A Simulation Approach to Design Contracts that Govern Emergent Multi-Agent Systems (Extended Abstract)

Maíra Gatti
Carlos Lucena
Dept. of Computer Science
Rio de Janeiro, RJ - Brazil
mgatti@inf.puc-rio.br

Simon Miles
King's College London
Dept. of Computer Science
London, United Kingdom
simon.miles@kcl.ac.uk

Nir Oren
King's College London
Dept. of Computer Science
London, United Kingdom
nir.oren@kcl.ac.uk

Michael Luck
King's College London
Dept. of Computer Science
London, United Kingdom
michael.luck@kcl.ac.uk

ABSTRACT

Agent-based normative systems offer the potential for a business to model, understand the consequences of, and then refine contracts to improve the outcomes for that business. In this paper, we combine a simulation technique designed for investigating and tuning emergent behavior in multi-agent systems with an approach to modeling norms of the complexity found in business contracts. We believe that our approach can aid in the refinement of such contracts by exposing the consequences of contract variations.

Categories and Subject Descriptors

D.2.10 [**Software**]: Software Engineering; I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*multi-agent systems*

General Terms

Design, Measurement

Keywords

Simulation, Norms, Contracts, Multi-Agent Systems

1. INTRODUCTION

Governance for open multi-agent systems can be defined as the set of approaches that aim to establish and enforce some structure, norms or conventions that restrain actions and interactions in order to make the overall system more predictable and effective.

One form of governance is the use of *contracts,* which are documents specifying the expected behavior between agents in the system in a form accessible to the agents. A contract specifies a set of clauses, each of which can be viewed as a normative statement, or *norm* [4]. More generally, norms are used to regulate and coordinate the behaviors of autonomous agents interactions [2][3]. Such regulation is required to mitigate against self-interested behavior, and thus ensure that the agents act to achieve the overall objectives of the multi-agent systems in which they are deployed [3].

Norm-based governance of multi-agent systems has been mainly explored from the specification and monitoring perspectives. However, in a system of any real complexity, it is not easy to specify norms such that the desired emergent macroscopic behavior is achieved. And, as most systems are somewhat nondeterministic, even if the desired system-wide behavior is eventually achieved at some time, the exact evolution is not predictable so we cannot know how quickly or uniformly the desired outcomes appear. In many applications, including the case study used in this paper, it is critical not just to achieve high system utility, but also do so *steadily* over time – businesses which make a substantial profit at one time but fail from then on are not generally considered successful.

By using simulation techniques and a sophisticated model of normative behavior, we can analyze the impact of contracts on the system, and so allow them to be refined for the benefit of a user: increase overall utility and ensure an overall positive trend as the system evolves. We extended De Wolf's [1] approach with the specification of the contracts.

Our technical contribution is an approach to analyzing the effects of different contracts using an iterative simulation approach augmented with contract definitions using a computational normative model. The outcome of a simulation can show flaws in the design. We then are able to analyze the impact of contracts on the system, and so allow them to be refined for the benefit of a user: increase overall utility and ensure an overall positive trend as the system evolves. As a first step, having found a promising solution, it is interesting to return to the simulation and tune norms and parameters until the desired behavior is achieved. The following sections show the application of this approach to the aerospace aftermarket case study.

2. SIMULATION DESIGN

As a case study, we make use of an aerospace scenario modeled as part of the IST-CONTRACT project [3][4]. In this scenario, aircraft operators buy engines for their aircraft from engine manufacturers. In order for the aircraft to keep functioning, the manufacturers must service aircraft engines over time, and provide working engines so that the airline operator's aircraft can be kept flying while other engines are being repaired, this is

Cite as: A Simulation Approach to Design Contracts that Govern Emergent Multi-Agent Systems (Extended Abstract), Maíra Gatti, Simon Miles, Nir Oren, Michael Luck, and Carlos Lucena, *Proc. of 9th Int. Conf. on Autonomous Agent and Multiagent Systems (AAMAS 2010)*, van der Hoek, Kaminka, Lespérance, Luck and Sen (eds.), May, 10–15, 2009, Toronto, Canada, pp. 1541-1542

referred to as *aftercare*. The engine manufacturer is paid by the hour and may face a penalty if planes are on the ground waiting for a working engine. In this business model, timely servicing becomes a key driver of long term profitability between parties.

Aftercare contracts are complex with stipulated service levels and penalties for failure to meet them. Operators want to understand whether the aftercare contracts they agree to will lead to their business goals being met. Specifically, they wish to know that aircraft will be in a state able to fly and that the contractual obligations will not be violated. Moreover, they would like to know whether these aims are both (1) met to a high degree in any period of time, and (2) met steadily over time.

We developed a simulator for aerospace aftercare systems that allows a user to execute the simulations needed in the analysis approach described. At the problem setting: there are airports on which the operator's airlines have employees. There are parts manufacturing sites which are not located at the same site as the airports. Hence, there is a need (and costs) to shipping parts. The engine manufacturers are at some airports, not all. And the edges are the routes and contain the distance information between airports and part manufacturing sites. In this context, there are four classes of contract clause, each clause expressed as a single *norm*, to be analyzed as follows:

N1. Permissions for the airline operator to use particular sites for servicing. The operator may decide to violate this norm if the aircraft is not able to fly to the next site because it requires repair.

N2. Permissions for engine manufacturers to use particular part suppliers. The engine manufacturers may decide to violate this norm when the next two norms are violated.

N3. Prohibitions on the part suppliers to supply parts already used by a list of different operators. The part supplier may decide to violate this norm, when there is no available part to dispatch.

N4. Obligation for the engine manufacturer to deliver serviceable engine in a specified time. The engine manufacturer may have to violate this norm when there is a high demand for engines.

The results we expect from an analysis of emergence are statements that assure the desired evolution of the average system-wide behavior. In our approach, the main guaranteed desired trend is expressible informally as follows:

The system is guaranteed to have a monotonically increasing integration of the difference between the flying rate for aircraft and the costs.

Because there are costs for each violated norm, if we continue maximizing the number of times the aircraft was in the flying state, the costs could reach unacceptable values. Therefore, we need to tradeoff that maximization with the minimization of costs due to contract violations. Hence we aim to maximize the the integral of the normalized difference between those metrics. Moreover, we need to mathematicaly quantify this metric in order to evaluate the results.

3. RESULTS AND CONCLUSIONS

We have performed the simulation for a set of *N=20* trials for each scenario with different contracts. Each trial consists of *n=3000* steps. We ran four trials with eight contracts. For each trial, we ran two different contracts, one for each airline operator: British Airlines and Air France. Then we computed the results regarding the desired guarantees. The main difference between the contracts is the different combination of the same set of norms N2 and N3 to the both airline operators in each trial.

The experiments considering the eight contracts showed us that the contract of Air France in which N3 was inactive gives the best result, while the contract from Air France in which norm N3 was active gives the worst result.

However, the contracts from Air France are not ideal because they allow the engine manufacturers to request parts from any part supplier (norm N2). This is an undesirable situation since some part suppliers do not provide parts of quality. Moreover, reusing parts previously used from different airline operators is also a non-desired situation (which is contract of Air France in which norm N3 was inactive).

Hence, although the contracts from Air France showed us that they are better regarding the desired guarantees, we need to analyze and identify which contract from British Airlines would be the best one. The contracts from British Airline have less permissions and more prohibitions for these norms (N2 and N3, respectively). Even being totally different, the contract from Air France in which the part suppliers are allowed to provide used parts, had almost the same behavior as the contract from British Airlines in which the part suppliers are not allowed to provide used parts.

Overall, our preliminary work suggests designing explicitly specified contracts through iterative simulations is a promising approach.

This work was supported by MCT/CNPq - National Counsel of Technological and Scientific Development, Brazil (Proc. CNPq 201859/2008-4).

4. REFERENCES

[1] DeWolf, T.; Analysing and engineering self-organising emergent applications, Ph.D. Thesis, Department of Computer Science, K.U.Leuven, Leuven, Belgium, May, 2007, 183 pags.

[2] Gatti, M.; Carvalho, G.R. de ; Paes, R. B. ; Lucena, C. J. P. de ; Briot, J. . On Fault Tolerance in Law-Governed Multi-agent Systems. In: Ricardo Choren; Alessandro Garcia; Holger Giese; Ho-fung Leung; Carlos Lucena; Alexander Romanovsky. (Org.). Software Engineering for Multi-Agent Systems V.: Springer Berlin / Heidelberg, 2007, v. 4408, p. 1-20.

[3] Lacroix, B., Mathieu, P., and Kemeny, A.; The Use of Norms Violations to Model Agents Behavioral Variety. In: J. F. Hübner, E. Matson, O. Boissier, and V. Dignum, Eds. LNCS In Artificial Intelligence, vol. 5428. Springer-Verlag, Berlin, Heidelberg, 220-234.

[4] Oren, N.; Panagiotidi, S.; Vazquez-Salceda, J.; Modgil, S.; Luck, M.; and Miles, S.; Towards a formalisation of electronic contracting environments. In Proceedings of COIN@AAAI, pages 61–68, 2008.