Intel Serv Robotics (2013) 6:1–3
DOI 10.1007/s11370-012-0127-x

EDITORIAL

The need for a research agenda in intelligent robotics

Fulvio Mastrogiovanni · Nak Young Chong

Published online: 18 December 2012

Through the development of a research agenda with the aim of enforcing the integration between artificial intelligence and robotics, researchers are expected to address global concerns about the current lack of agreement around the adoption of standards and common visions related to robot infrastructure, tools for the rapid prototyping of robot systems, software architectures for robots, benchmarking and human-robot interaction. As a matter of fact, this situation is preventing the worldwide adoption of robots as efficient service machines.

A roadmap towards the integration between artificial intelligence and robotics, which takes these aspects into account, must be first envisaged and then put into practice. Such integration poses a number of challenges at the scientific, technical and engineering points of view.

Infrastructure. It is becoming apparent that it is necessary to use huge computational resources to deploy complex robot systems in human-populated environments. The number of everyday events and situations robots are expected to face is virtually unlimited. Powerful and (above all) computationally demanding logic-based knowledge representation and reasoning techniques are considered a promising approach to deal with this issue. This is not surprising, since they allow combinatorial knowledge to be generated and maintained starting from a limited set of represented axioms [1]. However, the computational machinery usually on board robots is not powerful enough to process huge amounts of sensory data, to maintain complex models of the environment and to act accordingly in real-time. This is where infrastructure comes into play. According to the "Cloud Robotics" paradigm [2], robots are entities that are connected to a higher-level intelligent infrastructure, which acts as if it was a sort of world wide web for robots [3]. The challenge is to support the worldwide integration (at the software, hardware and communication levels) of large-scale knowledge representation and reasoning systems, which can be used to drive the behaviour of each robot in the network in the proper way.

Rapid prototyping tools for intelligent robots. If we want robots to operate purposively and sensibly in real-world scenarios, formal top-down approaches in development toolchains to robot behaviour design are needed, which are to a large extent independent from the actual robot hardware. Analogously to robot mechanical and hardware design, the field of robot behavioural design must be enforced as well [4]. In robotics research, this is not a novel claim [5,6]. However, novel exciting issues are emerging.

- The correctness of the software implementation of sensing and control algorithms, which are sound from a theoretical perspective. Of particular interest is the National Science Foundation's initiative about cyber-physical systems (CPS), which emphasizes formal procedures to obtain faithful software implementations replicating their theoretical counterparts [7,8].
- The soundness of the (usually multi-agent) software architecture as a whole, which must integrate different robot behaviours (usually implemented as separate and independent modules) in a full-fledged real-time framework. To this aim, it is necessary to enforce the adoption of formal methods to software design, for instance by

F. Mastrogiovanni (✉)
Department of Informatics, Bioengineering,
Robotics and Systems Engineering,
University of Genoa, Via Opera Pia 13, 16145 Genoa, Italy
e-mail: fulvio.mastrogiovanni@unige.it

N. Y. Chong
The School of Information Science, Japan Advanced Institute of Science and Technology, 1-1 Asahidai, Nomi, Ishikawa 923-1292, Japan
e-mail: nakyoung@jaist.ac.jp

specific requirements [9].

- The need for domain-specific languages and their integration in software architectures. Starting from well-known specialized languages in artificial intelligence domains, such as the planning domain definition language (PDDL) [10], it becomes clear that selected parts of complex robot software architectures may need *ad hoc* and specialized tools for their design. On the one hand, each specialized tool may be endowed with its own formal language to achieve a better design methodology; on the other hand, general-purpose scripting languages like Python [11] and Lua [12] may be used to allow non-technical personnel to easily code robot behaviours.
- The ability, both at the design and development levels, for a framework able to visualize the results of design and implementation choices in a direct way. In spite of the many models so far proposed in Software Engineering [13], development in robotics still follows a "design – develop – integrate – test – refine" approach. However, this approach suffers from well-known drawbacks, for instance the difficulty of integrating software components that are aimed at being completely decoupled. Taking inspiration from novel development paradigms in related fields [14], it is necessary to provide designers and developers with tools able to support an agile full-cycle development process, such as embedded simulations based on real data sets and hardware in the loop [15].

Software frameworks for intelligent robots. During the past few years, different frameworks emerged that can be considered first steps towards the standardization of solutions for developing software applications for robots [16,17]. Such frameworks are important for a number of reasons: on the one hand, they allow for the development of hardware-independent methods and algorithms for data sensing, representation and control; on the other hand, they enforce quality-assurance, long-term stability of code, and the interoperability with components and available tool chains. Based on these promising approaches, it is mandatory to develop software frameworks able to support the correct implementation of theoretically sound algorithms, to work in real-time (in order to allow for a principled integration of computationally demanding reasoning frameworks), to *semantically reproduce* the designed overall robot behaviour and to be deployed in a fully distributed fashion, to benefit from a complex computational infrastructure.

Validation and benchmarking in real-world scenarios. Once complex robot systems are developed, it is necessary to validate their overall behaviour in well-defined scenarios. It is necessary to identify potential scenarios, to derive the corresponding functional and non-functional requirements and to target robot behaviour development towards the fulfilment of such requirements. To this aim, in the past few years, four are the major initiatives to date: the FET Flagship Candidate Robot Companion for Citizens [18], which aims at revolutionising the notion of welfare in Europe, the RoboCup@Home league [19], which specifically targets the home scenario as a framework to develop advanced cognitive robot behaviours, the JST ERATO synergistic intelligence initiative in Japan, which aims at defining robot behaviours on a bio-inspired basis [20], and the Twenty-first Century Frontier Program for Intelligent Robotics Research in South Korea [21]. The challenge is to provide researchers wishing to join this effort with clear benchmarking parameters and complex data sets to produce research outcomes that are comparable to what other researchers are doing.

Human-robot interaction. It is nowadays evident that intelligent behaviours in robots emerge as the interplay among sensory, representation and motor activities in the real-world, specifically when interacting with the surrounding environment or with humans. As a matter of fact, the variability of human life in its own right constitutes a benchmark (i.e., a scenario) for achieving intelligent behaviour. In order for a robot to be credible, it must be able to engage humans not only at the cognitive level, but also at the physical level, specifically avoiding the well-known *uncanny valley effect* [22]. Furthermore, humans require the establishment with artefacts in their environment of long-term and stable relationships [23], and robots constitute no difference. This fact has non-obvious consequences on robot infrastructure, design tools and software architectures. As a matter of fact, long-term human-robot interaction requires huge amount of data to be stored and maintained (thereby requiring infrastructure), meaningful interaction procedures (thereby requiring design tools to be used by non-specialists), and powerful reasoning mechanisms to allow robot to purposively engage humans in interaction activities (thereby requiring real-time software architectures). The challenge is to be able to translate complex human-robot interaction tasks in actual robot design requirements.

Summarizing, it is necessary to call on researchers, practitioners and stakeholders from the industry to participate in the development of a roadmap to fill all the aforementioned gaps in intelligent robotics. The articles present in this Special Issue series are a first step in this direction.

Acknowledgments The Guest Editors would like to acknowledge the journal Editor-in-Chief, the Administrative and Editorial Staff, all the reviewers and, above all, all the authors for their valuable contributions.

References

1. Smolensky P, Legendre G (2006) The harmonic mind. The MIT Press, Cambridge

2. Kuffner J, (2010) Cloud-enabled Robots. Keynote talk at the, (2010) IEEE-RAS International Conference on Humanoid Robots (HUMANOIDS 2010). Nashville, TN, USA, December
3. http://www.roboearth.org
4. http://wiki.blender.org/index.php/Community:Science/Robotics
5. Brooks RA (1991) Intelligence without representation. Artif Intell 47:139–159
6. Arkin R (1998) Behaviour-based robotics. The MIT Press, Cambridge
7. Wolf W (2007) The good news and the bad news. IEEE Comput 40(11):104
8. Derler P, Lee EA, Sangiovanni-Vincentelli AL (2012) Modeling cyber-physical systems. Proc IEEE 100(1):13–28
9. http://www.roboticswikibook.org/
10. McDermott D, Ghallab M, Howe A, Knoblock C, Ram A, Veloso M, Weld D, Wilkins D (2001) PDDL - The planning domain definition language. Technical report CVC TR-98-003/DCS TR-1165.
11. http://pyrorobotics.org/
12. http://www.ros.org/news/2010/08/ros-client-library-for-lua.html
13. Sommerville I (2010) Software engineering, 9th edn. Addison-Wesley, Boston
14. http://worrydream.com/LearnableProgramming/
15. http://www.rosindustrial.org/
16. http://www.ros.org/wiki/
17. http://eris.liralab.it/yarp/
18. http://www.robotcompanions.eu/
19. http://www.robocup.org/robocup-home/
20. http://www.asada.erato.jst.go.jp/
21. http://irobotics.re.kr/eng_index
22. Mori M (1970) The uncanny valley. Energy 7(4):33–35 (originally in Japanese).
23. De Botton A (2006) The architecture of happiness. Penguin Books, USA