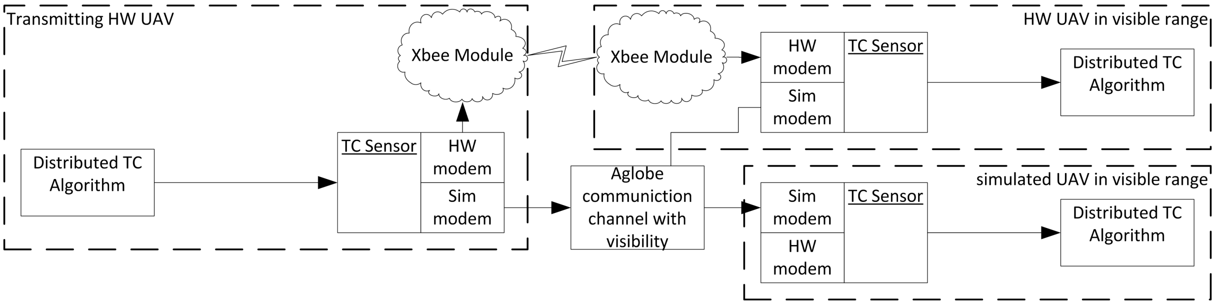


Figure 5.2: Exchange of TC message between real UAV and other real/simulated UAVs

link quality information sensing and accumulating has to be done in two different ways, because this module differs between real and simulated units in its neighborhood.

For real UAVs information about link quality is gathered from communication sniffer. This sniffer is connected directly into packet connector in communication stack. When any message is received from other unit through Xbee module, the received RSSI with transmitter power level is reported to the TC sensor, which than updates link quality record cache. To ensure, that each unit in full power communication range will be identified and link quality will be ascertained, the beaconing mechanism was implemented. TC sensor of each real unit periodically broadcasts short beacon message on full power. This message contains only ID of transmitting unit and power level flag. Only other TC sensors in full power transmitting range register this message and therefore it does not interfere with concurrent message transmission.

For simulated units link quality information is provided by Central unit through RSSI simulator, which is described below in section 5.4. In TC sensor is this update channel represented by Sim Modem component, which contains communication channel to RSSI simulator. RSSI simulator periodically informs about current RSSI value of each neighboring unit in full power visibility range.

Each link quality update received from both above mentioned components is than inserted into LinkQualityCache, briefly introduced in figure 5.3. It is implemented as self updated timeout cache. Each inserted record is provided with timer, which removes this record when timeout is reached. When the last record is removed for some unit, than *node removed* event is reported to TC sensor, which relies it into the TC actuator NeighborStateChange trigger. In opposite case, when the first record is inserted for some unit, *node added* event is also reported. In other cases the new value is only appended at the end of cache. This multi-record cache is useful to suppress extreme values, which could appear because of hardware malfunction or other interference errors. The record with ID=y in figure 5.3 refers to the case of simulated unit. In this case only one record is required, because the RSSI simulator is supposed to produce correct value for each generated update.

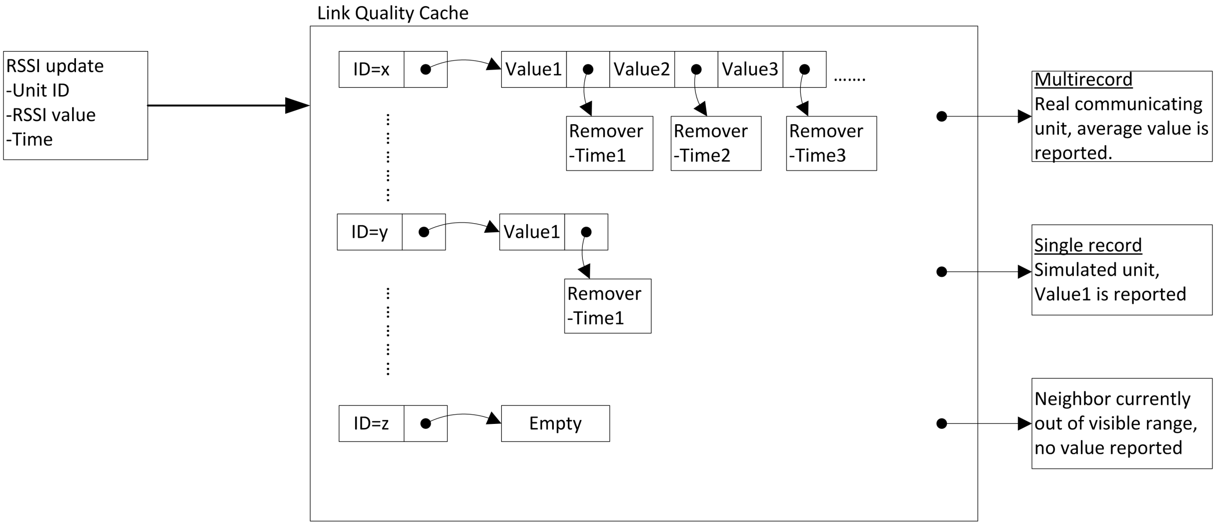


Figure 5.3: Cache for accumulated link quality records with self removing timers and time window averaging

5.4 Central unit

Central unit is simulated entity, which is responsible for accumulation of information about the TC system state, when the experiment is in progress. It has complete information about units' positions and actual power levels set. Therefore this unit is responsible for visualization of real-time state and production of online experiments results. For comparison purposes CentralUnit periodically computes centralized solution for TC problem by MST (Minimum spanning tree) algorithm. This TC solution is supposed to be the optimal one for 1-connected communication graph TC problem and it should appear as subgraph of other distributed solutions. The frequency of MST computation could be set quite often, because the algorithm is just simple weighted DFS (Depth First Search).

Fundamental part of the CentralUnit is the radar data aggregation mechanism, which provides for each node in simulation its actual position and distance to each other node. Telemetry updates for each node in simulation are collected and position update also triggers mutual distance update for other nodes in simulation. This continuously updated distance matrix is the basic collection, which is utilized by above mentioned central TC solution provider and visualization layer. Through TC sensor module the actual transmit power level setting for each node is also collected and later utilized for computation of real time experiment statistics and visualization.

Online visualization in mission control interface is done according to accumulated information about nodes' positions, distances and ranges. Range assigned to each node is visualized as a circle with appropriate diameter with center at node's actual position. Links between each node pair are expressed as lines. Different line colors are applied for bidirectional and unidirectional links. Centralized solution is for better visual comparison drawn by dashed lines and circles. Each element of visualization is handled as standalone key toggle graphic layer, which enables the operator to hide each layer separately.

For purpose of experiment evaluation online statistic are computed. The connectivity in the network is expressed as size of biggest connected component. Than the average value of logical and physical node degree is computed to measure the actual network capacity. Finally average and maximal range in the network is observed to provide information about power efficiency. Each of these proposed values is measured and logged for distributed and centralized TC solution to provide data for its comparison.

5.4.1 Link quality simulation

In this section we will present Link Quality simulation module. This module provides information about quality of link for simulated entities, which are not equipped with real Xbee module. Real entities equipped with XBee modules are supplied with RSSI value from every incoming datagram from other real entity. It is clear that in scenario, where simulated entities appear as well it is fundamental to provide same information from these simulated entities. In previous section was pointed that each implemented TC algorithm uses acquired link quality value to sort neighboring nodes. We need some estimation mechanism, which is able to simulate incoming RSSI values to provide stable mixed simulation.

Imagine a situation, when two units are on the same coordinates and in the same attitude. One of this unit is real UAV equipped with an Xbee module and second is simulated UAV. Another real UAV receives signal of some strength S_r through Xbee module from the first distant entity. We have to provide simulated signal with same (or very similar) strength S_s from the second distant unit. In the situation, when the received values of real and simulated signal differ a lot, then the ordering of neighbors is unstable and TC algorithm could make wrong decision about transmitting power level assignment. This situation strongly affects correctness of simulation.

Proposed mechanism for simulation of RSSI is based on stochastic sampling from given statistical model. The model should represent the real data obtained from Xbee modules. Data set, which was used to identify above mentioned model was collected in field. There was a simple utility implemented, which provides permanent loopback communication between couple of units. One loop provides two measured values of incoming RSSI. To each received RSSI value there was also the on board telemetry of each unit logged. This telemetry log was next used to compute mutual attitude. Collected dataset was strongly polluted with noise, therefore Savitzky-Golay filtering [15] was applied to smooth the data and correct outlying values. In 5.4 is figured original data set and result after filter application.

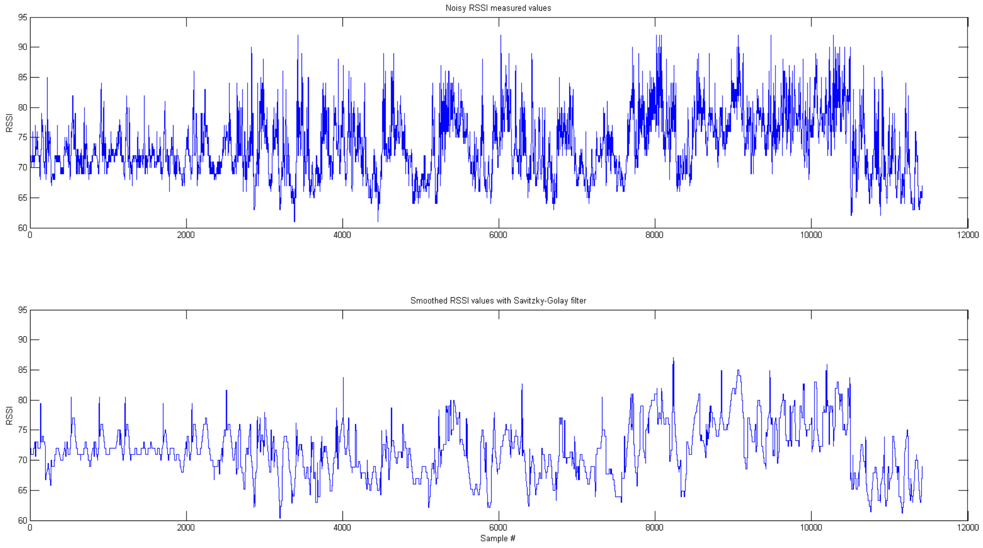
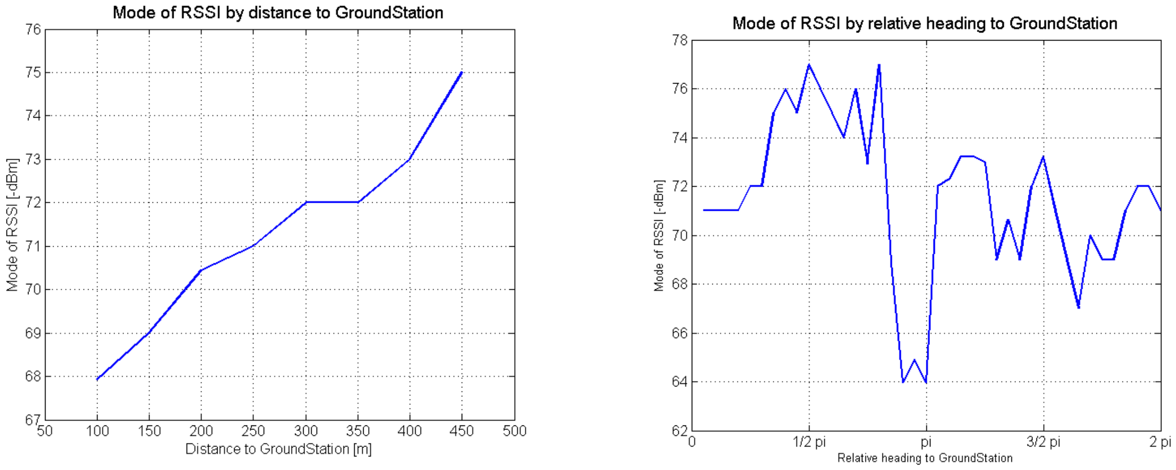


Figure 5.4: Original and smoothed RSSI data

Next dependency of particular mutual attitude of units on average RSSI was identified. It is clear that RSSI value mostly depends on mutual distance of units. Second dependence was found in mutual heading of units. It was not so surprising, because of specific radiation characteristic of utilized dipole antenna, which is localized on the end of UAV's wing. In figure 5.5 is shown the mode of measured RSSI value for accumulated bins of different mutual distances and headings. The attenuation increase with distance as follows from path loss specification. In second plot different situation is depicted. We can see that when mutual heading of units is about $\pi/2$ the attenuation is strong but in situation when mutual heading is near to π attenuation of signal decreases rapidly. This could be interpreted as situation when UAVs fly away from each other so antennaes are aimed directly against themselves. The increase on $\pi/2$ than should be caused by the fact that in this mutual attitude one of dipole antennas is covered behind the electric motor, which emits high frequency jamming.

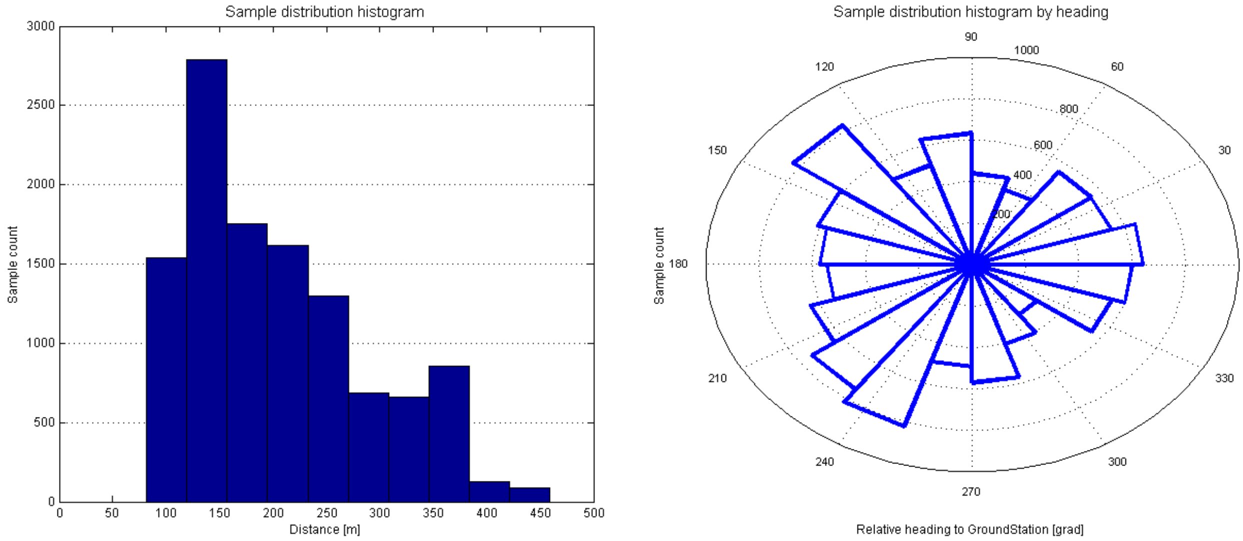
Before the model will be fitted on data it is a good practice to check if we have enough portion of data for each considerable interval of mutual attitude component. In this case we want to treat data from mutual distance and heading point of view. Figure 5.6 shows histogram of data distribution according to different bins of these considered descriptors. From the figure follows that model will be probably inaccurate in high mutual distances



(a) (b)

Figure 5.5: Mode of measured RSSI value in mutual (a) distance and (b) heading

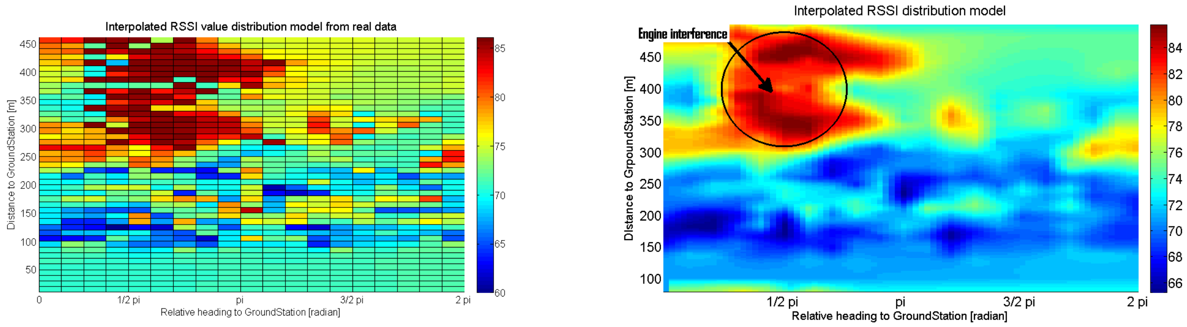
(greater than 400m). In heading data are distributed quite uniformly, from this point of view model will be accurate enough.



(a) (b)

Figure 5.6: Histogram of data distribution in mutual (a) distance and (b) heading

Finally missing values should be interpolated. It is clear that field experiment is not able to provide every mutual position of units. But data set collected was huge enough, so the gathered model before interpolation was not too sparse. Therefore new data generated by interpolation should be reasonable for fitting of correct model. Figure 5.7 shows the model of average values after interpolation and than after smoothing. In the smoothed model is in addition highlighted the anomaly caused by electric motor interference, which was pointed in previous part of this section. Complete model is formed as Gaussian distribution for each considered combination of mutual heading and distance. When the RSSI simulation module is requested for value, it computes the mutual attitude of queried units couple and than samples the value from appropriate distribution from the model.



(a) (b)

Figure 5.7: Model of average values after (a) interpolation and (b) smoothing

5.5 Communication stack

Developed communication stack consists of three separate layers, where each layer has a software interface defined, which enables simple plug-in of different modules.

5.5.1 Interface layer

First is the Interface Connector layer, which is responsible for establishing the connection to proper HW interface and handling appropriate Input/Output streams. In this specific case the HW platform is the RS-232 serial interface. As was noted before, Java platform is not equipped with some native library to access this kind of interfaces, therefore two independent implementations of this interface were introduced.

- RXTX library - platform independent library using a native implementation (via JNI), providing serial and parallel communication for the Java Development Toolkit (JDK).
- TCP pipe using external platform dependent utility providing the pipe between serial connector (OS interface) and virtual TCP socket (server) created on local host, which is then interfaced through native java.net library.

5.5.2 Packet layer

Packet Connector layer is responsible for proper transformation of packets to data frames, which should be directly transmitted through Interface Connector. This transformation process has to fit given communication pattern (Xbee API mode, transparent mode) and consists of following three parts:

- Start character and frame length write and read
- Encode/decode escape characters
- Checksum computation and backward consistency verification

When Packet Connector receives frame bytes, it checks the entire packet integrity, by sensing the start character and checking the mentioned length of the frame. To provide continuous reading from input stream, the producer/consumer pattern was used in separated thread for packet passing.

5.5.3 Message layer

Message Connector, the most complex layer of communication stack, incorporates lightweight transmission control, routing and medium access control (MAC) protocols. Transmission control protocol enables to exchange messages, which are bigger than 100B (maximal size of RF packet data part), and ensures successful receiving of messages in unicast communication. It is responsible for message segmentation and transmission. Messages are transmitted according to sliding window schema. In unicast communication the acknowledgment mechanism is utilized, which allows to retransmit fragments, which were corrupted. Transmission method is extended with link fail sensing, which is based on packet delivery success rate. This mechanism is crucial to prevent communication from getting stuck on the message, which cannot be delivered because of target node or link failure. When the message is dropped then the route finding process is triggered and state is reported to message originator. Message state reporting process is based on event handler registration mechanism. Various message states were introduced (QUEUED, SENDING, SENT, FAILED, EXPIRED etc.) and each enrolled handler in upper layers is notified about message state change. In considered application scenarios the priority and expiration time of exchanged messages could be also defined, therefore the outgoing message buffer is designed to provide appropriate expiration and prioritization mechanism. This forces communication layer to transmit messages with high priority first and prevents delays by removing of expired messages. Also the Message Connector is responsible for correct assembling of received message fragments. Because more than one unfinished message could be in reception process at the moment, separate receive buffer is created for each sender. This could happen, when the message is too large to be transmitted in one time slot (see medium access mechanism). The self-removing timeout mechanism was introduced for the receive buffer as well to ensure that the sender and receiver stay synchronized after re-establishment of lost connection. Timeout should expire in situation, when the sender detects link failure and drops the partially delivered message.

Routing part of the Message Connector is for now implemented just for direct delivery (without multihop delivery and route finding mechanism). The node neighbor list is maintained in passive fashion by communication overhearing and also through active nodes discovery procedure, which is available as service of the Xbee platform. Active approach is triggered when there is no communication performed in the node vicinity and as a simple route finding procedure.

Communication module utilizes two-layer approach to access the wireless medium, which is helpful to avoid collisions in transmission. Collisions are the main problem of proposed communication platform and incorrect collision avoidance mechanism is mainly responsible for strong decrease of communication throughput. First layer utilizes standard CSMA/CA mechanism with exponential random backoff. This MAC protocol is implemented directly in the Xbee platform and avoids RF packets to collide. But in situation when one node starts to transmit large message, the other nodes experience frequent transmit failures due to the backoff expiration. Therefore the second layer was introduced, which utilizes RTS/CTS handshake mechanism with channel reservation. Second level of MAC protocol gets only triggered, when the message is split into more than two fragments. There is a maximal size of time slot defined as well, which forces the node transmitting huge message to offer the wireless medium to other nodes from time to time.

6 Experiments

In this chapter results that show behavior of proposed and implemented TC algorithms will be presented. All simulations were carried out using a Tactical Agentfly framework, which was described in chapter 4. First s specific description of utilized scenarios will be introduced. Results of experiments will precede explanation of particular measurements. Finally we will conclude this chapter with some practical results, which follow from experiments.

6.1 Scenarios - closer look

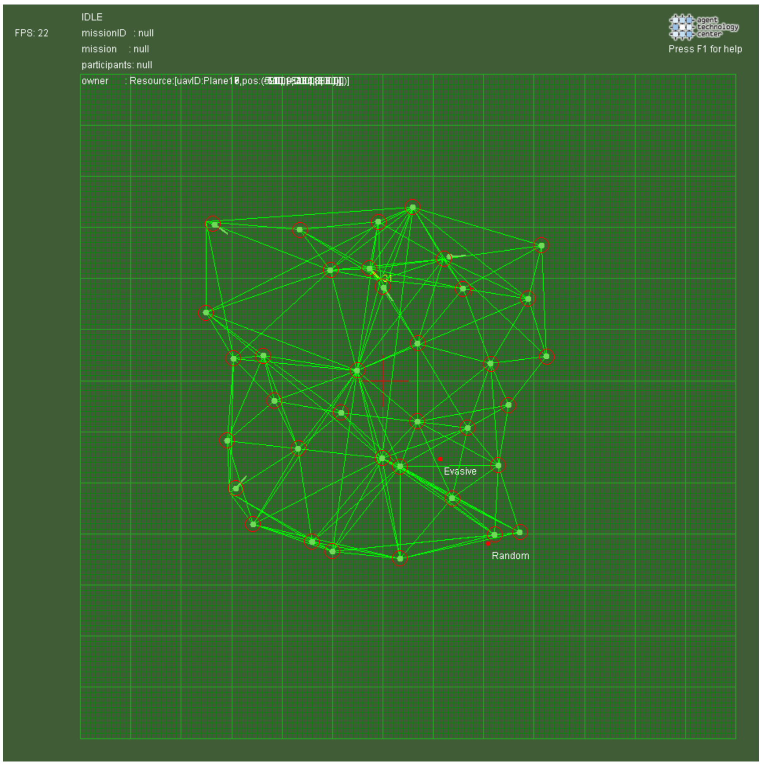
As was explained in section 4.4, two different scenarios were introduced to compare the behaviour of proposed TC algorithms, but there are some common characteristic properties. Scenarios are in both cases composed of *36* nodes. Nodes are sampled uniformly at random into square grid, which is centered in origin of the cartesian coordinate system. The reason for number *36* of nodes follows from the property of simulation system. This number is the greatest one, which creates uniform grid *6x6* and it is capable to compute its simulation on standard personal computer. Positions in grid are slightly rearranged with small synthetic noise. Both scenarios contain same initial position of each node, but the difference is, if units are allowed to move.

First case is a *low-mobility* scenario, because only *4* units are regular UAVs, which are allowed to move. Other units in scenario represent ground wireless sensor network (WSN). UAVs in this scenario accomplish the surveillance mission in the same region. From this follows that the main part of network is not forced to change its range assignment. In figure 6.1 is shown the screenshot for this scenario. Green dots stand for nodes. In the corners of deployment region are shown UAVs, which differ from the WSN nodes with direction vector.

Second proposed scenario, which is based on the same initial position of the nodes as in previous case, is called high-mobility scenario. Contrary to the first scenario most of the units are able to move and stands for standard UAVs. Only one unit, placed in the center of deployment is immobile and represents the operation center or data collecting gateway. In figure 6.2 is clear to see, that nodes have changed their positions significantly in comparison to initial positions from figure 6.1. This figure also shows surveillance regions, which are assigned as mission goals for separate groups of UAVs.

6.2 Topology control characteristics

To verify capabilities of TC algorithms to produce suitable solution of topology control problem, there should be two main properties observed. Each of these properties



missionID : null
mission : null
participants: null
owner : Resource:[uavID:Plane1[illegible],pos:([illegible])]

Figure 6.1: Screen shot of low-mobility scenario with highlighted symmetric links

represents one of the TC protocol goals:

- Average transmit range assigned by TC protocol
- Connectivity satisfaction or the giant component size to perform comparison of two inadmissible heuristics

In the case of mobile ad hoc networks it is necessary to measure these values for the whole network lifetime, because the network topology could change rapidly in time, when mobility is introduced. We want this time statistic to show, that benchmarked algorithm is able to provide good results for the most of the network lifetime.

6.3 Experiment's setting

Two of implemented algorithms were compared. First one is the simple K-NEIGH protocol, which represents cluster of TC algorithms, which were originally proposed for stationary networks. This protocol needs the k value to be assigned. This value was determined through preliminary tests, which were performed on low-mobility scenario, and the most suitable value appears to be *k=6*. Second value, which should be assigned correctly is the recomputation period, which depends mainly on mobility rate. For our purpose *1000 ms* period was determined.

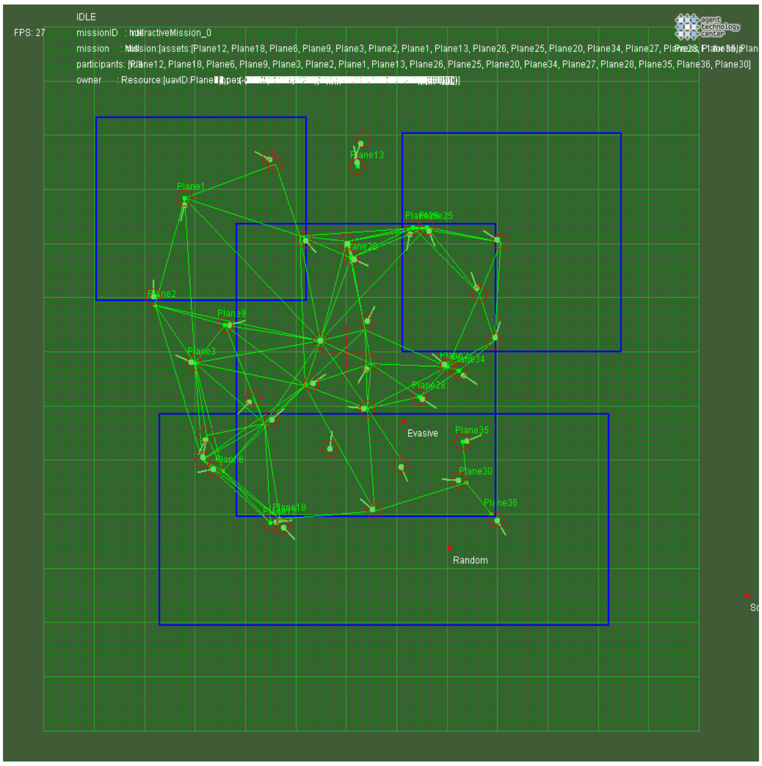


Figure 6.2: Screen shot of low-mobility scenario with highlighted symmetric links

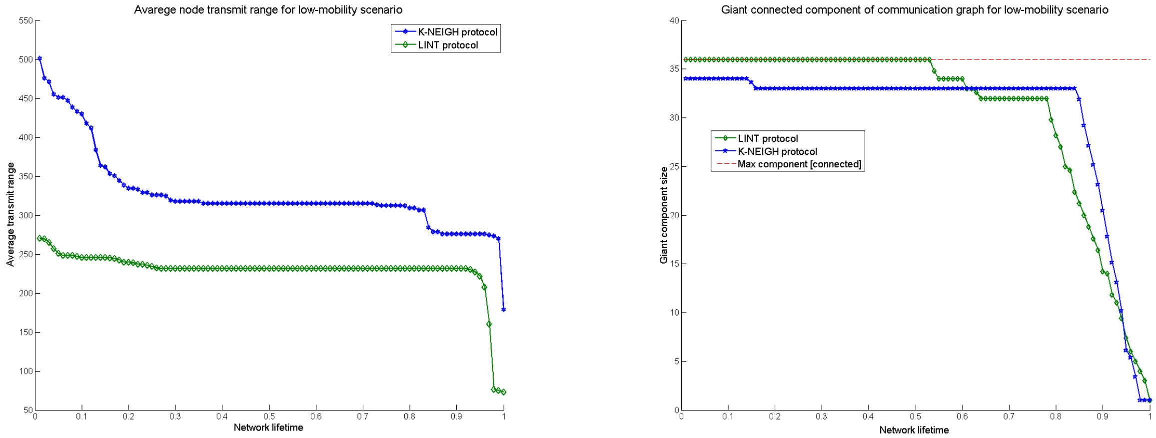
Second implemented algorithm was designed directly for use in mobile ad hoc networks. Because this algorithm belongs to neighborhood-based TC protocols, there should be the minimal and maximal number of direct communication links defined. These values are clearly presented in table 6.1. Table 6.1.

k-desired	5
k-minimal	3
k/maximal	8

Table 6.1: K parameters for LINT protocol

6.4 Results and comparison

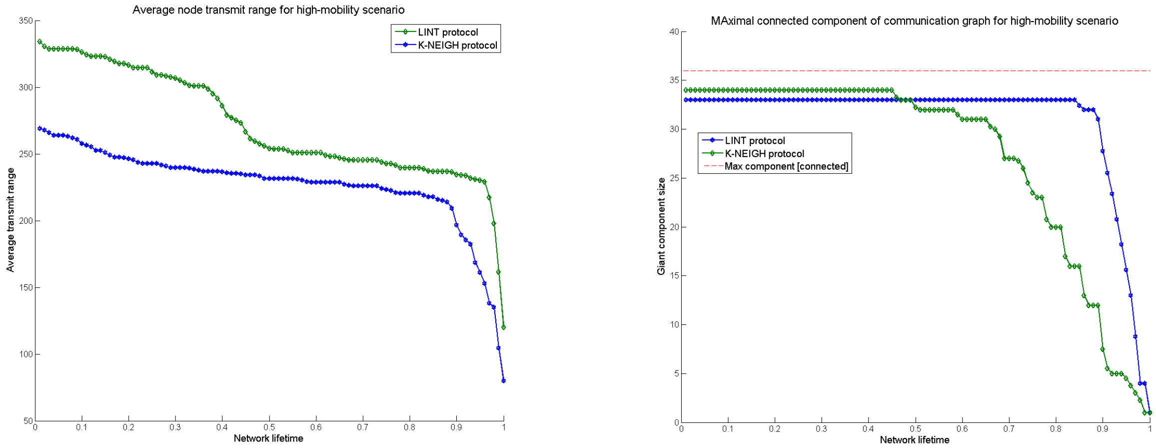
Proposed experiment were carried out to show the difference in resulted topology produced by couple of TC algorithms, which differ in the manner of mobility handling. In the first testbed each TC protocol was deployed in low-mobility scenario, which was described above. The results are shown in Figure 6.3. In chart (a) we can see that LINT algorithm is able to provide significantly lower average transmission range, which leads to better performance of communication. In chart (b) the trade-off between network connectivity and low range assignment is evident. It seems, that the K-NEIGH protocol



(a) (b)

Figure 6.3: Results of experiments for low-mobility scenario in (a) average transmit range and (b) giant component size

was outperformed by LINT, but it was achieved only by the fact that in this case network was partially disconnected for the whole network lifetime. K-NEIGH algorithm fulfils the connectivity requirement for more than half of network lifetime.So we can conclude that for this case of mobility the K-NEIGH algorithm is more suitable than LINT in terms of network connectivity maintenance.



(a) (b)

Figure 6.4: Results of experiments for high-mobility scenario in (a) average transmit range and (b) giant component size

Results for the second scenario are shown in figure 6.4. Chart (b) shows that connectivity of the whole network was never achieved by either tested protocol. This should not be surprising, because for network with such mobility rate full connectivity is probably unachievable. But in this case we can als see, that LINT protocol outperforms K-NEIGH by handling almost full connectivity for nearly whole lifetime of the network. It is fact, that K-NEIGH was able to achieve little bit better connectivity, but only for half of network lifetime. The dominance of LINT algorithm in this particular scenario is supported with result of average transmit range (Figure 6.4 (a)), which is significantly better than range achieved by K-NEIGH for whole network lifetime.

Above presented results illustrate the necessity of proper TC method selection according to considered target scenario, where the method will be utilized. Therefore the whole system presented in this work was designed with this in mind and a modular framework supporting simulation focused on mission control was implemented.

7 Conclusions

This work presents introduction into topology control problem with brief formal introduction of related topics. Also the taxonomy of topology control method was mentioned. Analysis of features of approaches to topology control problem was used to specify requirements for deployment of these algorithms into the real communication platform. The hardware platform was described in details and its specification and restrictions relevant to topology control were identified. Based on this topology control mechanism was implemented as part of the communication stack which uses Xbee communication platform in scenarios with UAVs. Tactical Agentfly project, based on the A-globe simulation platform was briefly introduced, with emphasis laid on approaches to integration of simulated and real entities. Typical scenarios investigated within this project were identified and then specified as experimental testbeds. A batch of experiments for two different implemented topology control protocols was carried out for these scenarios and bases on results specific models of mobility suitable for both of them were identified. Whole implementation of described topology control module has been done with emphasis on its deployment in mixed simulation in mind.

Future Work

Work carried out so far just prepares ground for deeper research in the field of topology control methods on real hardware platform. The main area of future research will be in proper handling of mobility and especially so-called group mobility, which is quite frequent phenomenon in mission aware scenarios. Another area of future work should be formalization of the topology control problem in the terms of Distributed Constraint Optimization Problem (DCOP) in a form suitable for some existing and verified DCOP solver.

8 Bibliography

[1] F. Adelstein, S. K. Gupta, G. R. III, and L. Schwiebert. *Fundamentals of Mobile and Pervasive Computing.* McGraw-Hill, 2004.

[2] C. L. Barrett, M. Drozda, M. V. Marathe, S. S. Ravi, and J. P. Smith. A mobility and traffic generation framework for modeling and simulating ad hoc communication networks. *Scientific Programming*, 12(1):1–23, 2004.

[3] C. Bettstetter. Mobility modeling in wireless networks: categorization, smooth movement, and border effects. *Mobile Computing and Communications Review*, 5(3):55–66, 2001.

[4] D. M. Blough and G. Resta. The k-neigh protocol for symmetric topology control in ad hoc networks. In *In Proc. of MobiHoc*, pages 141–152, 2003.

[5] C. chung Shen, C. Srisathapornphat, R. Liu, Z. Huang, C. Jaikaeo, and E. L. Lloyd. Cltc: A cluster-based topology control framework for ad hoc networks. *IEEE Transactions on Mobile Computing*, 3:18–32, 2004.

[6] M. S. G, P. Manzoni, and Z. J. Haas. Determination of critical transmitting range in ad hoc networks. In *Proceedings of Multiaccess, Mobility and Teletraffic for Wireless Communication Conference*, 1999.

[7] X. Jia, D. Kim, S. Makki, P. jun Wan, and C. wei Yi. Power assignment for k-connectivity in wireless ad hoc networks. In *In Proceedings of 24th Annual Joint Conference of the IEEE Computer and Communication Societies(InfoCom*, pages 213–222, 2005.

[8] V. Kawadia and P. R. Kumar. Power control and clustering in ad hoc networks. In *In INFOCOM*, pages 459–469, 2003.

[9] L. M. Kirousis, E. Kranakis, D. Krizanc, and A. Pelc. Power consumption in packet radio networks. *Theoretical Computer Science*, 243:289–305, 2000.

[10] A. Komenda, M. Cap, and M. Pechoucek. Multi-agent simulation approach to development of applications for decentralized tactical missions. In *Knowledge Systems for Coalition Operations (to appear)*, 2012.

[11] M. D. Penrose. On k-connectivity for a geometric random graph. *Random Struct. Algorithms*, 15(2):145–164, 1999.

[12] R. Ramanathan and R. Rosales-Hain. Topology control of multihop wireless networks using transmit power adjustment. In *In Proc. of IEEE Infocom 00, Tel Aviv*, pages 404–413, 2000.

[13] T. Rappaport. Wireless communications: Principles and practice, 2nd edition. Prentice Hall PTR, 2001.

[14] P. Santi. Topology control in wireless ad hoc and sensor networks. John Wiley & Sons Inc, 2005.

[15] A. Savitzky and M. Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry*, 36(8):1627–1639, 1964.

[16] D. Šišlák, M. Rehák, M. Pěchouček, M. Rollo, and D. Pavlíček. **$\mathcal{A}$-globe**: Agent development platform with inaccessibility and mobility support. In R. Unland, M. Klusch, and M. Calisti, editors, *Software Agent-Based Applications, Platforms and Development Kits*, pages 21–46, Berlin, 2005. Birkhauser Verlag.

[17] D. Šišlák, P. Volf, Š. Kopřiva, and M. Pěchouček. *AgentFly: Scalable, High-Fidelity Framework for Simulation, Planning and Collision Avoidance of Multiple UAVs*, chapter 9, pages 235–264. Aerospace Series. Wiley: John Wiley & Sons, Inc., 2012.

[18] J. Vokrinek, A. Komenda, and M. Pechoucek. Abstract architecture for task-oriented multi-agent problem solving. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 41(1):31 –40, January 2011.

[19] J. Vokrinek, P. Novak, and A. Komenda. Ground tactical mission support by multi-agent control of uav operations. In *Proceedings of the 5th international conference on Industrial applications of holonic and multi-agent systems for manufacturing*, HoloMAS'11, pages 225–234, Berlin, Heidelberg, 2011. Springer-Verlag.

[20] R. Wattenhofer and A. Zollinger. Xtc: A practical topology control algorithm for ad-hoc networks. In *In 4th International Workshop on Algorithms for Wireless, Mobile, Ad Hoc and Sensor Networks (WMAN)*, 2004.

[21] L. Zhao. Topology control for mobile ad hoc networks, 2007.

A Content of the enclosed CD

The CD attached to this work contains source codes of implemented topology control module as well as compiled distribution for running a demo with simulated UAVs. Further, there is the module documentation, LaTeX codes of this work and the work itself as a pdf file.

- Directory `source_codes` – contains source codes of implemented classes
- Directory `demo` – contains compiled demo. The demo can be started by running the `test.bat` file.
- Directory `pdf` – PDF version of this work.