Runtime Norm Revision using Bayesian Networks

Davide Dell'Anna, Mehdi Dastani and Fabiano Dalpiaz

Deptartment of Information and Computing Sciences
Utrecht University

PRIMA 2018

October 31, 2018

Normative Multiagent Systems

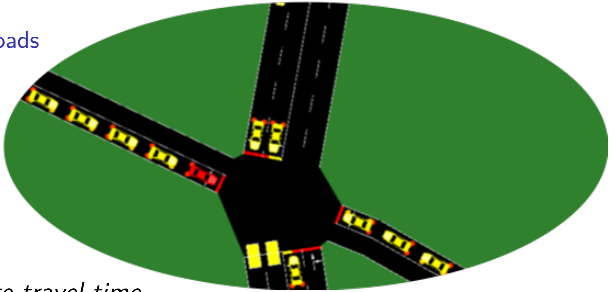
Norms: means to control and influence the behaviour of autonomous agents to guarantee the overall objective of multiagent systems.

Problem: autonomous agents operate in dynamic and uncertain environments s.t. the enforced norms may become ineffective.

Misalignment between norms and overall system objectives at runtime.

Illustrative Scenario

Autonomous Cars on Smart Roads



- System **Objectives**:
  - *minimise the average travel time*.
  - *minimise the number of accidents*.
- Enforced **Norms**:
  - $\mathbf{N}_1$: cars should follow static/adaptive navigation system.
  - $\mathbf{N}_2$: junctions should use static/adaptive traffic lights or line panels.
- Execution **Context**: extreme/normal weather and day/night time.

  $\Rightarrow$ Norm $\mathbf{N}_2$ may not be appropriate for extreme weather.

Research question

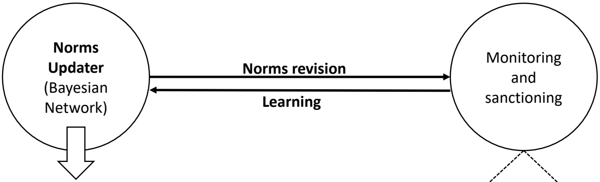
How to design and develop a **runtime supervision framework** that

1. learns at runtime the **effectiveness of the enforced norms**
2. **automatically revises** them, when necessary

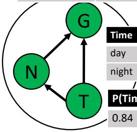
to ensure the overall objectives of a MAS?

Supervising a Normative MAS

An high-level view of the solution

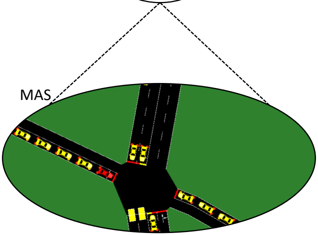


Time	N	P(G = true)	P(G = false)
day	obeyed	0.98	0.02
day	violated	0.8	0.2
night	obeyed	0.08	0.92
night	violated	0.07	0.93

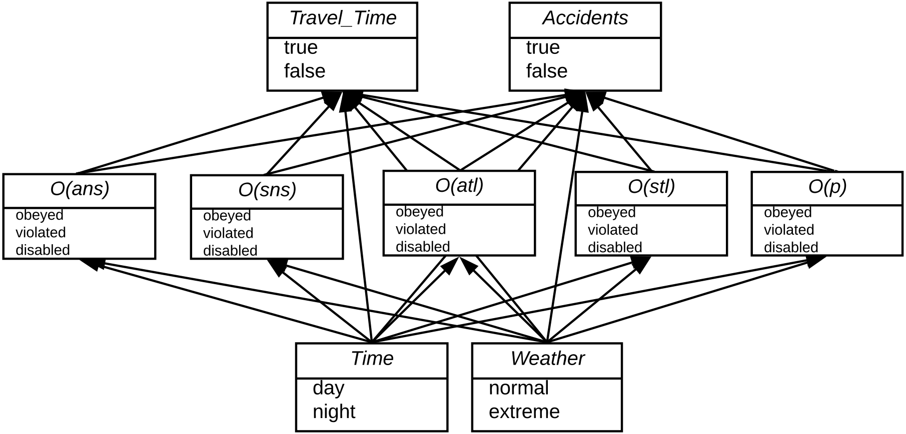


Time	P(N = obeyed)	P(N = violated)
day	0.95	0.04
night	0.85	0.14

P(Time = day)	P(Time = night)
0.84	0.16



Bayesian Network with Norms



Revision Trigger for Norms in mpc

- Norm revision is **triggered** when

  1. changes in the probability distributions in the Bayesian network are not significant anymore, and

  2. the objectives of the system is still not achieved, e.g.,

$$P(Travel\_Time_{true} \wedge Accidents_{true}) \geq 0.95$$

- Norms in the most problematic context **mpc** are subject to revision.

$$\mathbf{mpc} = argmax_{\mathbf{c} \in all(\mathbf{c})} P(\mathbf{O}_{false} \mid \mathbf{c})$$

Revision Procedure: Harmful norms should be disabled

For the enforced norm set **N** and the system objectives **O**, identify norms in **N** that are harmful for the achievement of **O**.

$$argmax_{\mathbf{n} \in \mathbf{n}^{\mathbf{N}}_{\{dis, \neg dis\}}} P(\mathbf{O}_{true} \mid \mathbf{n} \wedge \mathbf{mpc})$$

All norms in assignment n with value *dis* should be disabled.

Note that some of these norms may already have been disabled in which case they remain disabled.

$\mathbf{N}'_a$ is now the subset of **N** that are not harmful and should be active.

Revision Procedure: Norm violations good for Objectives

For the active norm set $\mathbf{N}'_a$ and the system objectives $\mathbf{O}$, identify norms in $\mathbf{N}'_a$ that can contribute to the achievement of $\mathbf{O}$ when violated.

$$argmax_{\mathbf{n} \in \mathbf{n}^{\mathbf{N}'_a}_{\{ob, viol\}}} P(\mathbf{O}_{true} \mid \mathbf{n} \wedge \mathbf{mpc} \wedge \mathbf{n}^{\mathbf{N}'_d}_{dis})$$

All norms in assignment n with value *viol* should be relaxed.

Revision Procedure: Most Likely Explanation for non-achievement of Objectives

For the active norm set $\mathbf{N}'_a$ and the system objectives $\mathbf{O}$, identify the most likely explanation for the non-achievement of $\mathbf{O}$.

$$argmax_{\mathbf{n} \in \mathbf{n}^{\mathbf{N}'_a}_{\{ob, viol\}}} P(\mathbf{n} \mid \mathbf{O}_{false} \wedge \mathbf{mpc} \wedge \mathbf{n}^{\mathbf{N}'_d}_{dis})$$

All norms in assignment n with value *viol* should be strengthen or altered.

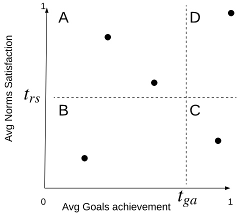
All norms in assignment n with value *Ob* should not be revised or strengthened.

Algorithm 1: Using Bayesian Network

1. Identify and disable harmful norms
2. Identify and relax norms that are useful when violated
3. Identify and strengthen/alter norms whose violations are harmful
4. Keep all other norms unrevised or strengthen them

Algorithm 2: Using Avg. Satisfaction of Objectives & Norms

1. Calculate avg norms satisfaction
2. Calculate avg objectives achievement



3. Section A: Relax norms that are obeyed but better violated, if any. Otherwise, Strengthen/Alter them.
4. Section B: Strengthen/Alter norms that are violated but better obeyed, and Relax norms that are better when violated.
5. Section C: Relax violated norms that are better violated, if any. Otherwise, Strengthen/Alter violated norms that are better obeyed.

Evaluation: The Space of Possible Configurations

We start various algorithms (including our two) in 84 configurations (possible norm sets in each context; 12 norms and 4 contexts) with different average probability of objectives achievement.

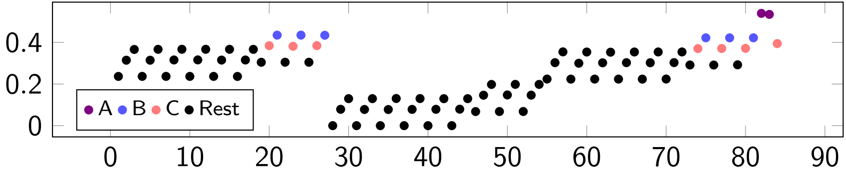


Figure: Avg. probability of objectives achievement for the 84 tried configurations.

Evaluation: Running 5 Algorithms with 3 Thresholds

	T1 ($t_{ga} = 0.5$, 2 optimal conf.)			**T2** ($t_{ga} = 0.4$, 8 optimal conf.)			**T3** ($t_{ga} = 0.37$, 15 optimal conf.)		
Algo	**Steps (σ)**	**Final (σ)**	**Opt.**	**Steps (σ)**	**Final (σ)**	**Opt.**	**Steps (σ)**	**Final (σ)**	**Opt.**
MD4	68.94 (21.47)	0.54 (0.00)	100%	24.48 (12.56)	0.45 (0.04)	100%	17.54 (9.56)	0.43 (0.05)	100%
ML10	88.54 (15.15)	0.54 (0.00)	100%	18.80 (10.62)	0.43 (0.02)	100%	11.15 (6.15)	0.40 (0.03)	100%
ML20	73.81 (17.93)	0.54 (0.00)	100%	23.80 (11.44)	0.44 (0.03)	100%	17.94 (9.08)	0.42 (0.04)	100%
HCPB	**64.86 (27.48)**	**0.54 (0.00)**	**100%**	**11.90 (8.04)**	**0.43 (0.02)**	**100%**	**2.99 (3.03)**	**0.40 (0.03)**	**100%**
HCSB	**79.70 (22.05)**	**0.54 (0.00)**	**100%**	**5.10 (3.50)**	**0.43 (0.02)**	**100%**	**0.82 (0.39)**	**0.40 (0.03)**	**100%**

Table: Comparison of the algorithms with thresholds T1, T2 and T3. Values of Steps and Final columns are average values over the 84 different simulations.

Evaluation: Comparison of Algorithms

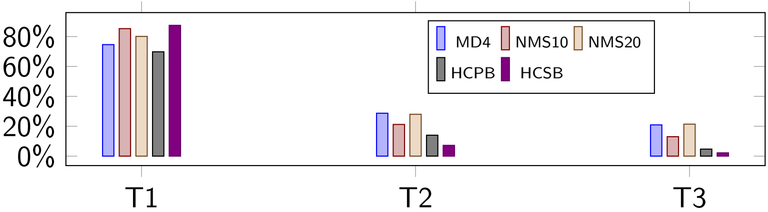


Figure: Average percentage of explored configurations before finding an optimal one. 100% means trying, on average, all the 84 configurations.

Current and Future work

- Runtime **norm-based mechanism design**
- Integration of **sanctions revision**
- Evaluation on case studies involving **rational agents**
- **Bayesian Networks** vs other probabilistic approaches
- "On-demand" **norm synthesis** (?!)