Data-Driven Revision of Conditional Norms in Multi-Agent Systems (Extended Abstract)*

Davide Dell'Anna[1] , **Natasha Alechina**[2] , **Fabiano Dalpiaz**[2] and **Mehdi Dastani**[2] and **Brian Logan**[2,3]
[1]Delft University of Technology, Delft, The Netherlands
[2]Utrecht University, Utrecht, The Netherlands
[3]University of Aberdeen, Aberdeen, United Kingdom
d.dellanna@tudelft.nl, {n.alechina, f.dalpiaz, m.dastani, b.logan}@uu.nl

Abstract

In multi-agent systems, norm enforcement is a mechanism for steering the behavior of individual agents in order to achieve desired system-level objectives. Due to the dynamics of multi-agent systems, however, it is hard to design norms that guarantee the achievement of the objectives in every operating context. Also, these objectives may change over time, thereby making previously defined norms ineffective. In this paper, we investigate the use of system execution data to automatically synthesise and revise conditional prohibitions with deadlines, a type of norms aimed at preventing agents from exhibiting certain patterns of behaviors. We propose DDNR (Data-Driven Norm Revision), a data-driven approach to norm revision that synthesises revised norms with respect to a data set of traces describing the behavior of the agents in the system. We evaluate DDNR using a state-of-the-art, off-the-shelf urban traffic simulator. The results show that DDNR synthesises revised norms that are significantly more accurate than the original norms in distinguishing adequate and inadequate behaviors for the achievement of the system-level objectives.

1 Introduction

Multi-agent systems (MASs) comprise autonomous agents that interact in a shared environment [Wooldridge, 2009]. For example, a smart traffic system is a MAS that includes autonomous agents like cars, pedestrians, smart traffic lights, etc. To achieve the system-level objectives of a MAS, the behavior of the autonomous agents should be controlled and coordinated [Bulling and Dastani, 2016]. Norms and their enforcement are one way to do this without limiting the autonomy of the agents in a MAS [Chopra *et al.*, 2018; Tinnemeier *et al.*, 2009; Testerink *et al.*, 2016]. Similar to our society, norms can be viewed as standards of behavior specifying that certain states or sequences of actions in a MAS should occur (*obligations*) or should not occur (*prohibitions*) in order for the objectives of the MAS to be achieved [Boella and van der Torre, 2004].

For many applications, it is assumed that the behavior of the agents and the norms enforced in the MAS will guarantee the achievement of the system's objectives [Dastani *et al.*, 2009]. This is possible, for example, when the MAS designer and agent designers are the same people, or when the agent's behavior is enforced via the regimentation of norms, so that the agents cannot deviate from the intended behavior.

Regimentation, however, is particularly difficult in *open MASs* [Dastani *et al.*, 2004; Artikis and Pitt, 2001], where agents can freely enter or leave the system, and their internal architecture is not known to the MAS designer. Furthermore, without awareness of how the agents internally behave, the MAS designer cannot fully predict how the agents will react to the norms, making it impossible, or computationally infeasible, to design norms that guarantee the satisfaction of the system's objectives in every possible combination of the agents' behaviors. Also, the MAS objectives may change over time, and the designed norms may become outdated and ineffective for the new objectives [Bicchieri, 2005].

To cope with these issues, norms need to be continuously evaluated, and possibly revised when they become inadequate for achieving the MAS objectives [Knobbout *et al.*, 2014; Knobbout *et al.*, 2016; Dell'Anna *et al.*, 2020].

In this paper, we make the following contributions. We introduce *DDNR*, a novel Data-Driven Norm Revision approach. *DDNR* revises norms from data describing the behavior of the agents in a MAS *at run-time*, and does not assume control over agents' design nor requires regimentation. We evaluate the complexity of *DDNR* (see full paper [Dell'Anna *et al.*, 2022a]). We apply and experimentally evaluate a Java implementation of *DDNR*, available online [Dell'Anna *et al.*, 2022b], on an agent-based traffic simulation where norms regulate the behavior (maximum speed and minimum safety distance) of vehicles on a highway.

2 Norm Revision in Normative MASs

As an illustrative example, we use the highway section shown in Figure 1.

We focus on conditional prohibitions with deadlines, which express behavioral properties [Tinnemeier *et al.*, 2009]. A *conditional prohibition* (over a finite propositional

*Full paper in Journal of Artificial Intelligence Research, Volume 75, 2022 [Dell'Anna et al., 2022a]

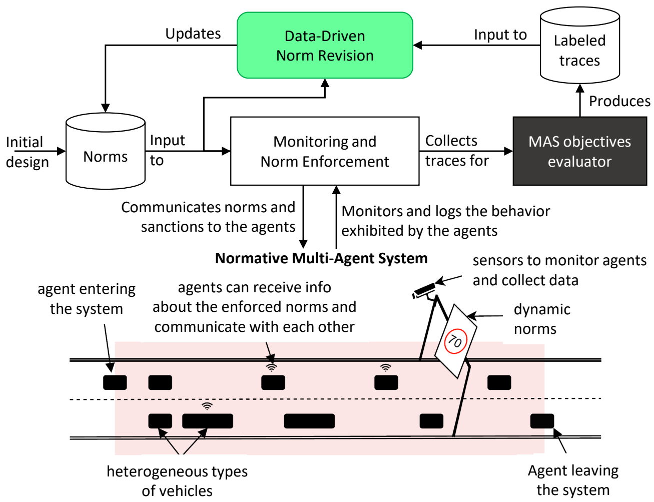


Figure 1: A Normative Multi-Agent System, where norms are used to control the behavior of the autonomous agents (small black rectangles resembling vehicles) in the MAS. The Data-Driven Norm Revision module revises the norms based on the collected data labeled by the MAS objectives evaluator (labeled traces). The MAS objectives evaluator provides a labeling of the monitored agents' behaviors (collected traces) w.r.t. the MAS objectives.

language L) is a tuple (ϕ_C, ϕ_P, ϕ_D), where ϕ_C, ϕ_P and ϕ_D are propositional formulas, expressed in Disjunctive Normal Form (DNF), over L. In this expression, ϕ_C is the (detachment) *condition* of the norm, ϕ_D is the *deadline*, and ϕ_P is a *target state* that is prohibited to occur after a state where the condition of the norm ϕ_C holds, and before a state where the deadline ϕ_D holds (the norm "expires").

Example. The norm *"if a car enters the 2nd km of the highway, it is prohibited from driving faster than 70 km/h until it reaches the 7th km of the highway"* can be represented as a conditional prohibition $(km_2 \wedge car, sp_{70}, km_7)$.

In our framework, a Monitoring and Norm Enforcement component monitors (e.g., via runtime monitoring [Alechina *et al.*, 2014]) agent behavior, and stores the collected data in a database in the form of a *data set of finite traces*. A *trace* in the running example represents the car journey through the highway, and it is generated by the actions of a vehicle. An example of a trace γ composed of 10 states is the following, where km_i indicates that the vehicle reached the i^{th} km of the highway, sp_x indicates that the vehicle's speed is *higher than* a certain speed x in *km/h*, and *car* indicates the vehicle's type.

$$\gamma = (\{km_1, sp_{30}, car\}, \{km_2, sp_{22}, car\}, \ldots, \{km_9, sp_{18}, car\}, \{km_{10}, sp_{14}, car\}) \quad (1)$$

The collected traces are then evaluated by the MAS objectives evaluator component, which labels each trace as either *positive* or *negative* with respect to the MAS objectives. The MAS objectives evaluator is assumed to be an external component, either human or automated, beyond the scope of this paper.

This paper focuses on the Data-Driven Norm Revision module (the green box in Figure 1), which revises the currently enforced norms so to ensure that (i) traces (behaviors) that are labeled as negative by the MAS objectives evaluator are prohibited by the revised norms, and (ii) traces that are labeled as positive by the MAS objectives evaluator are not prohibited by the revised norms. The Data-Driven Norm Revision module is *agnostic to the MAS objectives*, for it only relies on labeled traces provided by the MAS objectives evaluator, which are considered as ground truth. This guarantees that the proposed Data-Driven Norm Revision module is data driven and supports those cases where the MAS objectives do not correspond directly to properties expressible in the language of the norms (e.g., we can express norms concerning the speed of the vehicles, while the objectives may concern CO_2 emissions) or where the causal relationship between norms and objective is not known or unavailable.

Algorithm 1 *DDNR*

Input: data set Γ; set of norms $\mathcal{N}$; list T of $|\mathcal{N}|$ *types* of revisions, one for each norm in $\mathcal{N}$; list V of $|\mathcal{N}|$ triples $\langle V_n^C, V_n^P, V_n^D \rangle$ of propositional variables, one for each norm n in $\mathcal{N}$; number of samples k
Output: ordered list of revised norms $selected$
$\triangleright$ *Synthesis step*
$\mathcal{R}(\mathcal{N}) \leftarrow [\,]$ $\triangleright$ empty list
for all norm $n \in \mathcal{N}$ **do**
    $\mathcal{R}(\mathcal{N})$.APPEND(SYNTHESIS($\Gamma, n, T_n, V_n^C, V_n^P, V_n^D$))
$\triangleright$ *Selection step*
$candidates \leftarrow$ RANDOMSAMPLE($\mathcal{R}(\mathcal{N})$, k)
return $argmax_{cand \,\in\, candidates \,\cup\, \{\mathcal{N}\}}$ ML-ACC($cand, \Gamma$)

3 *DDNR*: Data-Driven Norm Revision

Given a set of norms N and a data set Γ of traces labeled w.r.t. the MAS objectives, *DDNR* (summarized in Algorithm 1) synthesises revised norms that are better aligned with the MAS objectives with respect to Γ.

DDNR consists of two steps: the *synthesis step* and the *selection step*. These are briefly described below for the case of revision of one norm. An in-depth technical description of *DDNR*, also for multiple norms, can be found in the full version of the paper [Dell'Anna *et al.*, 2022a].

3.1 Revising the Norm: the *Synthesis Step*

Consider a norm $n = (\phi_C, \phi_P, \phi_D)$ to be revised. We distinguish three types of revisions of a norm: *alteration*, *weakening*, and *strengthening*.

Alteration – Prohibiting Different Behaviors

An alteration of a norm n is a new norm n' that prohibits a *different* set of behaviors than n. An alteration of n can be realized by making at least one of the components of n, that is the condition, the target state or the deadline, either more or less specific. For example, an alteration of n is a new norm $n' = (\phi'_C, \phi'_P, \phi'_D)$ such that ϕ'_C is less specific than ϕ_C, while ϕ'_P and ϕ'_D are more specific than ϕ_P and ϕ_D, respectively.

Data set Γ				Norm n	Examples of weakening of n: $n_1 = (a \wedge b, c, e)$ $n_2 = (a, c \wedge d, e)$ $n_3 = (a, c, e \vee a)$			Examples of strengthening of n: $n_4 = (a \vee c, c, e)$ $n_5 = (a, c \vee i, e)$ $n_6 = (a, c, e \wedge f)$			Examples of alteration of n: $n_7 = (a \vee c, c \vee j, e \vee b)$
Trace	s_1	s_2	s_3	(a, c, e)	n_1	n_2	n_3	n_4	n_5	n_6	n_7
γ_1	a,b	c,d	e,f	✓	✓	✓		✓	✓	✓	
γ_2	a	c,d	k,l	✓		✓		✓	✓	✓	✓
γ_3	a,b	i,j	e,f						✓		
γ_4	a,b	i.j	k,l						✓		
γ_5	g,h	c,d	e,f					✓			✓
γ_6	g,h	c,d	k,l					✓			✓
γ_7	g,h	i,j	e,f								
γ_8	g,h	i.j	k,l								

Figure 2: Data set Γ composed of 8 finite traces, each made of 3 states (columns s_1-s_3). States are colored based on their types (more details in [Dell'Anna *et al.*, 2022a]). On the right, traces are labeled according to a norm n and to 7 examples of revisions of n obtained from the synthesis step: the cells containing ✓indicate that the corresponding trace is norm-violating, empty cells indicate that the trace is norm-compliant.

Weakening – Prohibiting Fewer Behaviors

A weakening of a norm n is a special case of alteration of n: the new norm n' prohibits a *subset* of the behaviors prohibited by n. To weaken a norm, it is necessary to do at least one of the following operations, each causing n' to prohibit fewer behaviors: (i) making the *condition more specific*, so that the norm is detached in fewer states; (ii) making the *target state more specific*, so that fewer target states are prohibited; or (iii) making the *deadline less specific*, so that the norm "expires" in more states.

Strengthening – Prohibiting More Behaviors

A strengthening of a norm n is another special case of alteration of n, leading to a n' that prohibits a *superset* of the behaviors prohibited by n. To strengthen a norm, it is necessary to do at least one of the following operations, each causing n' to prohibit more behaviors: (i) making the *condition less specific*, so that the norm is detached in more states; (ii) making the *target state less specific*, so that more target states are prohibited; or (iii) making the *deadline more specific*, so that the norm "expires" in fewer states.

Technical details and algorithms for the synthesis of these three types of revisions can be found in the full version of the paper [Dell'Anna *et al.*, 2022a]. Figure 2 provides examples of the types of norms that *DDNR* allows to synthesise from a data set of traces Γ.

3.2 Choosing the New Norm: the *Selection Step*

The *selection step* chooses a new norm from the set $\mathcal{R}(n)$ of candidate revisions of n synthesised during the *synthesis step*.

The relationship between the classification of traces according to a norm (i.e., whether the trace is classified as norm-compliant or norm-violating) and the correct classification of the traces according to the MAS objectives labeling can be described via a confusion matrix. Each cell (i, j) in the matrix contains the number of traces in the data set Γ that are classified as i by the MAS objectives evaluator and as j by the norm.

By analysing a confusion matrix that characterizes the traces in a data set w.r.t. the classification provided by a norm, we can determine the number of classification errors of a norm, the type of errors (i.e., whether negative traces are considered compliant more often than positive traces are considered violating), and we can determine whether a norm is better (aligned with the MAS objectives) than another.

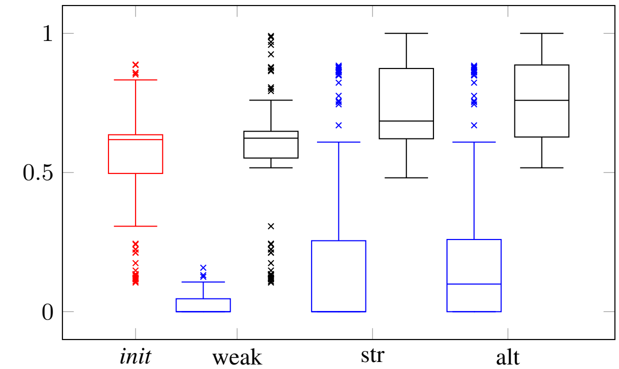
We characterize the concept of *alignment* of a norm with the MAS objectives using the *accuracy* metric, defined as $acc(n, \Gamma) = \frac{TP+TN}{|\Gamma|}$, with *TP* and *TN* being the number of true positives and true negatives obtained with a norm n on the data set of traces Γ.

Therefore, the *selection step* aims at choosing the revision of n with highest accuracy, i.e., given $\mathcal{R}(n)$, we choose, as a revision of n, the norm $n^* = argmax_{n' \in \mathcal{R}(n)}\ acc(n', \Gamma)$.

4 Empirical Evaluation

We make use of a traffic simulation of the highway scenario described in Section 2 to generate a data set of traces describing the behavior of agents in a MAS. We implement the scenario with the SUMO traffic simulator [Krajzewicz *et al.*, 2012], considering two types of vehicles (cars and trucks, each with their own properties such as maximum speed, acceleration, etc.). We use a population of agents randomly and uniformly distributed among *cars* and *trucks*. 75% of the agents are always compliant with the enforced norms, while the remaining 25% will ignore them (can afford the sanction resulting from violation) and focus on maximizing speed.

We execute 100 independent traffic simulations. In each simulation i, we enforce a norm n_i randomly generated. We run every simulation i until 1,500 vehicles drive through the highway section under the enforcement of n_i. Since the behavior of each vehicle corresponds to an execution trace (e.g., trace in Eq. 1), each simulation generates 1,500 traces.



(a)

	Original
M	0.54
SD	0.22
Min	0.10
Q1	0.49
Q2	0.62
Q3	0.64
Max	0.89

(b)

		Δ	*Final*
weakening	M	0.02	0.56
	SD	0.04	0.24
	Min	0	0.10
	Q1	0	0.55
	Q2	0	0.62
	Q3	0.05	0.65
	Max	0.16	0.99
strengthening	M	0.21	0.75
	SD	0.33	0.15
	Min	0	0.48
	Q1	0	0.62
	Q2	0	0.68
	Q3	0.26	0.87
	Max	0.88	1.00
alteration	M	0.24	0.78
	SD	0.32	0.15
	Min	0	0.52
	Q1	0	0.63
	Q2	0.10	0.76
	Q3	0.26	0.89
	Max	0.88	1.00

(c)

Figure 3: Sub-figure (a) shows the original accuracy in red, the accuracy change in blue, and the resulting accuracy after revision in black; sub-figure (b) reports the accuracy of initial norms; sub-figure (c) illustrates the Δ and the resulting accuracy after the revision for with the three types of revision.

Each trace is labeled by the *MAS objectives evaluator* as *positive* if the maximum emission of the vehicle from the beginning to the end of the highway section is below a threshold $t_{co2} = 100$ g/s and the travel time is below a threshold $t_{tt} = 450s$ (the time it takes to drive for 10 *km* at 80 km/h), and *negative* otherwise. Given the resulting labeled data set Γ_i, we execute *DDNR* for the three types of revisions. For each of the 100 norms initially enforced, we obtain 100 weakened norms, 100 strengthened norms and 100 altered norms. We analyze the accuracy of these revised norms in comparison with the 100 original norms.

Figure 3 compares the accuracy of the original norms with that of the revised norms, and shows the accuracy change when performing weakening, strengthening or alteration. All operations led to norms with *significantly* higher accuracy than the original norms. The improvement with weakening has a smaller magnitude, with a negligible effect size $d_{Cohen} = 0.087$, because in the data set obtained from simulation the number of negative traces was low (i.e., the original norms were already too weak and did not need further weakening). In the case of strengthening, instead, the average improvement of accuracy is around 20%, with a large effect size $d_{Cohen} = 1.115$. Finally, with alteration, the average accuracy improves even more, with a large improvement of around 24% ($d_{Cohen} = 1.275$). Note that half of the new norms have an accuracy higher than 76%, and 25% of the norms have an accuracy higher than 89%.

In the full version of this paper [Dell'Anna *et al.*, 2022a], we report on an in-depth analysis of these and additional experimental results. Experiments reported in the full paper also include results for the revision of multiple norms and a study on how well the revised norms generalize to previously unseen traces. Overall, results show that the revised norms are significantly more accurate (aligned with the MAS objectives) than the original norms, exhibiting an average improvement of accuracy on the given data set of traces of about 30% and an average improvement of accuracy of about 13% on unseen traces.

5 Conclusions and Future Work

We investigated the problem of norm revision in contexts where the internals of the agents in a MAS are unknown and where norms are expressed in a different language from that of the MAS objectives that they intend to bring about. We presented results regarding the automated synthesis and revision of conditional norms (prohibitions) with deadlines w.r.t. a set of observed traces representing the behavior of the agents in the MAS. The traces are partitioned into positive and negative ones, depending on whether each helps or hurts MAS objectives. Besides a boolean evaluation, the revision mechanism possesses no information about the relationship between a trace and the objectives. We proposed *DDNR* (Data-Driven Norm Revision): a practical heuristic approach to obtain approximate revisions of the conditional norms. We applied *DDNR* to a traffic simulation. Results show that the revised norms are significantly more accurate (aligned with the MAS objectives) than the original norms. In future work, we intend to embed *DDNR* in a runtime supervision framework presented in earlier work [Dell'Anna *et al.*, 2019; Dell'Anna *et al.*, 2020] that continuously monitors the system's execution and, based on probabilistic strategies, suggests how to revise the norms (i.e., whether to alter, weaken or strengthen them) to continuously guarantee the achievement of the MAS objectives. Considering the relative importance of different states and propositions in revising norms is another future direction of our work. Similarly, we also intend to extend the proposal to support multiple system objectives, and to more fine-grained evaluations (as opposed to the considered boolean evaluation) of the traces with respect to the objectives.

References

[Alechina *et al.*, 2014] Natasha Alechina, Mehdi Dastani, and Brian Logan. Norm approximation for imperfect monitors. In *Proceedings of the 13th International conference on Autonomous Agents and Multi-Agent Systems, AAMAS 2014*, pages 117–124, 2014.

[Artikis and Pitt, 2001] Alexander Artikis and Jeremy Pitt. A formal model of open agent societies. In *Proceedings of the Fifth International Conference on Autonomous Agents, AGENTS 2001*, pages 192–193, 2001.

[Bicchieri, 2005] Cristina Bicchieri. *The grammar of society: The nature and dynamics of social norms*. Cambridge University Press, 2005.

[Boella and van der Torre, 2004] Guido Boella and Leendert W. N. van der Torre. Regulative and constitutive norms in normative multiagent systems. In *Proceedings of the Ninth International Conference on Principles of Knowledge Representation and Reasoning, KR 2004*, pages 255–266, 2004.

[Bulling and Dastani, 2016] Nils Bulling and Mehdi Dastani. Norm-based mechanism design. *Artificial Intelligence*, 239:97–142, 2016.

[Chopra *et al.*, 2018] Amit Chopra, Leendart van der Torre, Harko Verhagen, and Serena Villata, editors. *Handbook of Multiagent Systems*. College Publications, London, 2018.

[Dastani *et al.*, 2004] Mehdi Dastani, Frank Dignum, and John-Jules Meyer. Autonomy and agent deliberation. In Matthias Nickles, Michael Rovatsos, and Gerhard Weiss, editors, *Agents and Computational Autonomy*, pages 114–127, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[Dastani *et al.*, 2009] Mehdi Dastani, Davide Grossi, John-Jules Ch. Meyer, and Nick A. M. Tinnemeier. Normative multi-agent programs and their logics. In *Normative Multi-Agent Systems, 15.03. - 20.03.2009*, volume 09121 of *Dagstuhl Seminar Proceedings*. Schloss Dagstuhl, 2009.

[Dell'Anna *et al.*, 2019] Davide Dell'Anna, Mehdi Dastani, and Fabiano Dalpiaz. Runtime revision of norms and sanctions based on agent preferences. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2019*, pages 1609–1617, 2019.

[Dell'Anna *et al.*, 2022a] Davide Dell'Anna, Natasha Alechina, Fabiano Dalpiaz, Mehdi Dastani, and Brian Logan. Data-driven revision of conditional norms in multi-agent systems. *J. Artif. Intell. Res.*, 75:1549–1593, 2022.

[Dell'Anna *et al.*, 2022b] Davide Dell'Anna, Natasha Alechina, Fabiano Dalpiaz, Mehdi Dastani, and Brian Logan. Supplementary Material for "Data-Driven Revision of Conditional Norms in Multi-Agent Systems". Zenodo, Jan 2022. https://doi.org/10.5281/zenodo.5907522.

[Dell'Anna *et al.*, 2020] Davide Dell'Anna, Mehdi Dastani, and Fabiano Dalpiaz. Runtime revision of sanctions in normative multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 34(2):1–54, 2020.

[Knobbout *et al.*, 2014] Max Knobbout, Mehdi Dastani, and John-Jules Ch. Meyer. Reasoning about dynamic normative systems. In *Proc. of JELIA*, pages 628–636, 2014.

[Knobbout *et al.*, 2016] Max Knobbout, Mehdi Dastani, and John-Jules Ch. Meyer. A dynamic logic of norm change. In *Proceedings of the 22nd European Conference on Artificial Intelligence, ECAI 2016*, volume 285, pages 886–894, 2016.

[Krajzewicz *et al.*, 2012] Daniel Krajzewicz, Jakob Erdmann, Michael Behrisch, and Laura Bieker. Recent development and applications of SUMO - Simulation of Urban MObility. *International Journal On Advances in Systems and Measurements*, 5(3&4):128–138, December 2012.

[Testerink *et al.*, 2016] Bas Testerink, Mehdi Dastani, and Nils Bulling. Distributed controllers for norm enforcement. In *Proceedings of the 22nd European Conference on Artificial Intelligence, ECAI 2016*, volume 285, pages 751–759, 2016.

[Tinnemeier *et al.*, 2009] Nick A. M. Tinnemeier, Mehdi Dastani, John-Jules Ch. Meyer, and Leendert W. N. van der Torre. Programming normative artifacts with declarative obligations and prohibitions. In *Proceedings of the 2009 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT 2009*, pages 145–152, 2009.

[Wooldridge, 2009] Michael J. Wooldridge. *An Introduction to MultiAgent Systems (2. ed.)*. Wiley, 2009.