Ad Hoc Networks 17 (2014) 98–113

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

# A reinforcement learning based solution for cognitive network cooperation between co-located, heterogeneous wireless sensor networks

Milos Rovcanin *, Eli De Poorter, Ingrid Moerman, Piet Demeester

*Ghent University – iMinds, Department of Information Technology (INTEC), Gaston Crommenlaan 8, Bus 201, 9050 Ghent, Belgium*

ARTICLE INFO

*Article history:*
Received 19 March 2013
Received in revised form 3 December 2013
Accepted 18 January 2014
Available online 27 January 2014

*Keywords:*
Network cooperation
Machine learning
Reinforcement learning
Least Squares Policy Iteration

ABSTRACT

Due to a drastic increase in the number of wireless communication devices, these devices are forced to interfere or interact with each other. This raises the issue of possible effects this coexistence might have on the performance of these networks. Negative effects are a consequence of contention for network resources (such as free wireless communication frequencies) between different devices, which can be avoided if co-located networks cooperate with each other and share the available resources. This paper presents a self-learning, cognitive cooperation approach for heterogeneous co-located networks. Cooperation is performed by activating or deactivating services such as interference avoidance, packet sharing, various MAC protocols, etc. Activation of a cooperative service might have both positive and negative effects on a network's performance, regarding its high level goals. Such a cooperation approach has to incorporate a reasoning mechanism, centralized or distributed, capable of determining the influence of each symbiotic service on the performance of all the participating sub-networks, taking into consideration their requirements. In this paper, a cooperation method incorporating a machine learning technique, known as the Least Squares Policy Iteration (LSPI), is proposed and discussed as a novel network cooperation paradigm.



## 1. Introduction

As the number of (wireless) communication technologies increases, we are witnessing a drastic increase in the number of co-located, heterogeneous wireless networks with different coverage, data rates, mobility capabilities and requirements. There is a growing need for the network solutions that would efficiently and dynamically support at run-time cooperation between devices from different sub-nets. An illustrative example of two co-existing wireless networks is given in Fig. 1.

One way to support connectivity between co-located devices is to statically (manually) group them into different sub-nets, according to their communication technology. This way, the same network policies can be used for a sub-net, regardless of the characteristics of the devices. Although possible, this approach is usually quite complex [1] and inefficient. Two major drawbacks are:

- Manual configuration is time consuming (computationally expensive).
- It does not take into account dynamically changing network requirements (e.g. changes in network topology).

A direct cooperation between the independent networks could remedy these problems and even significantly improve the network performance, since common resources, such as intermediary nodes for routing, can be shared amongst both networks. Improvements gained by

* Corresponding author. Tel.: +32 9 33 14946.
*E-mail address:* milos.rovcanin@intec.ugent.be (M. Rovcanin).

http://dx.doi.org/10.1016/j.adhoc.2014.01.009
1570-8705/

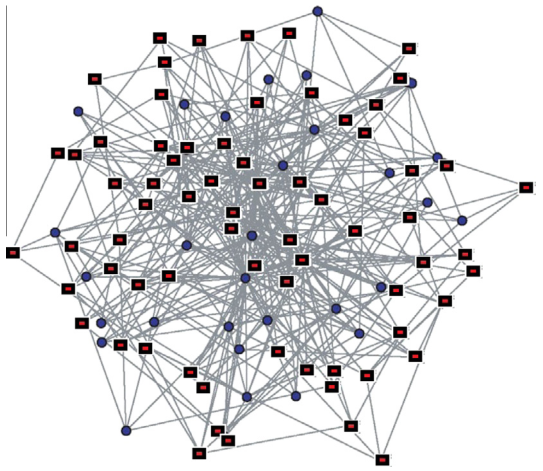

**Fig. 1.** An example of two co-located, heterogeneous, wireless networks, possibly eligible for cooperation. The way nodes are connected (blue dots and red squares) implies that these two networks are unaware of each other at the moment. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

cooperative networking are expected to have an impact on many networking aspects: energy consumption, interference, coverage, electromagnetic exposure, bandwidth allocation, availability, etc. However, the complexity of the configuration problem increases, since management copes with multiple heterogeneous networks, characterized with differing network requirements and capabilities. As an alternative, an independent intelligent entity (i.e. a 'cognitive engine') can be used to initiate and supervise the entire cooperation process [2]. This cognitive entity must be capable of a (1) dynamic optimization and decision making and (2) continuous exchange of collected measurements and environmental states (see Fig. 2).

For example, in [3], a described cooperation paradigm is based on activation of certain symbiotic services, that way influencing an overall network performance. 'Interference avoidance', 'acket sharing' and 'packet aggregation' are just a few of those services. The role of the cognitive engine is to determine the optimal set of services for each participating sub-network, so that the individual performance is improved, taking into account differing sub-net's requirements.


**Fig. 2.** A general scheme of a four stage cognition cycle: (1) gathering information, (2) planning actions, (3) acting, (4) collecting feedback.

In the initial phase of research [3], a linear programming based engine (ILPSolver) [4] was used for the purpose of coordinating the process of service negotiation. In order to calculate the optimal set of services for each sub-net, the ILPSolver requires a significant amount of a priori information about the influence that each service poses on the network requirements (impacts on reliability, delay, network lifetime, etc.). In most cases, this sort of information is extremely difficult to obtain, especially in highly dynamic environments. Possible source can be an existing literature – previously published papers dealing with similar issues or simulations. In any case, the accuracy of the collected predictions in a real case scenario will be questionable.

The approach described in this paper is an improvement of the above mentioned method. Instead of linear programming, a self-learning LSPI-based (Least Square Policy Iteration) [5] algorithm is used as a cognitive engine. LSPI is a form of machine learning [6] [7] that gathers knowledge through a number of trial-and-error episodes. LSPI uses *basis functions*, features from the network, to make an assessment about the influence that different service combination has on each network requirements. Therefore, the methodology proposed in this paper does not require an a priori knowledge about the service influences on the network performance, as opposed to an ILPSolver based approach. In addition, and in contrast to most reinforcement approaches, LSPI does not require fine tuning of the initial parameters such as learning rate.

The remaining of the paper is organized as follows. Section 2 presents related work about network optimizations using self-learning methods. In Section 3, the LSPI fundamentals, mathematical background, convergence and stopping conditions are described in detail. Section 4 introduces the use case LSPI is applied to and the evaluation of its performance. All the aspects of the implementation are thoroughly described in Section 5. Results are analyzed and major issues are identified in Section 6. Some future directions are presented in Section 7. Finally, Section 8 concludes the paper.

## 2. Related work

The following subsections will give an overview of various different approaches in which the reinforcement learning techniques have been used for wireless network optimization purposes.

### *2.1. The use of reinforcement learning as an optimization solution for cognitive radios*

Cognitive radios [8,9] allow devices to autonomously reconfigure transmission parameters based on the state of the environment in which they operate. Reinforcement learning (RL) has been used for solving various optimization problems in cognitive radio networks. For example, the authors of [10] tackled the problem of an efficient spectrum sharing. They defined a "cooperator" as the node that exchanges information with the neighboring nodes, in