Reasoning about Context in Ambient Intelligence Environments: A Report from the Field

Grigoris Antoniou, Constantinos Papatheodorou, Antonis Bikakis

Institute of Computer Science, FORTH and University of Crete, Greece
{antoniou, cpapath, bikakis}@ics.forth.gr

1. Contextual Defeasible Logic

The study of ambient computing (AmI) environments and pervasive computing systems has introduced new research challenges in the field of KR. These are mainly caused by the *imperfect nature of context information*, and the need to provide *distributed reasoning* capabilities. [4] characterizes four types of imperfect context information: unknown, ambiguous (inconsistent), imprecise, and erroneous. These imperfections may be caused by hardware, communication and sensor failures, and the need to integrate information from various sources.

So far, most ambient computing frameworks have followed fully centralized approaches, while others have used blackboard and shared memory paradigms. Collecting the reasoning tasks in a central entity certainly has advantages in terms of control and coordination between. However, such solutions cannot meet the demanding requirements of ambient environments. The dynamics of the network and the unreliable and restricted (by the range of the transmitters) wireless communications call for fully distributed solutions.

In previous work, the authors have presented a novel approach to reasoning about context in ambient intelligence environments, called *Contextual Defeasible Logic* (CDL) [1]. They adopted ideas of and the Multi-Context Systems [3], which consist of a set of contexts and a set of inference rules (known as mapping or bridge rules) that enable information flow between different contexts. These were extended by local nonmonotonic (defeasible) theories, defeasible bridge rules that query other contexts, and the use of trust information about the reliability of information sources [1]. Contextual reasoning proceeds roughly as follows: when a peer P processes a query q, it may query through bridge rules other peers, which in turn may pass on queries to further peers. Based on the information collected, P builds a support set and a blocking set for the query q; these sets contain information about the peers from which (supporting or attacking) information was received. These are compared to each other, based on the trust P places to other peers, and a positive or negative conclusion is drawn.

After these conceptual and formal works were completed, the authors moved on to realize the vision described in those works by implementing CDL on a number of devices, including small devices such as mobile phones, and using these implementations to develop and evaluate sample application scenarios in *real*, not simulated AmI environments. The aim of this paper is to report on the initial findings of this practical work.

2. Architecture and Implementation

In this section, we outline the architecture of the *Mobile CDL Application*, analyze the specifics of our generic implementation, and discuss the choices and associated motivation of the technologies adopted. Figure 1 depicts a layered overview of the system architecture.

Application
Profile and Knowledge Management
Communications
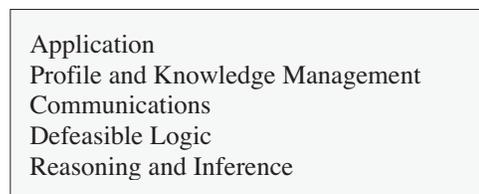Defeasible Logic
Reasoning and Inference

Fig. 1. Layered Architecture Overview

At the bottom is the *Reasoning and Inference layer* where the reasoning is performed. This layer is based on the Java 2 Micro edition (J2ME) programming language because it is reasonably fast, with very good APIs, and is able to run on any mobile device that features a Java Virtual Machine (nowadays the vast majority of cell phones, PDAs, set top boxes etc). Our system relies on Prolog for basic reasoning tasks, and any Prolog system implemented in J2ME can be used. In our current implementation we adopted TuProlog [8], as it is fast and provides a good API for integration.

Moving upwards, the *Defeasible Logic layer* is a Prolog implementation of the algorithms in [1]. Next comes the *Communications layer* which implements the Mobile CDL Service, a protocol for handling incoming or outgoing communication which uses any networking capabilities provided by a given device.

Next up lies the *Profile and Knowledge Management layer*. This is where the User Profile and Knowledge Base are stored and through the communication layer can be also accessed remotely. Finally, on top of all, is the A*pplication layer*, which orchestrates all the underlying component interactions.

The networking capabilities that can be used, where available, to access data stored remotely include: (a) access to the Internet through WiFi, GPRS or 3G, (b) access to other devices using P2P connections based on Bluetooth, and (c) use of GSM cellular network to send and receive SMSs. The above-mentioned communications must adhere to the Mobile CDL Service protocol.

Bluetooth basics. It's a low cost and easy to integrate wireless communication technology, facts that contributed to its widespread use on almost all mobile and non-mobile devices. Its built-in ability to discover other Bluetooth devices in proximity along with their supported services, is very important because thereafter communication can be achieved without the need for an Internet connection or standard IP addresses. Therefore, if any of the discovered devices implement the Mobile CDL service, then we know how to query it, get back the response and finally convert it to knowledge accordingly.

3. Two Application Scenarios

In this section we describe two application scenarios, highlighting different aspects. The first illustrates the integration of information from various devices (contexts), while the second highlights social interaction.

3.1 Scenario 1: Context-Aware Mobile Phone in an Ambient Classroom

The scenario involves a context-aware mobile phone that has been configured by Dr. Amber to decide whether it should ring (in case of incoming SMS) based on his preferences and context. To decide whether it should ring, the phone requires a number of context parameters related to Dr. Amber's current activity. Therefore, it attempts to contact through the wireless network of the university other ambient agents that are located nearby, import from them further context information, and use this information to reach a decision.

Agents involved in this scenario include Dr. Amber's laptop, with access to his calendar, a wireless network localization service providing information on his current location, and a classroom manager, i.e. a stationary computer which has access to other devices in the classroom, including the status of the projector and person detection service.

In the scenario there is a need to resolve conflicting information, as there is evidence both that there is class activity going on (Dr. Amber is located in the classroom at a time when a lecture is scheduled) and the contrary (projector is off and there is only Dr. Amber in the classroom). To resolve this conflict, the mobile phone must be able to evaluate the information it receives from the various sources. For example, in case it is aware that the information derived from the classroom manager is more accurate than the information imported from Dr. Amber's laptop, it will determine that Dr. Amber is not currently giving a lecture, and therefore reach the 'ring' decision.

3.2 Scenario 2: Social Networking

The basic idea behind this type of scenarios can be summarized as follows: *"When I need to know something, first ask an already known specialized source. Alternatively, if I don't know of such a source or I cannot get in touch with it, ask someone around me. In exchange, when I receive new information that may be on interest to others, I should pass it on to them."*

The concrete scenario takes place in Heraklion and involves three students carrying mobile phones with them. Consider that user A is at the University, user B downtown, and user C at the premises of FORTH (all three in different parts of the city).

All three users have their profiles on their mobile phone, share a common interest in the domain of semantic web, and are enrolled in the CS585 class. In addition, B has a public profile online on the university server, A and B have a common interest in tennis and are university friends, as are B and C. Finally, all three have the same preferences for being notified: generally they want to receive information when it is within their areas of interest. However, they do not want to be notified of leisure events when they are scheduled for work at the same time.

Assume that A passes by the University Front Desk, and its Bluetooth server advertises that the lesson of CS585 scheduled to take place later on that day is cancelled. A is notified, and his mobile phone automatically forwards this announcement to A's university friends via SMS. Consequently, B is also notified, and in turn his mobile forwards the information to his university friends. So C, who is a university acquaintance of B, is notified as well.

At the same time, C passes by the ICS-FORTH Lobby and gets informed by its Bluetooth server of a lecture about semantic web, and a tennis tournament both taking place at ICS-FORTH. Again, all users are informed of the lecture, but only B is informed of the tennis tournament: C has no interest in tennis, while A has to work on the same day with an event.

4. Experimental Evaluation

The goal of the experiments was twofold: (a) to test whether the actual computing time for defeasible reasoning on mobile devices is acceptable, and (b) to determine the communication cost for various network types, using realistic test theories in a real environment.

4.1 Scenario 1 Results

This scenario took place at the ICS-FORTH "AmI SandBox" [5]. It is a complex of three rooms equipped with state of the art AmI hardware, made available through a middleware infrastructure. It is called a sandbox as it is the predecessor of a full an intelligent building planned to be finished early 2011.

The time required for communication between the mobile phone and a server was ranging between 80 and 100milliseconds. This was expected because servers continuously compute-update the KB in the background and another thread answers immediately using the shared data structure (KB) so the computation cost of a query is just a read operation. Also, all servers were on same network, and since the message size was up to 10 bytes the actual overhead was only the round trip time. The scenario was also simulated with a cell phone emulator on a laptop and the timings were about 15ms; obviously, cell phones are slower but still within acceptable limits.

The initialization of the reasoning engine and the theory loading required about 600ms, while the query execution was completed in about 150ms. The total time for the scenario was about 800ms since it included some other tasks e.g. measuring and printing timings. These numbers are a rounded average of 10 executions.

For the timings we tested 3 devices: two regular phones and a PDA, all wifi enabled but rather old (2 to 3 years old). The oldest mobile phone was slower about 3 times (on all aspects) than the other cell phone and 5 times slower that the PDA, and thus is considered inadequate. For the cell phone simulator on the laptop the overall time needed was about 200ms.

4.2 Scenario 2 Results

This scenario involves Bluetooth communication between peers of the described network. The time cost for discovery of devices in proximity depends mostly on the number of present devices, and may also vary somewhat depending on hardware. Speed up of discovery is achieved by Bluetooth by pre-caching recently discovered devices.

Apart from the inevitable discovery time, the communication cost is practically better between peers using Bluetooth in contrast to those of the previous scenario that were using wifi and had about the same requirements on data size. This is logical since the data sent is text and Bluetooth communication the way we used it, involves only a direct connection of the two peers, as opposed to at least one more node e.g. a wireless router that is needed for connection to a wifi network. However, wifi is expected to outperform Bluetooth when the data volumes exchanged grows due to its greater bit rate.

For the two back to back queries sent by the Bluetooth server at ICS Lobby the overall time needed was about 5ms for data transmission and around 80ms for the two queries. The Reasoning Engine is initialized during application start-up.

As with the Bluetooth discovery cost, the time needed for SMS communication for this scenario also is not so important and generally varies highly. But once the SMS is received, its processing takes around 15ms and the rest is about the same since the two queries combined are under 160chars long thereby fitting within a single SMS (the case of user B informing A with SMS).

With given an average 5 second SMS delay and about 10 seconds discovery of C's mobile phone, the information from the ICS Lobby is shared with all three users within 21 seconds (about 10085ms for user C, 10ms to send the SMS to user B, and 5095ms receiving and reasoning, 10ms for userB to send to user A an SMS, and 5095 seconds for user A to receive and reason).

5. Conclusion

The paper's findings suggest that simple KR can play an important role in ambient intelligence and pervasive computing: it is rich enough to solve selected problems in these areas, has a formal foundation and semantics (described in earlier works), and is sufficiently efficient to meet the increased requirements in these environments.

Overall, we believe that ambient intelligence and pervasive computing are a rich testbed for KR: it is a rich area with specific requirements in terms of openness, distribution, heterogeneity and efficiency. Thus it can serve as a source of inspiration and advancement, just as the web has done so in the past decade (semantic web).

This work is just one step in an ambitious research plan, and there are concrete ideas on further works. So far, our approach assumes that devices/contexts are always willing to disclose information available to them. In future work, we intend to enrich our approach with a mechanism of access control, to address the key issues of privacy and security in AmI environments. In addition, we intend to study other rich forms of KR in ambient intelligence, including agent coordination to solve problems collaboratively, and reasoning about action. Finally, we intend to broaden the scope of contextual reasoning by allowing different types of peers to work together; in particular, we intend to study the use of recent developments in the area of reasoning about context [2].

References

[1] A. Bikakis, G. Antoniou: Contextual Argumentation in Ambient Intelligence. In *Proc. LPNMR 2009*: 30-43; an extended version of this paper has been accepted for publication in *IEEE Transactions on Knowledge and Data Engineering*.

[2] G. Brewka, T. Eiter: Argumentation Context Systems: A Framework for Abstract Group Argumentation. In *Proc. LPNMR* 2009: 44-57.

[3] F. Giunchiglia, L. Serafini: Multilanguage hierarchical logics, or: how we can do without modal logics. *Artificial Intelligence* 65(1) (1994).

[4] K. Henricksen, J. Indulska: Modelling and Using Imperfect Context Information. In: *Proceedings of PERCOMW* 2004, Washington, DC, USA, pp. 33–37. IEEE Computer Society, Los Alamitos (2004).

[5] C. Stephanidis, A. A. Argyros, D. Grammenos, X. Zabulis: Pervasive Computing @ ICS-FORTH. *Workshop Pervasive Computing @ Home*, International Conference on Pervasive Computing 2008.