FRONTS - Foundations of Adaptive Networked Societies of Tiny Artefacts

Tobias Baumgartner, Alexander Kröller, and Sándor P. Fekete
IBR, Algorithms Group
Braunschweig University of Technology
Braunschweig, Germany
{t.baumgartner, a.kroeller, s.fekete}@tu-bs.de

Claudia Becker and Dennis Pfisterer
Institute of Telematics
University of Lübeck
Lübeck, Germany
{becker, pfisterer}@itm.uni-luebeck.de

ABSTRACT

The European project FRONTS aims at understanding adaptive large-scale heterogeneous wireless sensor networks acting in a dynamic environment. This includes designing different mathematical models beyond the state of the art for such networks. Based on these models, a new network infrastructure will be developed, followed by a large set of algorithms solving global tasks like exploration of unknown areas or target tracking. We provide a coherent package consisting of a simulation framework, hardware, and a remotely accessible testbed, building a powerful tool for evaluation and validation of the results of FRONTS. Our framework allows for easy exchangeability between simulation and real-world experiment, as well as rapid engineering.

Keywords

Wireless Sensor Networks, Simulation, iSense, Shawn

1. INTRODUCTION

We are now on the verge of Wireless Sensor Networks (WSNs) leaving the landrush phase and becoming more and more mainstream. Many complex issues have been addressed, and we see a number of operating WSNs. This shifts the focus from initial questions about how to get working networks at all, to those about the shape of future networks and their applications. Being relieved from dealing with complex low-level issues, we now can concentrate on high-level applications and protocols.

It is our belief that soon sensor networks will exist with a number of properties:

Heterogeneity: WSNs will no longer consist of identical devices. The need to maintain networks over long periods of time will result in different software and hardware generations from one family to coexist. Also, there will be completely different devices in networks due to interactions between networks, the need for specialized sensory in certain areas, and because networks will grow organically.

Dynamics: Other than in today's networks, different kinds of dynamics will happen at the same time in a network. Some nodes may enter and leave the network due to off-duty cycling. Others are moving with some external carrier like vehicles, animals, or water flows. Finally, some nodes may actively move around to investigate events, help bridging communication gaps, or populate unexplored areas with passive nodes. Another source of dynamics is the environment. In long-lasting networks, there may be huge changes in the surroundings and therefore in the constraints and objectives of the network.

Massive Sizes: When issues about information exchange between neighboring sensor networks are solved, huge sensor networks arise instantly. Also, continuously decreasing hardware costs, and ongoing research on real sensor nodes lead to the feasibility of obtaining a great amount of nodes. Hence, current algorithms that were tested on mostly a few hundred nodes must also scale up to hundreds of thousands of nodes.

The FRONTS project [2] is part of the European Seventh Research Framework Programme (FP7, [1]). It is a collaboration of 11 European working groups, aiming at investigating the algorithmic foundations of future WSNs and providing answers for questions that arise now. The participants have a great deal of experiences with the fundamental theory of wireless sensor networks. Together, we will contribute research to the following issues:

- Development of validated mathematical models for large-scale wireless sensor networks beyond oversimplifications such as Unit Disk Graphs, infinite-capacity links, random waypoints movements.
- Design of dynamic and adaptable network infrastructures, including enabling security by developing appropriate algorithms, protocols, and strategies.
- Design of algorithms for solving global tasks like collective exploration of unknown areas, or target tracking in dynamically changing environments.

The paper is structured as follows. The project FRONTS itself and its aims are described in more detail in the following Section 2. The IBR at the Braunschweig University of Technology and the ITM at the University of Lübeck participate in FRONTS. The practical parts of the project are our responsibility. It is our aim to get valuable feedback from implemented algorithms, protocols, and applications that are developed within the project. Feedback will be achieved by evaluations and validations. Therefore we will build up the Experiments Repository, which will consist of an appropriate simulation package for wireless sensor networks as well as at least one physical testbed. It will be remotely accessible via the Internet for all project partners. The requirements concerning software and hardware technologies, and the technologies to cope with these needs are described in Section 3.

2. EUROPEAN PROJECT FRONTS

The aim of FRONTS is to understand the theoretical backgrounds of large-scale adaptive heterogeneous wireless sensor networks. These networks consist of nodes that provide different capabilities and act in a dynamic environment. Dynamics can either be active or passive. When dealing with active mobility, nodes are able to move autonomously to designated parts of the network. Passive mobility means that nodes are connected to moving objects like vehicles or human beings. With FRONTS, we will be able to get a clear grasp of such networks. It will be crucial not only for understanding the net behavior, but also for supporting algorithm development in nearly all related areas.

The goals of FRONTS can be summarized as follows:

- Design a unifying framework,
- Obtain a set of design rules,
- Provide distributed adaptation techniques,
- Provide laws on adaptation,
- Obtain knowledge of combining heterogeneous nodes,
- Provide a set of algorithms, and
- Evaluate with both simulation and experiments.

The main issue is to develop mathematical models that describe the focused adaptive large-scale networks of heterogeneous nodes in dynamic environments. Such models include mobility for dealing with different kinds of dynamics, computation models that consider the limited memory and limited knowledge of the nodes, and cooperation models that cover homogeneous, hierarchical, and diversified systems. With the aid of the resulting unifying framework we intend to get a consistent working set of design rules of such systems. These rules will help us to develop adaptation techniques for which we provide basic laws. This includes the effect of adaptation on the system performance, the cost of distributed coordination regarding adaptation, the communication overhead, overhead in terms of energy consumption, and possible trade-offs. However, these adaptation techniques will help designing a dynamic and adaptable network infrastructure to cover nodes that adapt to each other and to changing needs. This includes both the communication structure between nodes, and general data access even when the data moves dynamically in the network.

Another objective is the combination and interaction of heterogeneous nodes to obtain a useful global behavior. A network of nodes that have different capabilities should combine the various strengths resulting in a net that is much more powerful than a network of homogeneous nodes can ever be. A crucial issue here are strategies for role assignment that act dynamically according to the current needs of the system. Different capabilities of nodes lead also to new approaches for security. For example, work can be outsourced to more powerful nodes in the network, or completely new protocols, algorithms, and strategies can be designed.

Building on the previous work, the participants of FRONTS will examine strategies and algorithms, which are focused on solving global tasks in the network. This will result in a set of algorithms that contain a collective exploration of an unknown area by considering both active and passive mobility in the network, algorithms for target tracking in dynamic environments with either a moving sink to which data must be sent, or a mobile agent that follows the target, and algorithms for network connectivity maintenance with actively moving nodes.

3. EXPERIMENTS REPOSITORY

The previous section described the project objectives. A decisive point are the practical aims of the project, such as evaluation and validation of developed algorithms. We will contribute an *Evaluation Package* to obtain valuable feedback for the project's theoretical insights. This package will consist of a WSN simulator, appropriate hardware, and at least the provision of one sensor network testbed composed of about 50 sensor nodes. The common central testbed will be accessible via Internet for all project partners to test protocols, algorithms, and applications.

We had to cope with the following requirements concerning suitable software and hardware technologies:

- We need a simulation framework that aims particularly at high level algorithms and a quick and easy development process.
- We need a simulator that copes with simulations of large-scale networks with tens to hundreds of thousands of sensor nodes in an adequate time frame.
- We need a simulator that realizes realistic and universally valid, i.e. hardware-independent, evaluations.
- We also need working and easily maintained hardware for our practical evaluations.
- To narrow the gap between virtuality (simulations) and reality (experimentations on real hardware) it is desirable to have the possibility of easy transfers of implemented algorithms from one to the other.

In the following, technologies that address our above specified needs are described. They will be part of our *Evaluation Package*.

Shawn [4] is a discrete event simulator for sensor networks. It has been primarily designed for simulating large-scale networks with up to a million of nodes, with an algorithmic point of view. Instead of simulating a phenomenon itself, it simulates the effects. This approach leads to an essential performance gain. Shawn finishes simulations in minutes where, say, Ns-2 [3] is running for more than a day [8].

Other design focus points are flexibility and extensibility. All crucial parts that can influence the simulation are designed as exchangeable models. First of all is the communication model that defines whether two nodes in the network can in principle communicate with each other. Next is the edge model that is responsible for neighborhood representation. It provides access to links between nodes in a graph structure, using the communication model to decide whether there is a link between any two nodes. There is a very simple edge model with no memory overhead, but wasting computation time, up to a very fast model that requires much memory. Finally, there is the transmission model that is used to transmit messages. It uses the edge model to select potential receivers of a sent message. There are several models available that can also be put into a chain of multiple models. Beginning with a reliable model that simply transmits every message, there are also models for simulating message loss and delay, as well as CSMA and TDMA. All of these models can be arbitrary combined, which leads to very flexible and powerful options in simulation behavior. Also, models can easily be exchanged to run an algorithm in different scenarios without any additional effort.

The simulation controller manages the process of simulation, and provides access to the simulated world with all available nodes. This allows for an easy development of centralized algorithms. It is possible to write a task that is executed once and has access to the whole simulation via the controller. However, Shawn has been primarily designed for simulating distributed algorithms. Each node in the network can contain multiple independent processors, whereby each processor may implement an individual algorithm. These algorithms may either run independently from each other, or interact over common variables that are appended in a type-safe way to the nodes.

The design with processors, tasks, and the previously presented models enables a rapid algorithm engineering with short development cycles per algorithm. This makes Shawn an attractive choice, especially for implementing and evaluating high-level algorithms.

iSense [5] is a hardware and software platform. Its modular approach allows adapting the hardware and software used for a specific application exactly to the required functionality.

A fundamental design guideline is to use only languages and tools that are well understood by a large user community. Consequently, the object oriented C++ programming language and popular tools such as the Gnu Compiler Collection [7] (GCC) and the Eclipse [6] development framework are used. iSense includes a tiny STL-like implementation. Thus, implementations for standard containers such as lists, maps and sets are already part of iSense. Hence, the learning curve for many system designers and solution developers is extremely flat.

iShell is a tool that is provided by iSense. It provides an easy way for the communication between sensor networks and PCs. Functionalities like serial and over-the-air programming are supported, as well as a serial terminal and a plug-in system to facilitate the supplementation of individual desired functionality as monitoring or analyzing the network.

Application code that has been developed with the iSense framework is ready to run on any platform that supports an implementation of the iSense API. Primarily it is available for the iSense hardware platform, as well as for the simulation environment Shawn. Thus, system designers and solution developers are able to test their implemented applications in the simulator before transferring them onto the iSense hardware. That way time will be saved for the overall development process.

Thus, there are three options given to practically evaluate implemented protocols or algorithms:

(1) Simulating with Shawn.
(2) Testing on real hardware with iSense.
(3) Testing on a common centralized sensor node testbed.

When implementing algorithms, protocols, or applications for either of these three options, it is trivial to switch to one of the others afterwards. This is because of the compatibility between Shawn and iSense. This expands the possibilities for evaluating and validating, and reduces the overall time of development processes in addition to all the other technologies' features and advantages mentioned above.

4. CONCLUSION

We introduced the European project FRONTS. The main objectives of the project are to get a fundamental understanding of the properties of wireless sensor networks, to develop algorithms, design rules for algorithms and applications, and to form a unifying framework for adaptive wireless sensor networks.

One of the next steps will be to build up the Experiments Repository. This contains the aim of establishing a repository of applications for simulation, as well as building up a testbed composed of real sensor nodes. Further, the testbed shall be connected to the Internet via the tool iShell that is provided by iSense. That way all project partners will be able to test their algorithms on a common hardware platform.

5. REFERENCES

[1] FP7. `http://cordis.europa.eu/fp7/home_en.html`.
[2] FRONTS. `http://fronts.cti.gr/index.php/home`.
[3] ns-2: Network simulator-2. `http://isi.edu/nsnam/ns`.
[4] Shawn. `http://shawn.sf.net`.
[5] C. Buschmann and D. Pfisterer. iSense: A modular hardware and software platform for wireless sensor networks. Technical report, 6. Fachgespräch Drahtlose Sensornetze der GI/ITG-Fachgruppe Kommunikation und Verteilte Systeme, 2007.
[6] Eclipse Foundation. Eclipse – an open development platform, 2001. `http://www.eclipse.org/`.
[7] Free Software Foundation, Inc. Gnu Compiler Collection (GCC), 1984. `http://gcc.gnu.org/`.
[8] A. Kröller, D. Pfisterer, C. Buschmann, S. P. Fekete, and S. Fischer. Shawn: A new approach to simulating wireless sensor networks. In *Proceedings of the 3rd Symposium on Design, Analysis, and Simulation of Distributed Systems (DASD'05)*, pages 117–124, 2005.