Institution: Middlesex University

Unit of Assessment: 11 – Computer Science and Informatics

Title of case study: Model Checking Multi-Agent Systems

1. Summary of the impact

The research work undertaken at Middlesex University on model checking for multi-agent systems has made a significant contribution both to theory and to applications for the verification of complex and critical systems, such as autonomous rovers and avionic scenarios. These scenarios require the verification of properties that go beyond traditional temporal requirements and include epistemic and strategic modalities. Our work has contributed to the development of efficient model checking algorithms and tools that implement state-of-the art features; both the algorithms and the tools have been applied to a number of real-life instances, including scenarios from NASA applications.

2. Underpinning research

The role of model checking in formal system verification has been recognised in the past decade both in industry, where the initial work on hardware model checking lead to the development of tools such as SMV and NuSMV, and in academia, where Clarke, Emerson, and Sifakis shared the 2007 Turing Award for their seminal work on techniques and tools for model checking. The aforementioned body of work is mainly concerned with the verification of *temporal properties*. However, there is a growing interest in developing highly autonomous systems, ranging from autonomous rovers exploring Mars to un-manned autonomous vehicles. For such systems, temporal-only properties are not expressive enough to capture key requirements.

The research work carried out by Dr Franco Raimondi at Middlesex University covers the development of techniques and tools to enable the effective verification via model checking of this new class of requirements. In particular, our work is concerned with the verification of epistemic and strategic modalities, in addition to temporal modalities. The verification of strategic modalities, is crucial for verifying that agents are capable of achieving their goals. More importantly, the verification of strategies is a key element for the *synthesis* of strategies. MCMAS [4] is currently the ***only*** model checker that can synthesize strategies as witnesses for strategic modalities, and the recent work in collaboration with UC Louvain described in [1] provides constructive algorithms for the synthesis of *fair* strategies.

The long collaboration between Middlesex University and NASA Ames, based on annual sabbaticals has resulted in the development of a novel *verification framework* [2]. In this framework, key features of multi-agent system verification have been combined with the software model checker Java Pathfinder. Middlesex has contributed in the past to Java Pathfinder [7]. This new combined framework has enabled the verification of large avionic scenarios, such as the accident of Air France 447 [2] and the Überlingen mid-air collision [6]. In these scenarios it was recognised that temporal-only verification was not sufficient and more expressive specification languages and verification techniques were required, which Middlesex University was able to provide.

Recently, the excellence of the research work at Middlesex University has resulted in the award of a new EPSRC grant to explore the development of algorithms and tools for the verification of *resource-bounded* multi-agent systems (EP/K033905/1). This is an important class of agents that is employed in the specification and verification of a range of applications, from wireless sensor networks to autonomous devices exploring disaster areas such as earthquakes and nuclear plants. This is a joint project with the research group of Dr. Natasha Alechina at Notthingham University.

3. References to the research

This research was based on competitively funded projects, with robust peer review systems. The outcomes from the research were published in leading peer review journals and conferences in the field.

1. Busard, S., et al., *Reasoning about Strategies under Partial Observability and Fairness Constraints*, in *Strategic Reasoning 2013*2013, arXiv preprint arXiv:1303.0793.
2. Hunter, J., et al., *A synergistic and extensible framework for multi-agent system verification*, in *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*2013, International Foundation for Autonomous Agents and Multiagent Systems. p. 869-876.
3. Lomuscio, A., C. Pecheur, and F. Raimondi, *Automatic Verification of Knowledge and Time with NuSMV*, in *IJCAI*2007. p. 1384-1389.
4. Lomuscio, A., H. Qu, and F. Raimondi. *MCMAS: A model checker for the verification of multi-agent systems*. in *Computer Aided Verification*. 2009. Springer.
5. Raimondi, F., C. Pecheur, and G. Brat, *PDVer, a tool to verify PDDL planning domains.* 2009.
6. Rungta, N., et al., *Aviation Safety: Modeling and Analyzing Complex Interactions between Humans and Automated Systems*, in *ATACCS*2013.
7. von Rhein, A., S. Apel, and F. Raimondi, *Introducing binary decision diagrams in the explicit-state verification of Java code*, in *Proc. Java Pathfinder Workshop*2011. p. 82.

Grants:

2004-2013 NASA Ames supported research activities. Total value of approximately $100,000

2013 EPSRC EP/K033905/1, *Verification of resource-bounded multi-agent systems (VRBMAS).* Total grant for all partners £312,280

4. Details of the impact

The future of complex and critical systems in transportation and space exploration provides a challenge for systems engineering. Without the means of diagnosing and checking system correctness, system autonomy of the kind required for remote or long period exploration will be frustrated and the benefits of these new engineering technologies limited. Model checking for multi-agent systems, such as autonomous rovers and avionic control systems, has led to the development of a number of tools and techniques for deployment in current and emerging (new generation) control technologies.

Dr. Raimondi's work has focused on a number of areas with direct impact in aviation flight management and autonomous space systems - namely diagnosability, planning and verification. The initial research work of Dr. Raimondi in collaboration with NASA Ames [7] focussed on the verification of *diagnosability* for autonomous systems using temporal-epistemic logic and a bespoke version of NuSMV, implementing the verification of AR-CTL (a logic to reason about actions and CTL operators) with applications to NASA Livingstone models (http://ti.arc.nasa.gov/tech/rse/vandv/livingstone/). Support for the verification of diagnosability has since then moved to other model checkers, including MCMAS.

In addition to diagnosability, it is usually required to verify that autonomous systems do not violate a certain set of safety requirements, known as *flight rules*. As the behaviour of such systems is often regulated by AI planners, in a number of cases it is necessary to verify that all plans generated by an AI planner satisfy these *flight rules*. A flight rule for an autonomous rover could require that 'no two scientific instruments can be deployed at the same time, with the exception of the panoramic camera if battery levels are above 50%'. Our work detailed in two studies [5] [3] resulted in constructive algorithms for the generation of plan constraints from requirements, and in an tool implementing them. The tool has since then been released as open-source and it remains the only approach to provide verification mechanisms for planning domains.

Following these initial results in the verification of autonomy, we were able to exploit our expertise with Binary Decision Diagrams (BDDs) in model checking multi-agent systems. This resulted in the development of a hybrid explicit/symbolic technique for the verification of Java code that was incorporated into the NASA open-source model checker Java Pathfinder, as part of Google Summer of Code 2011 [7] [S3]. The ideas of this work have since then been adopted outside the area of multi-agent systems and resulted in the development of new approaches for the verification of Software Product Lines at the University of Passau (Germany).

More recently, the work on diagnosability and Java Pathfinder [7] has been at the root of the framework described in [2]. This framework addresses one of the main issues that have prevented the adoption of model checkers in 'real' scenarios. It is impractical and, in most cases, extremely inefficient, to convert the modelling languages currently in use by practitioners into the input languages of model checkers. This is true both for temporal-only model checkers and for model checkers targeted at multi-agent systems. For instance, modelling frameworks such as Brahms (http://www.agentisolutions.com/) are currently employed by NASA to model the interactions between humans and automation in aviation. However, direct translation from Brahms to any model checker would result in the creation of cumbersome and inefficient input code to support inheritance and a number of other Java-like constructs. The research work carried out at Middlesex University, in conjunction with NASA Ames, has lead to a new approach exploiting software model checkers to explore the state space of Brahms models, and then translating this state space to the input language of model checkers such as SPIN, PRISM, and MCMAS. As described in [2, 6], this approach has enabled the verification of complex avionic scenarios, adopting a rich language for requirements, and enabling the analysis of models whose size is orders of magnitude larger than previously possible.

5. Sources to corroborate the impact

Evidence that Dr Raimondi's work has influence an external research community is provided by further joint papers as noted by references. Specifically, Dr Raimondi has had an on-going sabbatical engagement with NASA Ames organised via a third party.

S1. Contract engagement for sabbatical arranged with SGT Inc. on behalf of NASA (2012).

S2. Contract engagement for sabbatical arranged with SGT Inc. on behalf of NASA (2013).

S3. Java Pathfinder extension developed through Google Summer of Code at NASA Ames. http://babelfish.arc.nasa.gov/trac/jpf/wiki/summer-projects/2011-bdd

S4. Java Pathfinder (JPF) extension developed through Google Summer of Code at NASA Ames. Extension to JPF dealt with construction of linear temporal property verification. http://babelfish.arc.nasa.gov/trac/jpf/wiki/summer-projects/start (See 2010 section).