Multi-agent Case-based Reasoning Inference Engine Proposal for Reusable Robotics

Francisco Gindre, D. Kotlirevsky, S. Scaine, N. Jafelle Fraga and M. Daniela López De Luise
AIGroup, Universidad de Palermo, Mario Bravo 1050 8^{th} floor, Ciudad Autónoma de Buenos Aires, Argentina

Keywords: Case-based Reasoning, Reusable, Service Robotics, SaaS, Intelligent Agents.

Abstract: Different articles and surveys on the matter identify a remarkable success of robotics on static environments that has not replicated on dynamic areas of application. This paper presents the proposal of a research project that identifies the main hurdles of this issue and present feasible solutions for the matter.

1 INTRODUCTION

The application of robotics as a mean to simplify or outperform different tasks performed by humans has been a matter of research since the 1960's. An exhaustive survey on "The Evolution of Robotics Research" (Armada et al., 2007), reveals many aspects of this discipline that have been in sight of many research teams throughout the last five decades.

The success of applied robotics is widely known in the manufacturing industry (Armada et al., 2007), providing improvements in key areas as productivity, quality assurance and safety. This is a key aspect for today's trends in robotics application as remarked on the mentioned article: "*Currently, the creation of new needs and markets outside the traditional manufacturing robotic market (i.e., cleaning, demining, construction, ship building, agriculture) and the aging world we live in is demanding field and service robots to attend to the new market and to human social needs"* (Armada et al., 2007). Several authors address many of the hurdles that stand between the actual state of the art of service robotics and the demands expressed in Garcia Armada's survey. The following subsections identify three major issues to overcome in the field of reusable robotics.

1.1 The Dynamic Environment Issue

The manufacturing field presents one significant advantage over other possible applications of robotics: a structured, controlled and limited environment. Uncertainty is considered an undesired deviation and therefore is handled by human operators (Armada et al., 2007). Robots working on dynamic environments require the ability of *autonomously* dealing with uncertainty. Many Soft Computing techniques have been applied to meet this requirement such as Bayesian and Neural nets, Expert Systems (Russell and P., 2003); (Amit, 2000); Concept Learning (López De Luise et al., 2010), Genetic Algorithms (Gindre et al., 2010) and Case-Based Reasoning (CBR) (Kolodner, 1992) as problem solving tools.

1.2 Hardware Variability: Reuse Vs. Re-work

The cited survey on robotics evolution reveals a variety of fields in which robotics is being applied and enumerates various examples of architectures used for each one of them. This variety is recognized as limitation in terms of "reusability against re-work" due to incompatibilities between different hardware architectures and programming languages and paradigms. NASA's CLARAty (Nesnas et al., 2006) and the "Player/Stage" (Gerkey et al., 2003) architectures are two approaches towards the mitigation of the re-work issue by adding abstraction layers and standardizing communications between them (with socket APIs).

A study on robotics software reusability (Mallet et al., 2007) identifies two different aspects of that matter: Software Architecture and Control Architecture. The former refers to the usage of software engineering best practices that facilitate

Gindre F., Kotlirevsky D., Scaine S., Jafelle Fraga N. and Daniela López De Luise M..
Multi-agent Case-based Reasoning Inference Engine Proposal for Reusable Robotics.
DOI: 10.5220/0004252204860491
In *Proceedings of the 5th International Conference on Agents and Artificial Intelligence* (ICAART-2013), pages 486-491
ISBN: 978-989-8565-39-6

reuse. The latter refers to the assembly of such components into an executable piece of software that runs on real hardware.

Furthermore, Mallet's research enumerates certain factors that help reuse like: "Avoid all-in-one approaches", "Avoid static design" and usage of frameworks as a mean to software abstraction. It also introduces various examples on *OpenRobots* (LAAS-CNRS, 2012) projects that attempt to reduce re-work and improve reuse specific domains of application.

1.3 Per-agent Knowledge Issue

Machine learning techniques as CBR or Concept Learning combine present and past contexts as feedback to address new situations that must be dealt with. When having many units (agents) deployed on the same or a similar environment, local knowledge bases are generated as a result of using such algorithms. This derives in the local availability of new information of the past experiences of each agent, which generates a "Per-Agent" information island (Cengeloglu et al., 1994). Such information schema is a known *Anti-Pattern* (ISRD Group, 2005) in terms of Software Engineering can be turn into a reusable software *Service-Client* pattern by adding a service-like repository that allows the information to be shared among the deployed agents (Patterson and Fox, 2012).

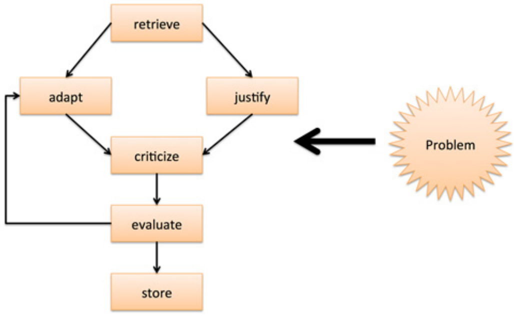

Figure 1a: The CBR Cycle (Kolodner, 1992).

This paper presents the architecture and prototype design of the SIR project (for its name in Spanish "Sistema Inteligente Robotizado"). A Case-Based Reasoning inference engine that helps intelligent robot agents to cope with dynamic environments and their challenges. Designed with a component-based architecture to improve reusability and reduce re-work by adding abstraction layers and using broadly used communication protocols to decouple software from hardware variability and avoid per-agent information islands.

2 THE SIR_CORE CBR-ENGINE PROPOSAL

The SIR project aims to develop a system that provides a centralized inference engine to multiple collaborating service robots regardless their hardware or software capabilities. This centralized inference system (SIR-CORE) aims to deliver a common framework that allows these robots to have the capability of acting upon unknown situations by reusing their own or their collaborators' past experiences. The following subsections explain the selected approaches to address the difficulties shown in sections 1.1, 1.2 and 1.3.

2.1 CBR for Dynamic Environments

Expert or Knowledge Based Systems (KBS) have been widely used to capture and represent the knowledge of an expert of a specific domain and translate it into a representation that can be of use to a computer system that emulates the assertions of the expert. An review of the state of the art of CBR (Watson and Marir, 1994) recollects that the efficiency and accuracy of the KBS is tightly coupled to how effective and exhaustive was the knowledge elicitation process and how well it is implemented and represented as a piece of software. It also identifies the following drawbacks of KBS:

- Knowledge elicitation is a difficult process, often being referred to as the *knowledge elicitation bottleneck*;
- Implementing KBS is a difficult process requiring special skills and often taking many man years;
- once implemented model-based KBS are often slow and are unable to access or manage large volumes of information; and
- they are difficult to maintain

These setbacks represent the opposite requirements for a reusable inference engine. CBR presents itself as a technique to overcome the mentioned setbacks of KBS (Kolodner, 1992).

Firstly, CBR eliminates the *knowledge elicitation bottleneck* by reducing it to the act of gathering the different cases that are presented to a given CBR agent in a particular area of application. Its elicitation process means, providing base cases plus the appropriate means to sense the environment and gather useful information from it to conform a case to the *case-based reasoner*.

Even though CBR systems are initialized with a set of known cases provided by experts on the area of application, the elicitation process continues within the CBR cycle itself (figure 1). When a dynamic environment presents an unknown context to a given CBR agent, this context can be elicited as a new case that the agent will attempt to solve with solution of the most similar case present on it's own case knowledge base.

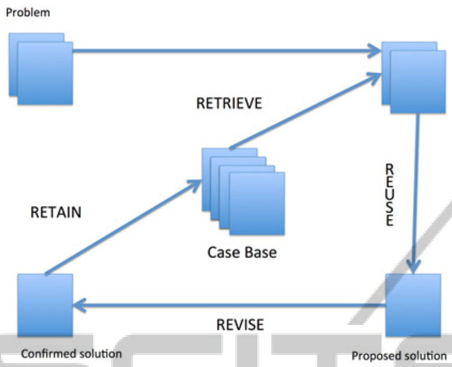

Figure 1b: The CBR Cycle (Aamodt and Plaza, 1994).

Secondly, a single case or a subset of similar cases can be selected without traversing a set of IF-THEN rules. Cases can be selected by a variety of methods depending the data that they encapsulate. This property *enforces reusability*, since the case retrieval can be generalized to the use of a metric that measures the grade of similitude between 2 or more cases. However, there is an underlying setback in it. Querying a vast CBR knowledge base is not effortless. Kolodner identifies it as the *indexing problem*: "is the problem of retrieving applicable cases at appropriate times" (Kolodner, 1992). This problem is accentuated when the number of cases in the KB increases significantly since retrieval performance decays.

Efforts to mitigate this problem include *case labelling (indexing), two-step* ranking that perform an O(n) search (Walker, 2007). A Hierarchical organization of cases with *induction* and *decision Trees* achieved a good case retrieval performance. Nevertheless, according to (Walker, 2007) new case elicitations tended to render the classifier useless.

In terms retrieving cases (Blanzieri and Ricci, 1999) and (Walker, 2007) conclude that Nearest Neighbour Classifiers show superior results opposed to the mentioned techniques. Cunningham's paper (Cunningham, 2009) shows different techniques to measure the distance between two or more cases. Other works show the usage of relational databases as Case KB's by adapting the SQL language to cope with its limitations to query non-relational data (Portinale and Montani, 2008) and the use of data mining techniques as a knowledge discovery aid for CBR KB (Aamodt et al., 1998).

The evidence shows that the *indexing problem* constitutes a scalability limitation for Case Based Reasoning. Since are non-relational data, RDBMS need to be modified in order to be viable Case knowledge bases. The SIR project proposes the use of NoSQL document databases to handle large collections of non-relational information (Padhy et al., 2011).

Thirdly, "*A case-based reasoner solves new problems by adapting solutions that were used to solve old problems"* (Reisbeck and Shank, 1989). This means that "maintenance" of the knowledge base occurs on the CBR cycle intrinsically, as a desired effect instead of as an action to fix the KB itself. Figures 1a and 1b describe how the Adapt-Criticize-Evaluate sub-cycle fits in the CBR cycle proposed by Janet Kolodner. When a CBR agent senses the environment a new case is created. This new case will be matched with the existing cases. Given there is feasible match, the known resolution will be applied to the sensed case. The application of this solution will be criticized and evaluated for its results and adapted if necessary. The result will be stored in the knowledge base as a new elicited case.

In conclusion, because of the mentioned contrasts, the CBR cycle can be applied to many problems (Watson and Marir, 1994). This wide range of application is major requirement in the design of the SIR project and one of the main reasons of using CBR instead of other KBS.

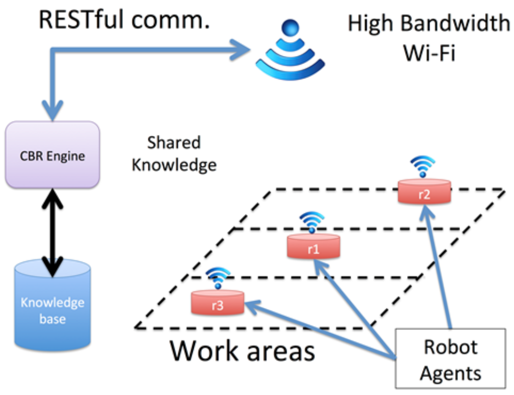

Figure 2: RESTful communications over HTTP.

2.2 RESTfulness for Reuse

Software abstractions are a mean to achieve software reuse. Both CLARAty and Player/Stage focus on abstracting communication layer between components to separate behavior from implementation. In these terms the SIR project focuses in designing a communication layer over a protocol (TCP/IP) that relies on a standard structure to represent the data rather than sending arbitrary bytes transferred over sockets. SIR_CORE is a Web Service (Patterson and Fox, 2012) that relies on REST (Fielding, 2000) over HTTP (Berners Lee et al., 1996) to exchange data and requests with the deployed agents. Service robot agents use the SIR_CORE CBR engine "as a Service" by requesting feedback from a given case sending a GET request to a RESTful URI provided by SIR_CORE to all the agents. RESTfulness provides a standardized communication between clients and servers that avoids the All-in-One approach and provides the necessary interfaces to for each party to implement them as needed in any hardware platform provided that the data interchange adopts the standard protocols and formats. HTTP supports many data interchange formats such as JSON (json.org, 2006) that is a widely used "lightweight, text-based, Language-independent data interchange format" or XML (W3 Consortium, 2012).

This makes possible that human operator request information to the SIR_CORE from a Web application as well.

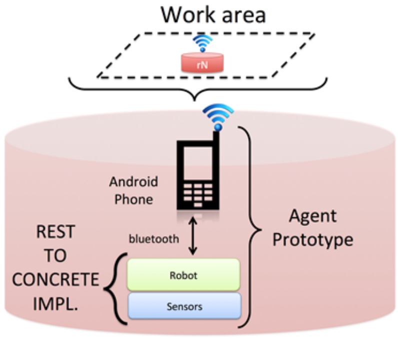

Figure 3: Proposed service robotic agent architecture.

2.3 Shared Knowledge Base

Robot Swarms are presented as a collection of homogenous robots with a simple hardware architecture collaborating towards a common (and collaborative) objective (Mohan and Ponnambalam , 2009). Opposed to that, autonomous standalone robots are have complex architectures (that can be heterogeneous as CLARAty) that implement different mechanisms, like concept learning to "survive" in the environment they are introduced into while they attempt to accomplish a given task. Deploying various robots of the same kind, configured with the same software and KB on an equal dynamic environment, can result in diverse knowledge bases depending on the situations these different agents went through.

In order to avoid information islands, autonomous robots would have to synchronize their knowledge bases either with a central entity or between peers. Although there are frameworks that address knowledge exchange between peers (Cengeloglu et al., 1994), that information would be shared among those agents who are in range to establish a communication and that are mutually compatible. The SIR project proposes a centralized client/server or Software as Service (SaaS) architecture that allows a set of heterogeneous agents to share their knowledge by persisting the sensed cases in the SIR_CORE server.

This design allows cases to flow *transparently* among peers while also contributes to simplify the hardware implementation of each client, since part of the computing needs (the CBR) are outsourced to the SIR_CORE service. Since the communication protocols used are basically the Internet protocols, there is virtually no limitation of distance between these agents between themselves or the SIR_CORE in order to share their knowledge base. The only limitation is how much latency is acceptable for the specific tasks the service robotic agents perform.

3 THE SIR_ROBOT MODULE

The SIR project is designed to let the SIR_CORE deal with the computational complexity of the inference engine, so that robotic agents can be as simple as the problem allows them to be.

A manufacturer adopting the SIR architecture will only be obliged to provide the necessary hardware resources for a piece of software to create and transmit RESTful requests through (wireless) network that reaches a route to SIR_CORE and parse the service's response into executable instructions for the robot agent.

4 PROJECT FEASIBILITY

4.1 Off-the-Shelf Hardware

Figure 3 depicts a retail off-the-shelf SIR_ROBOT implementation. The proposed agent prototype consists of a Lego MindStorms® NXT 2.0 kit (Lego, 2012) with the Open Source LeJOS firmware (Bagnall, 2011) which runs a Java Virtual Machine and an Android 2.3 compatible mobile phone (Google, 2012).

Since the Lego kit is not capable of reaching SIR_CORE through HTTP an additional level of abstraction was added which makes this implementation of SIR_ROBOT a 2-tiered application. The uppermost tier (android phone) provides the means of complying with mandatory requirement of SIR: reaching SIR_CORE over HTTP RESTful requests. Additionally, this tier parses the response and communicates the actions over serialized objects via Bluetooth 2.0 to the LeJOS firmware and vice versa. A Linksys 100Mbit router is being use as access point between the android terminal and the SIR_CORE.

4.2 Wireless Network Capabilities

RESTFul requests add an additional abstraction layer over hardware and network communications to maximize software reuse and tolerance to hardware variability at the expense of additional latency. The feasibility of this architecture in terms of network resources consumption is being estimated on current Wi-Fi standards like IEEE 802.11n (IEEE, 2009) and projected to state of the art Gigabit Wireless estimations (Kneckt et al., 2011). For cases where response times are critical, it is being considered to enhance the SIR_ROBOT with an additional tier that handles certain self-preservation behaviours that do not depend of RESTful requests to the server. (López De Luise et al., 2010).

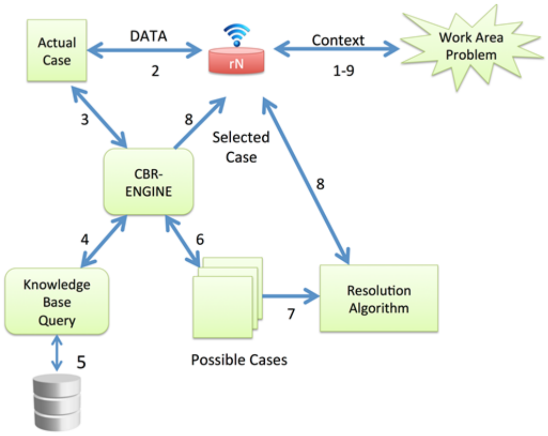

Figure 4: The SIR Cycle.

5 CONCLUSIONS

The objective of the SIR project is to design and implement software architecture that allows n service robotic agents that rely on diverse hardware implementations and capabilities to interact with an inference SaaS platform (SIR_CORE) to perform different tasks.

This paper presents the rationale upon which the SIR project was designed. Along with its strengths and weaknesses and the proposals for enhancing the former and mitigating the latter. Additionally, Figure 4 show how the SIR architecture fits *n* robotic agents into the CBR cycle with minimum hardware requirements by relying in State of The Art SaaS and WebServices communication protocols and existing an network infrastructure.

6 FUTURE WORK

Activities planned for Q3, Q4 2012 and Q1 2013 include the implementation of the SIR_CORE basic architecture, assembly of two distinct SIR_ROBOT implementations to follow basic cases from a unique SIR_CORE Service, latency and stress tests using 802.11b, g, n and public 3G cellular networks.

REFERENCES

Aamodt, A. and Plaza, E. (1994) 'Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches', AI Communications, 39-59.

Aamodt, A., Sandtorv, H. A. and Winnem, O. M. (1998) 'Combining Case Based Reasoning and Data Mining - A way of revealing and reusing RAMS experience ', Safety and Reliability; *Proceedings of ESREL '98* , Balkena, Rotterdam, 1345-1351.

Amit, K. (2000) *Artificial Intelligence and Soft Computing Behavioral and cognitive modeling of the human brain*, 1st edition, CRC Press.

Bagnall, B. (2011) *Intelligence Unleashed: Creating LEGO NXT Robots with Java*, 1st edition, Variant Press.

Berners Lee, T., Fielding, R. and Frystyk, H. (1996) *RFC-1945 Hypertext Transfer Protocol -- HTTP/1.0*, May, [Online], Available: http://tools.ietf.org/html/rfc1945 [14 may 2012].

Blanzieri, E. and Ricci, F. (1999) 'Probability Based Metrics for Nearest Neighbor. Classification and Case-Based Reasoning', ICCBR '99 Proceedings of the Third International Conference on Case-Based Reasoning and Development, London, 14-28.

Cengeloglu, Y., Khajenoori, S. and Linton, D. (1994) 'A Framework for Dynamic Knowledge Exchange Among Intellingent Agents', Symposium Control of the Physical World by Intelligent Agents, New Orleans, LA, USA, 11-18.

Cunningham, P. (2009) 'A Taxonomy of Similarity Mechanisms for Case-Based Reasoning', *Knowledge* *and Data Engineering, IEEE Transactions on*, vol. 21, no. 11, Nov, pp. 1532-1543.

Fielding, R.T. (2000) *Architectural Styles and the Design of Network-based Software Architectures*, Irvine: University Of California, Irvine.

García Armada, E., Jiménez, M. A., González de Santos, P. and Armada, M. (2007) 'The Evolution of Robotics Research: From Industrial Robotics to Field and Service Robotics', *IEEE Robotics and Automation Magazine*, vol. 14, no. 1, March, pp. 90-103.

Gerkey, B. P., Vaughan, R. T. and Howard, A. (2003) 'The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems', In Proceedings of the 11th International Conference on Advanced Robotics, 317-323.

Gindre, F., Trejo Pizzo, D. A., Barrera, G. M. and López De Luise, M. D. (2010) 'A Criterion-based Genetic Algorithm Solution to the Jigsaw Puzzle NP-Complete Problem', Lecture Notes in Engineering and Computer Science , 367-372.

Goodrich, M. and Tamassia, R. (2005) *Data Structures and Algorithms in Java*, 4th edition, Wiley.

Google (2012) *Android Developers*, 14 may, [Online], Available: http://developer.android.com/index.html [14 may 2012].

IEEE (2009) *IEEE Standards Association - 802.11n*, [Online], Available: http://standards.ieee.org/findstds/standard/802.11n-2009.html [14 may 2012].

ISRD Group (2005) *Introduction to Database Management Systems*, Tata McGraw-Hill Education.

json.org (2006) *ietf.org*, july, [Online], Available: http://www.ietf.org/rfc/rfc4627.txt?number=4627 [14 may 2012].

Kneckt, J., Alanen, O., Zheng, C., Huovinen, T. and Nihtila, T. (2011) 'IEEE 802.11ac: Enhancements for very high throughput WLANs', International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC), Helsinki, Finland, 849-853.

Kolodner, J. (1992) 'An introduction to case-based reasoning', *Artificial Intelligence Review*, vol. 6, no. 1, march, pp. 3-34.

López De Luise, M. D., Barrera, G. and Iglesias, J. J. (2010) 'An autonomous robot prototype using Concept Learning model', Lecture Notes in Engineering and Computer Science , Buenos Aires, 547-550.

LAAS-CNRS (2012) *Open Robots*, 13 august, [Online], Available: http://www.openrobots.org/ [13 august 2012].

Lego (2012) *Lego*, 14 may, [Online], Available: http://mindstorms.lego.com/en-us/Default.aspx [14 may 2012].

Mallet, A., Kanehiro, F., Fleury, S. and Herrb, M. (2007) 'Reusable Robotics Software Collection', Proceedings of Second Int. Workshop on Software Development and Integration in Robotics, Roma.

Mohan, Y. and Ponnambalam , S., (2009) 'An Extensive Review of Research in Swarm Robotics', Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on, Coimbatore, India, 140-145.

Nesnas, I. D., Simmons, R., Gaines, D., Kunz, C., Diaz-Calderon, A., Estlin, T., Madison, R., Guineau, J. and McHenry, M. (2006) 'CLARAty: Challenges and Steps Toward Reusable Robotic Software', *International Journal of Advanced Robotic Systems*, vol. 3, no. 1, March, pp. 23-30.

Padhy, Rabi, Prasad, Patra, Ranjan and Satapathy , , (2011) 'RDBMS to NoSQL: Reviewing Some Next-Generation Non-Relational Database's', *International Journal Of Advanced Engineering Sciences And Technologies* , vol. 11, no. 1, September, pp. 15-30.

Patterson, D. and Fox, A. (2012) *Engineering Long-Lasting Software: An Agile Approach Using SaaS and Cloud Computing*, Alpha Edition edition, Strawberry Canyon LLC.

Portinale, L. and Montani, S. (2008) 'A Fuzzy Logic Approach to Case Matching and Retrieval Suitable to SQL Implementation', Tools with Artificial Intelligence, 2008. *ICTAI '08. 20th IEEE International Conference on*, Dayton, 241-245.

Reisbeck, C. K. and Shank, R. C. (1989) *Inside Case-Based Reasoning*, 1st edition, Hillsdale: Lawrence Erlbaum Associates.

Russell, R. and P, N. (2003) *Artificial Intelligence: A Modern Approach*, 2nd edition, Upper Saddle River: Prentice Hall.

W3 Consortium (2012) *W3C*, 24 january, [Online], Available: http://www.w3.org/XML/ [14 may 2012].

Walker, D. (2007) *Similarity Determination And Case Retrieval In An Intelligent Decision Support System For Diabetes Management*, Russ College of Engineering and Technology of Ohio University.

Watson, I. and Marir, F. (1994) 'Case-Based Reasoning: A Review', *The Knowledge Engineering Review*, vol. 9, no. 4, October, pp. 355-381.