Managing Social Constraints on Recharge Behaviour for Robot Companions Using Memory *

Amol Deshmukh, Mei Yii Lim, Michael Kriegel, Ruth Aylett
Heriot-Watt University, Edinburgh, UK
{amol,myl,michael,ruth}@macs.hw.ac.uk

Kyron Du Casse, Koay Kheng L, Kerstin Dautenhahn
University of Hertfordshire, Hatfield, UK
{k.ducasse,k.l.koay,k.dautenhahn}@herts.ac.uk

ABSTRACT

In this paper, we present an approach to monitor human activities such as entry, exit and break times of people in a workplace environment. The companion robot then learns the users' presence patterns over a period of time through memory generalisation and plans a suitable time for recharging itself causing less hindrance to human-robot interaction.

Categories and Subject Descriptors

I.2.9 [**Artificial Intelligence**]: Robotics; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms, Human Factors

1. INTRODUCTION

Autonomous mobile robots acting as companions in a social environment should be capable of managing their own energy needs. Generally, autonomous mobile robots draw power from batteries attached to their bodies in order to operate various sensors, actuators and perform tasks. Batteries have a limited power life which constrains the operational time of the robots and take a long time to recharge via a power source. Most battery recharging mechanisms for mobile robots limit the robots' movement to a defined space where the robots dock and wait until the recharge is complete[1]. While the recharge behaviour is active, the companion robot may be prevented from performing tasks which may hinder continuous human-robot interactions. Thus, it is important for a social companion robot to take into consideration humans presence when planning to recharge its battery.

In order for robots to act as companions in social environments such as homes and offices, they should be able to manage their own energy needs, and do so in a socially intelligent manner [1]. A companion robot should be able to choose a suitable time to recharge and also learn to adapt its recharge behaviour from previous experiences. In this paper, we present an approach for managing social constraints in regard to human activities for a workplace environment where the robot companion learns users' presence patterns through memory generalisation and tailors its recharge schedule based on these patterns. The paper is organised as follows, Section II describes the scenario, Section III discusses our approach to monitor human presence in the workplace environment, Section IV explains the memory mechanism used to learn users presence patterns, Section V describes the preliminary experiment and illustrates initial results and Section VI summarises our approach and discusses future work in brief.

*This work is supported by European Commission (EC). The authors are solely responsible for the content of this publication and it does not represent the opinion of the EC.
[1]http://www.irobot.com/uk/home_robots_roomba530.cfm

HRI'11, March 6-9, 2011, Lausanne, Switzerland.
ACM 978-1-4503-0561-7/11/03.

2. SCENARIO

The work reported here is carried out as a part of the EU project LIREC (LIving with Robots and IntEractive Companions, www.lirec.eu). The project aims to create interactive, emotionally intelligent companions which are capable of establishing long-term relationships with humans in social environments. The "Spirit of the Building" showcase at the Heriot-Watt University, Edinburgh, aims to produce a social helper robot that can act as a "Team Buddy" – an assistant within a lab inhabited by a group of 5 people who work there – and facilitate long-term relationships with users. The "Team Buddy" would act as a workplace buddy, performing tasks such as carrying the phone/printed materials to the users, giving out reminders, providing a lab tour to visitors, approaching and greeting the users, maintaining a collective memory about user preferences such as lunch breaks, entry/exit time. The robot (Pioneer P3AT with enhanced superstructure) is equipped with a laptop PC, 6 lead acid batteries (12V, 7Ah each) offering an approximate operational time of 6 hours when fully charged (depending on usage) and takes about 8 hours to recharge. Considering that it takes a long time for the robot to recharge its batteries to full capacity, there is an urgent need for a mechanism to decide when is the best time to do so.

3. HUMAN MONITORING

It is important for the companion robot to be aware about the presence of users in its environment. In the existing office setup, a group of 5 users are working on their assigned desks, each of which has a desktop PC attached with a web camera over the PC screen facing the user. Using a program that runs on each PC, we are able to detect if a particular user is present/absent at their desk. The program uses a standard face detection algorithm [2] to detect a face in front of the web camera. Additionally, it monitors users' keyboard and mouse activities. This information is communicated to a central PC which then sends it to the robot. The program utilises only about 2% of the CPU resources on each machine, so it does not impact users' workflow. Moreover, it allows the system to collect users presence information without the need of putting active tags on users.

The communication between the program and the central PC is enabled using SAMGAR [3] which utilises the YARP[2] framework that supports distributed computation and communication between modules. Using face detection along with keyboard and mouse input gives a better prediction of users' presence or absence at their desk. For situations when the user is reading, which may not involve keyboard or mouse activity, information from face detection is useful. We used time-outs to perceive events such as break, entry and exit. The time-outs are basically events triggered when a particular user is not detected for specified time intervals. For example, Break: 20 minutes, Exit: 600 minutes, Entry: User detected after 600 minutes. This user monitoring information is continuously recorded for each user in the memory of the robot over an extended period of time.

4. MEMORY

The robot's memory is biologically-inspired, consisting of short-term and long-term components, each further divided into a semantic memory (knowledge about the world) and an episodic memory (events memory) [4]. The episodic memory records all events taking place in the robot's environment, in particular for this experiment, the users' entry, exit and break activities. As the memory is equipped with human-like memory mechanisms for retrieval and forgetting, the robot has the capability to generalise the information in its memory. It uses the data mining Apriori algorithm [5] to detect patterns in the data. As a result, the robot knows users' general presence patterns and hence adapts its actions to these patterns.

5. PRELIMINARY EXPERIMENT

We carried out a preliminary experiment to test the effectiveness of our approach. For one week the system recorded activities (entry, exit, break) of 5 users in the office. The memory generalisation mechanism learns users' activities and time relationship taking into consideration all recorded data. It then abstracts the patterns that occur for more than 4 times as common events. The graph (Figure 1), illustrates the user activity coverage for each event occurred (entry, exit, break) on Y-axis on a time line on X-axis (time where no events occurred have been omitted from graph). The 3 vertical lines on the graph show the memory generalisation results – entry (the earliest time is chosen): 9, exit (the latest time is chosen): 18 and break: 14 (the most occuring) hours.

Noticeably, the user activity coverage shows that more entry events occurred in the beginning of the day, exits at end of the day and breaks during mid-day, usually when the users left for their lunch. From the results, we can observe that the memory generalisation mechanism was effective in finding out the users activity patterns. The results can aid the robot in determining an appropriate recharge session, for instance, between exit and entry time and also during break time if necessary.

[2]http://eris.liralab.it/yarp/

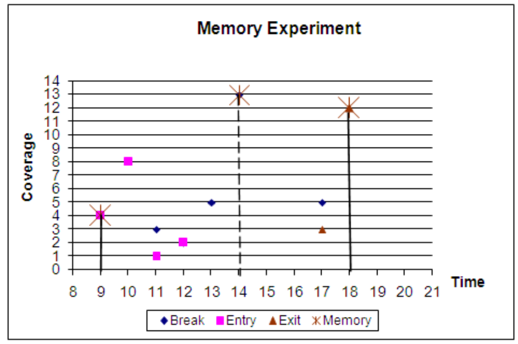

Figure 1: Memory results and user activity graph

6. SUMMARY AND FUTURE WORK

In this paper we present an approach to manage social constraints on recharge behaviour for robot companions, in regard to human activity patterns (entry, exit, break) of people in a workplace environment. It has to be noted that the approach used for human monitoring may not work in a generic way in all social environments, as this can be applied only for users with assigned desks and PCs. Our robot companion learns users' presence patterns through generalisation of information in its memory so that a suitable time to schedule a recharge can be determined. Hence, the memory mechanism enables the companion robot to learn from experiences leading to more adaptive and flexible behaviours. Additionally, a continuous interaction and task performance with the human users can be maintained.

In the future we want to develop an auto-docking recharge mechanism and carry out experiments over an extended period of 6 weeks to measure the number of social errors caused by the robot. An example of a social error could be the non-performance of a task due to occupation while recharging or the battery drying up due to starvation and a human having to plug in the robot for recharging. Besides that, an ontology will be integrated so that the companion can use time concepts such as morning, afternoon, evening and weekends while interacting with humans rather than reciting the exact hour of events.

7. REFERENCES

[1] K. Dautenhahn. Socially intelligent robots: Dimensions of human- robot interaction. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 362(1480):679–704, 2007.

[2] P. Viola and M Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition*, 2001.

[3] K.Du Casse, K.L.Koay, W.C.Ho, and K.Dautenhahn. Reducing the cost of robotics software: Samgar, a generic modular robotic software communication architecture. In *ICAR*, pages 1–6. IEEE, 2009.

[4] M. Y. Lim, R. Aylett, W. C. Ho, P. Vargas, and S. Enz. A socially-aware memory for companion agents. In *Intelligent Virtual Agents (IVA)*, pages 51–62, Amsterdam, 2009. Springer.

[5] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *SIGMOD Conference*, pages 207–216, 1993.