Perspectives on Regulation Adaptation in Multi-Agent Systems: from Agent to Organization Centric and Beyond

Elena Yan[1], Luis G. Nardin[1], Jomi F. Hübner[2], Olivier Boissier[1], Jaime S. Sichman[3]

[1]Mines Saint-Etienne, Univ Clermont Auvergne, INP Clermont Auvergne, CNRS, UMR 6158 LIMOS, F-42023 Saint-Etienne France

[2]Department of Automation and Systems Engineering, Federal University of Santa Catarina, PO Box 476, Florianópolis (SC), 88040-900, Brazil

[3]Laboratório de Técnicas Inteligentes, Escola Politécnica, Universidade de São Paulo, São Paulo, Brazil

{elena.yan, gnardin, olivier.boissier}@emse.fr, jomi.hubner@ufsc.br, jaime.sichman@usp.br

Abstract. *In Multi-Agent Systems (MAS), the regulation of agents aims to define a balance between the control of the system and the agents' autonomy. The ability of a MAS to adapt its regulations at run-time is an important feature that enables it to be flexible to changing situations. There is no unique approach to designing such ability. In this paper, we discuss the different options along the multi-agent oriented programming dimensions, i.e., agent, environment, interaction, and organization. We show that regulation adaptation can be managed within a single dimension or distributed in multiple dimensions. We use a case study in the manufacturing system domain to motivate the regulation adaptation in each of these dimensions.*

1. Introduction

Multi-Agent Systems (MAS) offer promising foundations for modeling complex and decentralized systems composed of interacting autonomous agents. Agent autonomy poses challenges for MAS to achieve their overall objectives. Regulation management systems enable governing agents in MAS toward these objectives, while preserving their autonomy. Yan et al. [Yan et al. 2025b] proposed a unified view for regulation management in MAS, categorizing the design options of regulation management with respect to three perspectives: (i) the regulation capabilities perspective, referring to the procedures, mechanisms, and functionalities for regulation management, (ii) the Multi-Agent Oriented Programming (MAOP) dimensions perspective, referring to the regulation management structured according to the MAOP paradigm [Boissier et al. 2019, Demazeau 1997], and (iii) the architectural perspective, referring to the distribution of the regulation management capabilities among the different MAS components (i.e., agent, environment, interaction, and organization).

In this paper, we focus particularly on the MAOP dimensions perspective proposed in the MAOP paradigm. The MAOP paradigm provides a conceptual foundation for structuring the design and programming of MAS along four dimensions: (i) the Agent dimension that refers to the individual agents with their internal mental state and deliberation; (ii) the Environment dimension that refers to the shared spaces and resources in the environment; (iii) the Interaction dimension that refers to the direct and indirect interactions among all dimensions, and (iv) the Organization dimension that refers to the social structure and coordination of agents in terms of roles, groups, and missions. In [Yan et al. 2025b], it is proposed that regulation management uses abstractions in each of the single MAOP dimensions, originating the views of *agent-centric*, *environment-centric*, *interaction-centric*, *organization-centric* regulation management, and the view of *hybrid-centric* regulation management when combining the use of abstractions from multiple MAOP dimensions. Their analysis of the literature shows that most regulation management frameworks are structured around and use abstractions of a single MAOP dimension, with few proposals aligned with the hybrid-centric view.

Among the regulation management capabilities proposed in [Yan et al. 2025b] (i.e., regiment, enforce, and adapt), this paper focuses on the adapt capability, which governs how regulations evolve at run-time to respond to changing contexts, goals, or system dynamics. The regulation adaptation has been studied sparsely in the literature (e.g., [Campos et al. 2013, Conte et al. 2013, Dell'Anna et al. 2020, Governatori and Rotolo 2010, Hübner et al. 2004]) and its implementation using abstractions of multiple MAOP dimensions remains underexplored. In this paper, we discuss how the adapt capability can be managed using abstractions of single or multiple MAOP dimensions. We thus propose to decompose this capability into sub-capabilities (i) to detect the need to adapt regulations (*detect*), (ii) to design new regulations (*design*), and (iii) to deploy these regulations in MAS (*execute*).

The dynamicity of real-world systems (e.g., manufacturing systems) frequently requires a more nuanced regulation adaptation that spans multiple dimensions, such as automated detection by environment resources, global coordination by the organization, and localized execution by agents. This characteristic motivates the need for run-time regulation adaptation since rigid regulations can lead to delays or inefficiencies, especially in dynamic settings. We illustrate in a skateboard assembly line case study how regulation adaptation operates using abstractions and mechanisms across multiple MAOP dimensions. Different MAOP perspectives provide varying trade-offs. We describe how the adapt sub-capabilities (i.e., detect, design, and execute) can be distributed across a single or multiple MAOP dimensions, in the skateboard assembly line case study.

The paper is structured as follows. Section 2 presents the foundational concepts for understanding regulation management in MAS along the MAOP dimensions. Section 3 presents the perspectives on regulation adaptation, from a single to multiple MAOP dimensions within the skateboard assembly line case study. Finally, Section 4 concludes with a discussion and future work.

2. Background

Here, we introduce the foundational concepts underlying our work. We start by describing the MAOP paradigm (Section 2.1), followed by how regulation management can be realized across multiple MAOP dimensions (Section 2.2). Finally, we present the regulation representations (Section 2.3).

2.1. MAOP Paradigm

Multi-agent systems (MAS) are systems composed of autonomous agents that interact with each other in a shared environment and eventually under one or more organizations [Wooldridge 2009]. The Multi-Agent Oriented Programming (MAOP) paradigm [Boissier et al. 2019, Demazeau 1997] proposes abstractions structured along different dimensions for the purpose of separation of concerns of the representations and mechanisms for their management in MAS (Figure 1), for what concerns:

- the social structure, composed of groups of agents coordinated with roles and responsibilities (The *Organization* dimension [Pynadath et al. 1999]);
- the mental state and the deliberation of agents (The *Agent* dimension [Shoham 1993]);
- the shared space and environment resources made available to agents (The *Environment* dimension [Ricci et al. 2011]); and
- the direct and indirect interactions between the Agent, Environment, and Organization dimensions (The *Interaction* dimension [Huhns 2001]).

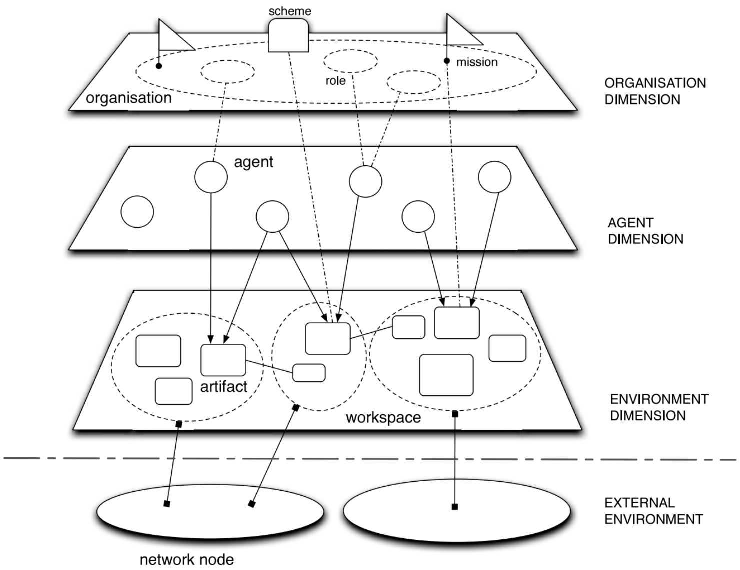

Figure 1. Representation of the MAOP Dimensions from [Boissier et al. 2013], highlighting the Organization, Agent, and Environment dimensions in layers, and the Interaction dimension that refers to the relations and connections between the dimensions.

2.2. Regulation Management along MAOP Dimensions

Yan et al. [Yan et al. 2025b] proposed that, in a global perspective, the capabilities for managing regulations are: (i) the *regiment* capability, which prevents agents from violating the regulations specified in the specifications, (ii) the *enforce* capability, which governs agents' behaviors towards the expected behaviors, and (iii) the *adapt* capability, which enables the adaptation of the regulations that are regimented or enforced to cope with changing situations.

The MAOP Dimensions perspective introduced in [Yan et al. 2025b] is used to structure the regulation representations and regulation capabilities based on the abstractions in each dimension of the MAOP paradigm, originating the organization-centric, agent-centric, environment-centric, and interaction-centric regulation management views. The *agent-centric regulation management* view is when regulations are represented and managed using the abstraction and mechanisms of the Agent dimension. Regulation management is carried out by domain agents using their mental state (e.g., beliefs, goals, and plans) and their deliberation (e.g., [Conte et al. 2013, Yan et al. 2025c]). The agent-centric regulation management view allows agents to manage regulations based on their local perception and participation in the domain problem, but may be restricted by not taking into account the global system state. The *organization-centric regulation management* view is when regulations are represented and managed using the abstraction (e.g., roles, groups) and mechanisms of the Organization dimension (e.g., [Noriega and de Jonge 2016, Boissier et al. 2016, Padget et al. 2016]). The management of regulations is carried out by organization components that strictly depend on the platform (e.g., in [Noriega and de Jonge 2016] are organization agents, in [Boissier et al. 2016] are organization agents and artifacts). The *environment-centric regulation management* view is when regulations are represented and managed using the abstractions (e.g., resources) and mechanisms of the Environment dimension. Regulations can be constrained by the infrastructure or managed by domain resources (e.g., [Piunti et al. 2009, Rodrigues et al. 2003]). The *interaction-centric regulation management* view is when regulations are represented and managed using the abstractions (e.g., messages) and mechanisms of the Interaction dimension. Examples are communication protocols, social commitments, or interaction policies (e.g., [Baldoni et al. 2018, Dastani et al. 2017]).

2.3. Regulation Representations

Regulation management includes regulation representations and regulation capabilities [Yan et al. 2025b]. In the literature, the representations of regulations are categorized in constitutive norms, regulative norms, and sanction rules. *Constitutive norms* define how brute facts are considered from the institutional perspective [Searle 1995]. Constitutive norms produce constitutive facts, which are institutional interpretations of the current state of the world, according to what the constitutive norms specify [de Brito et al. 2019]. *Regulative norms* define actions or states of affairs that are obliged, prohibited, or permitted [Peczenik and Hage 1989]. They regulate the behavior of agents. Regulative norms produce regulative facts, which are the expected behaviors of agents, according to what is specified by the regulative norms. *Sanction rules* define the consequences to be applied to agents in case of compliance with or violation of the associated regulative norms [Gibbs 1966]. Sanction rules may be associated with one or more regulative norms and are triggered by the fulfillment or unfulfillment of the latter [Nardin et al. 2016, Yan et al. 2025c] Sanction rules produce sanction facts, which are the sanction content created for a sanctioned agent.

3. Perspectives on Regulation Adaptation

One challenge in regulation management is the adapt capability. In this section, we discuss the various perspectives on regulation adaptation that can be designed along a single MAOP dimension and in multiple MAOP dimensions (i.e., hybrid-regulation adaptation). We use a case study in the manufacturing system domain to motivate and illustrate the concepts of regulation adaptation.

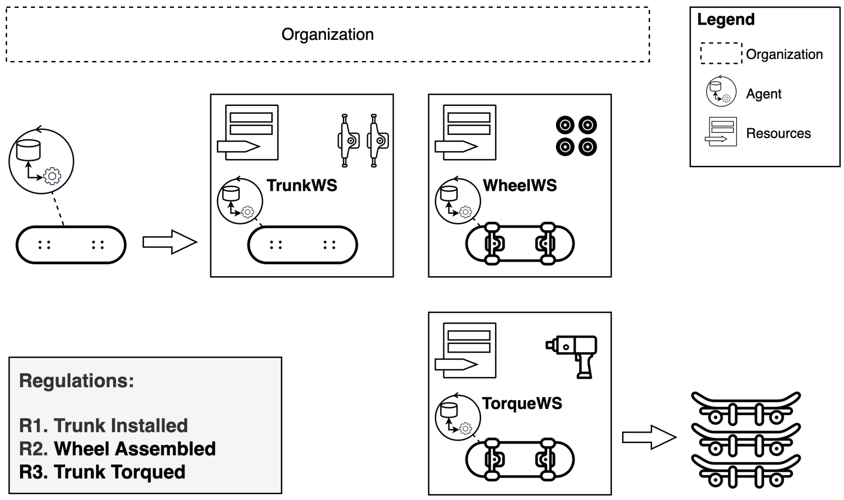

Figure 2. The skateboard assembly line controlled by a multi-agent system.

3.1. Case Study: The Skateboard Assembly Line

Here, we use as a motivation example the Skateboard Assembly Line of the *Fábrica do Futuro*[1] at the University of São Paulo. The assembly line aims to produce custom skateboards according to orders placed by customers. When a skateboard order is received, the assembly line assembles the skateboard by going through three workstations: (i) $TrunkWS$ is the workstation where the trunks are installed on the skateboard, (ii) $WheelWS$ is the workstation where the wheels are assembled according to the colors specified in the order, and (iii) $TorqueWS$ is the workstation where the trunks are torqued according to the customer's preferences (e.g., soft or rigid).

The assembly line is a product-centric manufacturing, i.e., a digital counterpart of a product decides the sequence of tasks to request to the manufacturing services to assemble the product. Figure 2 illustrates a multi-agent system of the skateboard assembly line. The MAS is structured in an organization that defines the roles and responsibilities of agents. Agents play the role of skateboard handler $SkHandler$. When a new order SID for assembling a skateboard is received, the organization assigns to a $SkHandler$ the responsibility to handle the SID according to the customer specification. In the $TrunkWS$, $WheelWS$, and $TorqueWS$ workstations, there are resources that perform the installation of the trunks, assembly of the wheels, and torque of the trunks, respectively.

As a simplified illustrative case study, the regulations define the correct sequence of activities in the workstations. The representations of these regulations are illustrated in Table 1. These regulations define obligations for the agent Ag playing the role of $SkHandler$ and responsible for a SID in the assembly line.

$R1$ specifies that if a skateboard order SID is received, there is an agent Ag playing the role $SkHandler$ to handle the SID, the trunk is not yet installed, and the skateboard is at an available trunk workstation, the agent Ag is obliged to install the trunk to the skateboard. Then $R2$ specifies that if the trunk is installed, the wheels are not yet installed, and the skateboard is at an available wheel workstation, the responsible handler agent Ag is obliged to assemble the wheels to the skateboard. Finally, $R3$ specifies that if

[1]https://sites.usp.br/fabricadofuturo/

Id	Condition	Consequence
$R1$	$order\_received(SID) \wedge$ $play(Ag, SkHandler\langle SID\rangle) \wedge$ $\neg trunk\_installed(SID) \wedge$ $available(TrunkWK) \wedge$ $at(SID, TrunkWK)$	Obligation$(Ag, trunk_installed(SID))$
$R2$	$play(Ag, SkHandler\langle SID\rangle) \wedge$ $trunk\_installed(SID) \wedge$ $\neg wheel\_assembled(SID) \wedge$ $available(WheelWS) \wedge$ $at(SID, WheelWS)$	Obligation$(Ag, wheel_assembled(SID))$
$R3$	$play(Ag, SkHandler\langle SID\rangle) \wedge$ $wheel\_assembled(SID) \wedge$ $\neg trunk\_torqued(SID) \wedge$ $available(TorqueWS) \wedge$ $at(SID, TorqueWS)$	Obligation$(Ag, trunk_torqued(SID))$

Table 1. Representations of the regulative norms in the skateboard assembly line. A regulative norm is represented with an identifier Id, the condition that activates the regulation, and the consequence of the regulation.

the wheels are assembled, the trunk is not yet torqued, and the skateboard is at an available trunk workstation, the responsible handler agent Ag is obliged to torque the trunk to the skateboard.

Note that these regulations assume a classic sequence of tasks and may be too rigid to handle unexpected or evolving conditions in the assembly process. For instance, if the workstation becomes unavailable due to a failure or maintenance, the strict ordering imposed by the regulations may block the progress with an inefficient result. Therefore, to ensure flexibility and resilience, the system requires to adapt these regulations at run-time to cope with evolving changes.

3.2. Adapt Capability

To enable flexibility to the regulation adaptation, we decompose the adapt capability into three sub-capabilities [Yan et al. 2025a]: (i) *detect* potential regulations to be adapted, (ii) *design* regulations adaptations, and (iii) *execute* the designed regulations adaptations.

The *detect* sub-capability is carried out by MAS components responsible for the regulation management that aims to detect potential regulations to be adapted based on the current context. Considering the Skateboard scenario, the detect sub-capability detects that there is an overload of orders, and the sequential constraints imposed by the regulations $R2$ and $R3$ introduce a bottleneck in the workstations $WheelWS$ and $TorqueWS$. Thus, the detect sub-capability identifies $R2$ and $R3$ as potential regulations to be adapted.

The *design* sub-capability is carried out by MAS components responsible for the regulation management that aims to design regulation adaptations with their associated operation (i.e., create, modify, and remove). In the Skateboard scenario, the design sub-capability aims to design the regulation adaptation to the detected regulations from the detect sub-capability according to the context. Table 2 presents an example of a designed regulation adaptation. Specifically, it shows the adaptation of the regulation $R3$, resulting in a new regulation $R3'$, where the condition $wheel\_assembled(SID)$ is replaced by $trunk\_installed(SID)$.

Id	Condition	Consequence
R3′	$play(Ag, SkHandler\langle SID\rangle)\wedge$ ~~wheel_assembled(SID)~~$\wedge$ $\mathbf{trunk\_installed(SID)}\wedge$ $\neg trunk\_torqued(SID)\wedge$ $available(TorqueWS)\wedge$ $at(SID, TorqueWS)$	Obligation$(Ag, trunk_torqued(SID))$

Table 2. Designed regulation adaptation.

The *execute* sub-capability is carried out by MAS components responsible for the regulation management that aims to execute the designed regulation adaptations according to their operations to adapt the regulations' representations. Finally, the execute sub-capability aims to execute the designed regulation adaptation along its operation in the regulation repository. After the execution of the adaptation, the initial set of regulations (Table 1) is updated to $\{R1,\ R2,\ R3'\}$ where $R3'$ is the designed regulation adaptation (Table 2). Thus, the skateboard assembly line can install the trunk and torque in any order.

The regulation adaptation process can continue performing the detect, design, and execute sub-capabilities for further adaptations. For instance, in certain contexts, regulations may need to be adapted for sequential execution of the assembly line or to respond to events such as failures or additions of workstations.

3.3. Regulation Adaptation in a Single MAOP Dimension

To capture the regulation adaptation in a single MAOP dimension, we define X-centric regulation adaptation, where the components to carry out the regulation adaptation are concentrated in one MAOP dimension.

Definition 1 (X-Centric Regulation Adaptation) *The X-centric regulation adaptation denotes a view in the MAOP Dimensions perspective in which the representations and sub-capabilities (i.e., detect, design, and execute) of the adapt capability are made of abstractions and mechanisms of the X dimension of the MAOP paradigm, where $X \in \{Agent, Organization, Environment, Interaction\}$.*

In an *agent-centric regulation adaptation*, each domain agent can autonomously manage the adaptation based on its perception of the environment and its mental state. The detect sub-capability can be localized into agents, allowing agents to quickly recognize issues or inefficiencies that affect their operations. The design sub-capability can be performed based on the individual agent's observations and experiences. The execute sub-capability can also be faster and direct to the agent's own regulation representations. In a pure agent-centric view, without any explicit form of coordination and alignment, agents themselves have to negotiate, align, and learn the regulations. This flexibility may not take into account the global objectives and may lead to inconsistencies or undesirable states in the system. In the skateboard case study, each skateboard agent can decide whether to adapt its regulations, for instance, based on its own observations about workflows or processing delays.

In *organization-centric regulation adaptation*, the organization can exploit its global view of the system's current and historical state to make regulation adaptation decisions that are aligned with organizational objectives. The detect sub-capability can be performed by monitoring the current and historical global state to detect any inefficiency, conflicts, or non-compliance. The design sub-capability can be performed by considering the organization's state and objectives, while the execute sub-capability applies the adapted regulations, propagating the changes to all agents. In the skateboard case study, the organization can perform the detect, design, and execute sub-capabilities using its global view over all the skateboard handler agents and workstations. However, in highly dynamic or large-scale systems, the organization may face numerous regulation adaptation decisions, which potentially lead to bottlenecks and reduced responsiveness.

The *environment-centric regulation adaptation* relies on tools or resources in the environment to perform the detect, design, and execute sub-capabilities. Since these resources are typically non-autonomous, their regulation adaptation mechanisms are often automatic or hardcoded. It is possible to embed in the resources conditions that trigger the detect, design, or execute sub-capabilities. In the skateboard case study, resources operating on the $WheelWS$ and $TorqueWS$ workstations can embed automated mechanisms to detect, design, and execute regulation adaptation.

A pure *interaction-centric regulation adaptation* is more difficult to envision, since the interaction dimension in the MAOP paradigm primarily serves to connect other dimensions. Nonetheless, structured interaction representations and mechanisms can be realized to facilitate regulation adaptation.

3.4. Regulation Adaptation Along Multiple MAOP Dimensions

When the regulation adaptation sub-capabilities are structured in multiple MAOP dimensions, we have the *hybrid-centric regulation adaptation*. By expanding Definition 1, we can define the hybric-centric regulation adaptation where the components to carry out the adapt capability are represented and managed in more than one dimension.

Definition 2 (Hybrid H-Centric Regulation Adaptation) *The Hybrid H-centric regulation adaptation denotes a view in the MAOP Dimensions perspective in which the representations and sub-capabilities (i.e., detect, design, and execute) of the adapt capability are distributed using the abstractions and mechanisms of all the H dimensions of the MAOP paradigm, where $H \subseteq \{Agent, Organization, Environment, Interaction\}$ such that $|H| \geq 2$.*

The sub-capabilities in the adapt capability can be distributed across multiple dimensions to combine the advantages offered by each dimension. Given the various possible combinations of the various dimensions in distributing the adapt sub-capabilities, here we only highlight some examples and discuss their hybrid configurations.

In a *hybrid agent- and interaction-centric regulation adaptation*, interaction mechanisms and agents are used to represent and manage the adapt capabilities. Agents can carry out the regulation representations and detect, design, and execute sub-capabilities. For instance, agents detect the need for adaptation through their perceptions and deliberations. Structured interaction mechanisms for regulation adaptation can be integrated to enable agents to negotiate and coordinate the design of regulation adaptation. Once a consensus is reached, each agent can execute the regulation adaptation. In the skateboard case study, if multiple skateboard handler agents detect delays at $WheelWS$, they can communicate and agree on the design of the adapted regulations. Each agent can then execute its regulation representation accordingly. The hybrid agent- and interaction-centric regulation adaptation enables decentralized and collaborative flexibility, but requires robust coordination to avoid inconsistencies.

In a *hybrid agent- and environment-centric regulation adaptation*, the adapt capabilities are distributed using abstractions in the agent and environment dimensions. For example, in simpler scenarios, environment resources can autonomously detect the need for regulation adaptation using automated mechanisms. Upon detection, the agent can then perform the design and execute sub-capabilities. In the skateboard case study, when resources in the $WheelWS$ have detected the need for regulation adaptation, e.g., a large queue of skateboards waiting for the wheel assembly, it can trigger an alert for adaptation. In response, skateboard handler agents can perform the design and execute sub-capabilities to update the regulations and change their behavior, e.g., performing first in the $TorqueWS$.

Rather than having each skateboard handler agent that manages adaptation individually, the *hybrid organization- and environment-centric regulation adaptation* relies on an organization that performs the regulation adaptation with the support of environmental resources. The detect sub-capability can be delegated to resources in the environment, enabling localized and scalable detection, while the organization handles global design and execution of regulation adaptations. In the skateboard case study, when the resources in $WheelWS$ trigger an alert for regulation adaptation, the organization can design and execute the regulation adaptation and spread these adapted regulations to skateboard handler agents. This ensures consistency and alignment across the system while still offloading detection to distributed environment resources.

Since environment components are typically passive, an alternative is to delegate some of adapt sub-capabilities to agents. This is *hybrid organization- and agent-centric regulation adaptation*. The detect sub-capability can be carried out by agents that can be performed based on their localized view and own experiences. The organization can collect these detected requests and, using its global perspective, design the adaptation. Agents then execute the adaptation, ensuring that their regulations remain aligned with organizational goals. In the skateboard case study, skateboard handler agents that detect inefficiencies can request adaptations. The organization evaluates these requests and, if accepted, designs a suitable regulation adaptation and spreads it to the agents to execute.

Another perspective is the *hybrid agent-, organization-, and environment-centric regulation adaptation*. In this perspective, we can have for instance the detect sub-capability in the environment resources, the design sub-capabilities shared between agents and the organization, and the execute sub-capability performed by agents. This setup supports both localized responsiveness and global coordination, but introduces complexity in consistency and alignment. In the skateboard case study, the resources in $WheelWS$

and $TorqueWS$ can trigger alerts upon detecting overload. The design sub-capability can then be collaboratively carried out by the organization (to maintain global consistency) and the affected skateboard handler agents (to reflect localized needs). Finally, the execute sub-capability can be performed by the agents to update their local regulations.

The interaction dimension can also be integrated in this perspective, generating the *hybrid agent-, organization-, environment-, and interaction-centric regulation adaptation*. In this perspective, environment resources can detect the need for adaptation. Agents and the organization can collaboratively design the regulation adaptation through interaction protocols by considering both agents' local needs and organizational goals. Finally, agents and the organization can execute the regulation adaptation by deploying the changes. This perspective enables flexibility, responsiveness, and alignment in regulation adaptation, but introduces complexity and communication overhead.

Here, we have described some examples of possible combinations for *hybrid-centric regulation adaptation* involving abstractions from multiple MAOP dimensions. Other combinations in the MAOP dimensions and distribution of the adapt sub-capabilities are also possible.

4. Conclusion and Future Work

In this paper, we described different perspectives on regulation adaptation, focusing on how detect, design, and execute sub-capabilities can be represented and managed within a single or multiple MAOP dimensions. We discussed the various perspectives, from the agent, organization, environment, and interaction with the hybrid perspectives in regulation adaptation with their benefits and limitations. The agent-centric regulation adaptation offers fast and localized adaptation, but risks inconsistency. The organization-centric regulation adaptation enables global coordination, but lacks flexibility. The environment-centric regulation adaptation enables the embedded automated responses, but is limited in its decisions. The interaction-centric regulation adaptation enables communication and alignment, but may incur overhead. Finally, the hybrid centric can balance the strengths and limitations of these perspectives, but adds complexity in regulation adaptation. We have discussed these perspectives with an illustrative manufacturing domain case study of a Skateboard assembly line in the *Fábrica do Futuro*. Notice that in this paper, we focus primarily on the MAOP dimension perspective for positioning the adapt capability; however, it is also important to consider the other architectural perspective identified in [Yan et al. 2025b] in designing regulation management mechanisms.

As future work, we plan to formally define the representations and management mechanisms of the regulation adaptation model, enabling the detect, design, and execute sub-capabilities. We also aim to implement these ideas in a MAS platform and validate them using the skateboard case study. In order to realize these perspectives in practice, the MAS platform must support (i) the dynamic adaptation of regulations, and (ii) the representation and management of adaptation across components tied to different MAOP dimensions. Furthermore, once regulation adaptation is achieved from multiple perspectives, we envision that adaptation can be targeted beyond the regulation representations to the regulation management structure, enabling the dynamic shift of the regulation management responsibilities across the various MAOP dimensions.

Acknowledgements

This study is partially funded by ANR-FAPESP NAIMAN project (ANR-22-CE23-0018-01, FAPESP 2022/03454-1). The authors are also members of the UNBIAS team, which is a component of the THUS pillar of the USP-CNRS International Research Center.

References

Baldoni, M., Baroglio, C., Capuzzimati, F., and Micalizio, R. (2018). Commitment-based agent interaction in JaCaMo+. *Fundam. Informaticae*, 159(1-2):1–33.

Boissier, O., Bordini, R. H., Hübner, J. F., and Ricci, A. (2019). Dimensions in programming multi-agent systems. *The Knowledge Engineering Review*, 34:e2.

Boissier, O., Bordini, R. H., Hübner, J. F., Ricci, A., and Santi, A. (2013). Multi-agent oriented programming with JaCaMo. *Science of Computer Programming*, 78(6):747–761.

Boissier, O., Hübner, J. F., and Ricci, A. (2016). *The JaCaMo Framework*, pages 125–151. Springer International Publishing, Cham.

Campos, J., López-Sánchez, M., Salamó, M., Avila, P., and Rodríguez-Aguilar, J. A. (2013). Robust regulation adaptation in multi-agent systems. *ACM Trans. Auton. Adapt. Syst.*, 8(3):13:1–13:27.

Conte, R., Andrighetto, G., and Campennl, M. (2013). *Minding Norms: Mechanisms and dynamics of social order in agent societies*. Oxford University Press.

Dastani, M., van der Torre, L. W. N., and Yorke-Smith, N. (2017). Commitments and interaction norms in organisations. *Autonomous Agents and Multi-Agent Systems*, 31(2):207–249.

de Brito, M., Hübner, J. F., and Boissier, O. (2019). Coupling the normative regulation with the constitutive state management in situated artificial institutions. *Knowledge Engineering Review*, 34:e21.

Dell'Anna, D., Dastani, M., and Dalpiaz, F. (2020). Runtime revision of sanctions in normative multi-agent systems. *Autonomous Agents Multi Agent Systems*, 34(2):43.

Demazeau, Y. (1997). Steps towards multi-agent oriented programming. In *First International Workshop on Multi Agent Systems, Boston, Mass.*

Gibbs, J. P. (1966). Sanctions. *Social Problems*, 14(2):147–159.

Governatori, G. and Rotolo, A. (2010). Changing legal systems: legal abrogations and annulments in defeasible logic. *Logic Journal of IGPL*, 18(1):157–194.

Hübner, J. F., Sichman, J. S., and Boissier, O. (2004). Using the Moise+ for a cooperative framework of MAS reorganisation. In Bazzan, A. L. C. and Labidi, S., editors, *Advances in Artificial Intelligence - SBIA 2004, 17th Brazilian Symposium on Artificial Intelligence, São Luis, Maranhão, Brazil, September 29 - October 1, 2004, Proceedings*, volume 3171 of *Lecture Notes in Computer Science*, pages 506–515. Springer.

Huhns, M. N. (2001). Interaction-oriented programming. In Ciancarini, P. and Wooldridge, M. J., editors, *Agent-Oriented Software Engineering, First International Workshop, AOSE 2000, Limerick, Ireland, June 10, 2000, Revised Papers*, volume 1957 of *Lecture Notes in Computer Science*, pages 29–44. Springer.

Nardin, L. G., Balke-Visser, T., Ajmeri, N., Kalia, A. K., Sichman, J. S., and Singh, M. P. (2016). Classifying sanctions and designing a conceptual sanctioning process model for socio-technical systems. *The Knowledge Engineering Review*, 31(2):142–166.

Noriega, P. and de Jonge, D. (2016). *Electronic Institutions: The EI/EIDE Framework*, pages 47–76. Springer International Publishing, Cham.

Padget, J., ElDeen Elakehal, E., Li, T., and De Vos, M. (2016). *InstAL: An Institutional Action Language*, pages 101–124. Springer International Publishing, Cham.

Peczenik, A. and Hage, J. C. (1989). *On law and reason*, volume 379. Springer.

Piunti, M., Ricci, A., Boissier, O., and Hübner, J. F. (2009). Embodying organisations in multi-agent work environments. In *Proceedings of the 2009 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT 2009, Milan, Italy, 15-18 September 2009*, pages 511–518. IEEE Computer Society.

Pynadath, D. V., Tambe, M., Chauvat, N., and Cavedon, L. (1999). Toward team-oriented programming. In Jennings, N. R. and Lespérance, Y., editors, *Intelligent Agents VI, Agent Theories, Architectures, and Languages (ATAL), 6th International Workshop, ATAL '99, Orlando, Florida, USA, July 15-17, 1999, Proceedings*, volume 1757 of *Lecture Notes in Computer Science*, pages 233–247. Springer.

Ricci, A., Piunti, M., and Viroli, M. (2011). Environment programming in multi-agent systems: an artifact-based perspective. *Auton. Agents Multi Agent Syst.*, 23(2):158–192.

Rodrigues, M. R., da Rocha Costa, A. C., and Bordini, R. H. (2003). A system of exchange values to support social interactions in artificial societies. In *The Second International Joint Conference on Autonomous Agents & Multiagent Systems, AAMAS 2003, July 14-18, 2003, Melbourne, Victoria, Australia, Proceedings*, pages 81–88. ACM.

Searle, J. (1995). *The Construction of Social Reality*. Free Press.

Shoham, Y. (1993). Agent-oriented programming. *Artif. Intell.*, 60(1):51–92.

Wooldridge, M. (2009). *An introduction to multiagent systems*. John wiley & sons, USA.

Yan, E., Nardin, L. G., Boissier, O., and Sichman, J. S. (2025a). A regulation adaptation model for multi-agent systems. In *28th European Conference on Artificial Intelligence (Accepted)*.

Yan, E., Nardin, L. G., Boissier, O., and Sichman, J. S. (2025b). A unified view on regulation management in multi-agent systems. In *Coordination, Organizations, Institutions, Norms, and Ethics for Governance of Multi-Agent Systems - International Workshop, COINE 2025*.

Yan, E., Nardin, L. G., Hübner, J. F., and Boissier, O. (2025c). An agent-centric perspective on norm enforcement and sanctions. In Cranefield, S., Nardin, L. G., and Lloyd, N., editors, *Coordination, Organizations, Institutions, Norms, and Ethics for Governance of Multi-Agent Systems XVII - International Workshop, COINE 2024, Auckland, New Zealand, May 7, 2024, Revised Selected Papers*, volume 15398 of *Lecture Notes in Computer Science*, pages 79–99. Springer.