

Controlling multi-party interaction within normative multi-agent organizations

Olivier Boissier¹, Flavien Balbo^{2,3}, and Fabien Badeig^{2,3}

¹ Ecole Nationale Supérieure des Mines,
158 Cours Fauriel, 42100 Saint-Etienne, France

`Olivier.Boissier@emse.fr`

² Université Paris-Dauphine - LAMSADE,
Place du Maréchal De Lattre de Tassigny, F-75775 Paris 16 Cedex, France

`balbo@lamsade.dauphine.fr`

³ INRETS - GRETIA,
2, Rue de la Butte Verte, 93166 Noisy Le Grand, France
`badeig@inrets.fr`

Abstract. Multi-party communications taking place within organizations lead to different interaction modes between agents (e.g. (in)direct communication between roles, (in)direct communication restricted to a group, etc). Fully normative organisations need to regulate and control those modes as they do for agents' behaviors. This paper proposes to extend the normative organization model *MOISE* in order to specify such interaction modes between autonomous agents participating to an organization. This specification has two purposes: (*i*) to make the multi-agent organization able to monitor the interaction between the agents, (*ii*) to make the agents able to reason on these modes as they can do on norms. The paper is focused on the first point. We illustrate with a crisis management application how this extension has been implemented thanks to a specialization of the *EASI* interaction model.

1 Introduction

In a Multi-Agent System (MAS), interaction and organization play key and essential roles. A MAS is often described as composed of agents situated in a shared environment interact directly or indirectly with each other to execute and cooperate in a distributed and decentralized setting according to an organization. The *EASI* model⁴ [7] proposes a multi-party environment based interaction model and is therefore able to support the complexity of the interaction within an organization. On one hand, agents are able to send messages to other agents situated in the environment and, on the other hand, any agent situated in the environment is able to perceive the exchanged message. It is thus possible to consider, for the same message, direct, indirect and overhearing communications. If needed, the *EASI* model preserves the privacy of the interaction, more details are given in [7].

⁴ Environment as Active Support for Interaction

However, it is not possible to make the agents aware of this interaction settings. There is no declarative representation usable at the agent level.

Agents in a MAS are often structured along one organization that helps and/or constraints their cooperation schemes. Current proposals offer modeling languages usable either by agents either by an organization management system dedicated to the regulation and supervision of the agents within the defined organization. The *MOISE* model [5] is one of these proposals. Its organization modeling language is composed of two dimensions – structural and functional - connected to each other by a normative specification. Such a feature makes it possible to easily extend the model with new dimensions. Currently, there doesn't exist any dimension dedicated to the definition of interaction modes within the organization. It is thus not possible to govern the agents interaction modes resulting from the multi-party communications offered in the EASI framework.

In this paper, our aim is to propose a unified model for interaction and organization. To reach this objective we propose the enrichment of the *MOISE* organization modeling language with a new and independent dimension connected to the other ones by the normative specification. This way it is thus possible to make the agents able of reasoning on their use of the interaction modes offered in the EASI platform. The next step will be to develop such reasoning capabilities in the agents. In this paper we focus on the presentation of the unified model and how it is translated to be monitored by the facilities offered by the EASI platform. The MAS designer will be able to use the resulting specification of both the organization and the interaction and to get the corresponding support environment.

The paper is organized as follows. In section 2, we present the background of the proposal and motivate our choices. In section 3, we expose how *MOISE* organization modeling language has been extended to specify the interaction modes proposed by EASI. The section 4 describes how this specification is mapped to the EASI model. In section 5, we show the expressing capabilities of the proposal with different examples issued of a crisis management application. Before conclusion, we compare our proposal to the current related approaches.

2 Background

In this paper, we consider a crisis management application where different dedicated emergency services must be coordinated in order to solve a crisis situation. The main difficulty in the modeling of such an application consists in the definition of the interaction constraints between those services, given that each service has the possibility to decide on its own which interaction mode to use. We use this application all along the paper to illustrate the components of our proposal. In the following sections, we specify parts of the multi-agent organization governing this application thanks to the models *MOISE* and EASI.

2.1 \mathcal{M} oise

The \mathcal{M} oise framework [4] is composed of an organization modeling language, an organization management infrastructure and organization based reasoning mechanisms at the agent level. In this paper, we focus on the organization modeling language. Our aim is to use it with the EASI platform in order to specify and regulate the different interaction modes available on this platform (see next section).

The organization modeling language considers the specification of an organization along three independent dimensions⁵: structural (SS), functional (FS) and normative (\mathcal{NS}). Whereas SS and FS are independent, \mathcal{NS} defines a set of norms binding elements of both specifications. The aim is that the agents enact the behaviors specified in \mathcal{NS} when participating to the organization. The organization modeling language is accompanied by a graphical language (cf. Fig. 1, 2) and XML is used to store the organizational specifications.

Structural Dimension: The structural dimension specifies the *roles*, *groups*, and *links* of an organization. It is defined with the following tuple: $\langle \mathcal{R}, \sqsubset, rg \rangle$ with \mathcal{R} set of the roles, \sqsubset , inheritance relation between roles, rg organization root group specification. The definition of this group gives the *compatibility* relations between roles, the maximal and minimal *cardinality* of agents that can endorse roles within the group, the *links* connecting roles to each other (communication, authority, acquaintance) and *sub-groups*. In \mathcal{NS} , the role is used to bind a set of constraints on behaviors that the agent commits to satisfy as soon it endorses the role.

In the crisis application, we define (cf. Fig. 1) two main groups which correspond to the tactical spheres used in a crisis management: decision-making sphere (**Decision-making**) and operational sphere (**Operational**). For each of them, we define the roles *manager* and *operator* inheriting the generic role *role-player*. These roles are specialized respectively in *coordinator*, *leader_D* for the group **Decision-making** and *leaders_S* for the subgroups of group **Operational**. The role *coordinator* (resp. *leader_D*) can be played by only and only one agent - 1..1 - (resp. several agents - 1..* -). A compatibility link connects the role *leader_D* to *leaders_S* meaning that any agent playing *leader_D* will be able to play also the role *leaders_S*. Six communication links (cf. l_1 to l_6) have been defined between these roles (e.g. l_1 communication link between *coordinator* and *leader_D*).

Functional Dimension: The functional dimension is defined by $\langle \mathcal{M}, \mathcal{G}, \mathcal{S} \rangle$ with \mathcal{M} set of *missions*, consistent grouping of collective or individual goals. A mission defines all the goals an agent commits to when participating in the execution of a social scheme by the way of the roles that they endorse. \mathcal{G} is the set of the *collective or individual goals* to be satisfied and \mathcal{S} is the set of *social schemes*, tree-like structurations of the goals into plans.

⁵ In this paper, we will provide the only necessary details in order to globally understand the model as well as the proposed extensions. For further details, readers should refer to <http://moise.sourceforge.net/>.

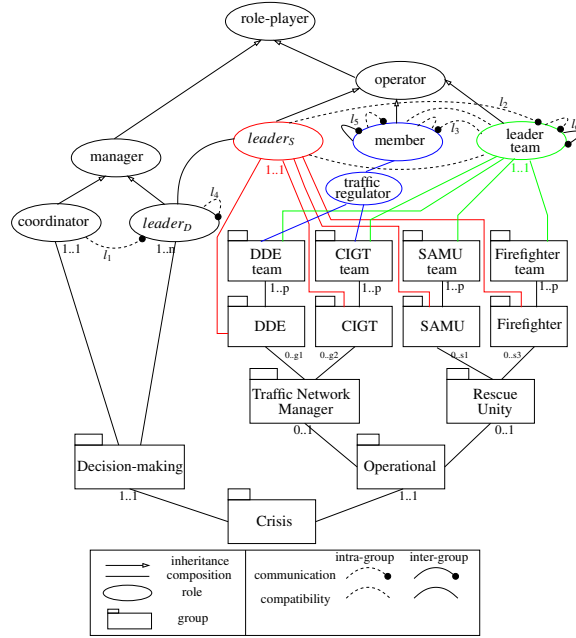


Fig. 1. Partial graphical view of the structural specification for the crisis management application

The Fig. 2 illustrates a social scheme of FS expressing the collective plan for deciding within the crisis management application. According to it, agents should aggregate the different information in relation to the crisis situation Refining crisis perception, Safeguarding zone by executing one of the two social schemes (scheme 1 or scheme 2 that are not detailed here) and execute the scheme 3. The different goals are organized into missions.

Normative Dimension: The normative dimension \mathcal{NS} defines a set of norms as: $\langle id, c, \rho, dm, m \rangle$ with id norm identifier, c activation condition of the norm⁶, ρ role concerned by the deontic modality, dm deontic modality (obligation or permission), m mission. A normative expression can be read as : “when c holds, any agent playing role ρ has dm to commit on the mission m ”. Within this language, norms are either a *permission*, either an *obligation* for a role to commit to a mission. Goals are indirectly connected to roles since a mission is a set of goals. Interdictions are supposed to exist by default: if the normative specification doesn’t have any permission or obligation for a couple mission, role, any agent playing the role is forbidden to commit to mission. A norm becomes in the *active* state (resp. *inactive*) as soon as the condition c holds (resp. doesn’t

⁶ Predicates bearing on the current organization state (e.g. *plays*, *committed*, etc) and/or bearing on particular configurations of the application.

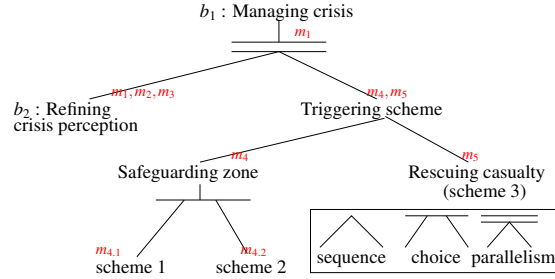


Fig. 2. Graphical view of the social scheme for decision within the crisis management application. Goals are the nodes of the tree. Missions to which the goals are assigned are in red.

hold). When the norm is active, the deontic expression attached to the norm can be verified. The norm can thus become *fulfilled* or *unfulfilled*.

For instance, in the crisis management application, the norm obliging agents playing the role $leaders_S$ in the group Traffic Network Management (TNM) to safeguard the zone where the accident took place (mission m_4) is: $\langle n_1, c_1, leaders_S, obligation, m_4 \rangle$ where c_1 is $plays(bearer, leaders_S, TNM)$. The term *bearer* refers to the agent that will play the role “bearer” in the context of the obligation issued from the instantiation of the norm in the organization entity (see below) and *plays* is a predicate satisfied when the agent plays the $leaders_S$ in an instance of group TNM. When the zone is secured, the agents playing the same role within the context of the group Rescue Unity (RU) deploys the intervening scheme (mission m_5) following the norm: $\langle n_2, c_2, leader_D, obligation, m_5 \rangle$ where c_2 is $plays(bearer, leader_S, RU)$.

Organizational Entity: An organizational entity (OE) is defined from the organizational specification OS and a set of agents \mathcal{A} by the following tuple: $\langle OS, \mathcal{A}, \mathcal{GI}, \mathcal{SI}, \mathcal{O} \rangle$ where \mathcal{GI} is the set of concrete groups of the organization, i.e. groups dynamically created from the group specification of the OS , \mathcal{SI} is the set of concrete social schemes dynamically created in the OE from the social schemes specification in the OS and \mathcal{O} is the set of obligations issued from the norms \mathcal{NS} attached to agents of \mathcal{A} whose conditions are satisfied [6].

2.2 Easi

The multi-party interaction model EASI supports the management of direct, indirect and overhearing communications in a MAS [7]. For cognitive agents, the common point between all these communication modes consists in the *routing* of the messages by identifying *which agent* should obtain *which message* and in *which context*. Solving this problem requires taking into account both sides of the sender and potential receivers. To this aim, EASI manages meta-informations on the MAS (agents, messages, context) in the communication environment and use them to help the agents to interact. The interaction model EASI is thus

defined by $\langle \Omega, \mathcal{D}, P, \mathcal{F} \rangle$ with $\Omega = \{\omega_1, \dots, \omega_m\}$ the set of entities ($\mathcal{A} \subset \Omega$ and $MSG \subset \Omega$ - \mathcal{A} set of agents and MSG set of messages -), $\mathcal{D} = \{d_1, \dots, d_m\}$ set of domain descriptions of the properties, $P = \{p_1, \dots, p_n\}$ set of properties, and $\mathcal{F} = \{f_1, \dots, f_k\}$ set of filters.

Entity: The entities are the meta-information on the MAS that EASI manages. An entity $\omega_i \in \Omega$ is defined by $\langle e_r, e_d \rangle$ where e_r is a reference to an element of the MAS and e_d is the description of that element. An element of the MAS can be agents (\mathcal{A}), messages (MSG) and a reference is its physical address on the platform or other objects such as URL, mailbox, The description e_d is defined by a set of couples $\langle p_i, v_j \rangle$ where $p_i \in P$ and v_j is the value of the property for this entity. Any agent of the MAS has its own processing and knowledge settings. It is connected to the communication environment by the way of its description that it stores and updates in this environment. This description e_d is used for the routing of the informations to the reference e_r .

Property: A property gives an information on an entity. A property $p_i \in P : \Omega \rightarrow d_j \cup \{unknown, null\}$ is a function whose description domain $d_j \in \mathcal{D}$ can be quantitative, qualitative or a finite set of data. The *unknown* value is used when the value of the property cannot be set, and *null* is used to state that the property is undefined in the given description. In order to simplify the notation, only the value of the description domain is given to specify a property.

For instance, in the crisis management application, the properties attached to agents and messages are *id, role, position, subject, sender* with:

- *id* : $\Omega \rightarrow N$,
- *role* : $\Omega \rightarrow \{coordinator, leaders\}$,
- *position* : $\Omega \rightarrow N \times N$,
- *subject* : $\Omega \rightarrow \{alert, demand\}$,
- *sender* : $\Omega \rightarrow N$.

An agent a can have the following description $\langle role, coordinator \rangle$ and an agent $b \langle role, leaders \rangle, \langle position, (10, 20) \rangle$ and a message $m \langle subject, alert \rangle, \langle position, (15, 20) \rangle$.

Filter: The filter identifies the entities according to their description (e_d) and realizes the interaction between the concrete objects (e_r). A filter $f_j \in \mathcal{F}$ is a tuple $f_j = \langle f_a, f_m, [f_c], n_f \rangle$ with n_f filter name. The assertion $f_a : A \rightarrow \{T, F\}$ identifies the receiving agents (*which agent*), the assertion $f_m : MSG \rightarrow \{T, F\}$ identifies the concerned messages (*which message*), and $f_c : \mathcal{P}(\Omega) \rightarrow \{T, F\}$ is an optional set of assertions identifying other entities of the context (*which context*).

Each agent $?r$ ('?' preceedings a letter denotes a variable) whose description validates f_a receives in its mailbox the message $?m$ that satisfies f_m if there exists a set of entities in the $?c$ such that f_c is true.

A filter is therefore valid for any tuple $\langle ?r \in \mathcal{A}, ?m \in MSG, ?c \subset \Omega \rangle$

For instance, in the crisis management application, the filter Fe sets the routing of the communication as follows ('=' is the comparison operator):

- The agents with the role `leaderD` that are situated in the crisis origin:
 $f_a : [role(?r) = leader_D] \wedge [position(?r) = (0, 0)]$
- should receive the alert messages:
 $f_m : [subject(?m) = alert] \wedge [sender(?m) = ?ide]$
- of the agent playing the role `coordinator`:
 $f_C : [id(?e) = ?ide] \wedge [role(?e) = coordinator]$

In this example, the description of the message sender ($?e$) that is identified thanks to the property *sender* in the message belongs to the context. Agents wishing to send or receive a message, update their description in the communication platform and add/retract dynamically in/from the environment filters that involve them. Thus the environment supports simultaneously direct interaction (including dyadic, broadcast multi-cast and group communication) and indirect interaction (including mutual-awareness and overhearing). If the filter is added by the future receiver of the message then the interaction is indirect: the depositary agent defines which message it wants to receive. If the filter is added by the future message sender then the interaction is direct: the depositary agent defines which agent it wants to contact.

According to the state of the different descriptions within the environment, the triggered filter will enable the routing of the messages in the different interaction modes towards the corresponding targets.

Even if EASI offers an advanced communication management by identifying precisely the interaction context, it cannot be used by the agents in order to reason on the causes of the interaction. For instance, the filter Fe will permit the routing of messages but the reasons of this requirement is not expressed within EASI. For the filter Fe , the choice of the communication mode may depend on the relations between the roles `coordinator` and `leaderD`: the *coordinator* sends messages to *leaders* (direct mode) for dedicated messages whereas the *leaders* listen to all the messages issued from the *coordinator* (indirect mode). Using this knowledge, an agent could reason on the current interactions. For instance, the coordinator may choose a direct interaction to handle certain informations and indirect interaction for others. The leaders can deduce the importance to the informations according to the filter used to receive informations. The specification of communications within an organizational model would help the agents to relate communication filters to the reasons that cause the use of such a communication channel.

3 Extending \mathcal{M} oise for Easi

In order to clearly specify the interaction modes used in EASI, we are going to enrich and extend the organization modeling language of \mathcal{M} oise with a new dimension. This new dimension is called *communication mode specification* (noted \mathcal{CS}). It is dedicated to expressing the communication modes that will be used within the organization. As the other \mathcal{M} oise dimensions, we keep it independent of SS and FS . We use the same principle to connect it to the other dimensions and enrich the normative specification accordingly. The aim is to be able to

connect the communication modes to the structure and functioning of the organization by the way of norms. Those norms will be made accessible to the agents when interacting with other agents of the organization.

The organization specification is thus enriched into the following 4-uple: $\langle SS, FS, CS, NS \rangle$ with CS communication modes specification and NS the modified normative specification. We detail these two components in what follows.

3.1 Communication modes specification

The specification CS is composed of the set of communication modes $cm \in CS$ considered in the organization.

A communication mode is defined as: $\langle type, direction, protocol \rangle$ with *type*, the type of the communication mode (*direct* or *indirect*), *direction*, the message transmission direction (*unidirectional* or *bidirectional*), *protocol*, the interaction protocol that is used. The values of this last variable correspond to the name of the different interaction protocols that the designer wishes to be used and deployed in the organization (e.g. *FIPAREQUEST*, *Publish_Subscribe*, ...).

As we will see below, a communication mode qualifies the communication link defined in the structural specification between roles. The communication link is directed from the *initiator* of the communication - source of the link - to the *participant* - target of the link -. Therefore, a link can be considered as:

- a unidirectional channel, letting circulate messages in only one direction,
- a bidirectional channel, letting circulate messages in both directions from the initiator to the participant and inversely.

Orthogonal to these two directions, we consider the direct and indirect interaction models proposed within EASI.

In the crisis management application, we define, for instance, the two following communication modes $cm_{d,b}$ and $cm_{i,u}$:

$cm_{d,b}$: $\langle direct, bidirectional, FIPAREQUEST \rangle$

$cm_{i,u}$: $\langle indirect, unidirectional, PublishSubscribe \rangle$

where $cm_{d,b}$ is used to directly ask for information whereas $cm_{i,u}$ is used to provide informations to agents that will consult it when they want.

3.2 Communication Norms

In order to bind communication link and communication mode as defined in CS by making explicit the deontic modalities attached to their use, we generalize the writing of the norms described in the MOISE initial version: $\langle id, c, \rho, dm, object \rangle$ where *id* is norm identifier, *ca* the activation condition, ρ the role on which the deontic modality bears, *dm* the deontic modality (obligation or permission), *object* the subject of the norm.

Object of a norm: The object of a norm *object* is defined as the two following expressions:

- $do(m)$ in the case where a mission m has to be executed - case initially considered in *MOISE*,
- $use(l, cm, \alpha)$ in the case where the communication mode cm should be used for the link l in the context α .

Context: The context α defines the constraints bearing on the descriptions e_d of the entities $\omega_i \in \Omega$ (cf. Sec. 2.2) involved in the interaction using this communication link: sender, receiver, message. It is also possible to add additional descriptions issued from other entities of the MAS (e.g. requirements of the agent, ...). It is thus possible to use a mission as context of use of the link or a particular goal as context of use of this link. We will define in the following section the way we express these constraints when describing how EASI has been specialized to handle *MOISE*. When α 's status is T (true), the link is usable in any situation.

Let's consider the communication link l_1 used by the agents playing the role *coordinator* towards agents playing the role *leader_D* (eg. Fig. 1) in the crisis management application. Given the normative specification that we have defined, it is possible to bind to it the communication mode $cm_{i,u}$ defined above, by writing the following norm: $n_1 \langle n_1, c_1, coordinator, obligation, use(l_1, cm_{i,u}, T) \rangle$ with $c_1 : committed(m_1)$ to express that l_1 ought to be used by agents playing the role *coordinator* when they are committed to the mission m_1 . No particular context is attached to the use of the communication mode $cm_{i,u}$.

We can also attach to this link another communication mode $cm_{d,b}$, by defining a new norm $n_2 : \langle n_2, c_2, coordinator, obligation, use(l_1, cm_{d,b}, \alpha_2) \rangle$ with $c_2 : committed(m_4)$ by specifying a context α_2 (cf. following section for the syntax) stating that the communication on the link l_1 takes place for the sending of messages to agents belonging to group *CIGT* (Center of the Engineering and Management of the Traffic). The link l_1 can also be bound to the same constraints but for communication in the context α_3 stating the sending of messages from the agent playing role *coordinator* to agent belonging to group *TNM*: $\langle n_3, c_2, coordinator, obligation, use(l_1, cm_{d,b}, \alpha_3) \rangle$.

In the following, we will need to access to the different features of a communication link from the structural specification. We will use pointed notation $l_j.initiator$ (resp. $l_j.participant$) to access to the source role (resp. target) of the link l_j , and $l_j.group$ to access to the group in which l_j is defined.

4 Specializing Easi for *Moise*

Our objective is to generate filters for the communication environment from the specifications defined in the organization modeling language. These filters use informations on the organization. These informations should be stored in the description of the entities that are managed by the communication environment

in order to be accessible to the filters. In this section, we identify and define the necessary properties for describing the agents and messages in the communication environment. Then we describe a generic filter generated from the communication norms that we just defined in the previous section.

4.1 Properties

In order to connect organization and interaction, it is necessary to give a minimal description of an agent, of a message while incorporating this new dimension in them. Given the definition of an entity in section 2.2, we define the following properties that are accessible in the environment for each type of entity.

Agent Properties: The description of an agent is at least composed of the *id* and *org* properties, where:

- *id* returns the identifier of the agent ($id : A \rightarrow ID_A$ with ID_A set of agent identifiers),
- *org* returning the subset of organizational descriptions coming from the participation of the agent to the organization ($org : A \rightarrow \mathcal{P}(OC)$ with OC set of organization descriptions).

An organization description $oc_i \in OC$ is defined by: $oc_i = \langle ig : g, r, m, go \rangle$ with $ig \in \mathcal{IG}$, $g \in \{rg\} \cup rg.subgroups$, $r \in R$, $m \in M$, $go \in G$. *ig* is a concrete group created from the group specification *g* defined in the SS of the organization. The parameter *rg* and the sets *R*, *M*, *G* are defined in the organization specification (cf. Sec. 2.1).

For instance, in the crisis management application, the agent *a* described by $org(a) = \{\langle g1 : Decision_making, leader_D, m_2, b_2 \rangle, \langle g2 : DDE, leader_S, m_1, b_1 \rangle\}$, belongs to the group *g1* of type Decision-making and to a group *g2* of type DDE, in which it plays respectively role *leader_D*, committed to mission *m₂*, trying to achieve goal *b₂* and the role *leader_S*, committed to the mission *m₁* trying to achieve the goal *b₁*.

This description of an agent is minimal. We defined two management modes. Being related to the organization, these properties can be managed without being intrusive: management by the organization management infrastructure. However, if this set is complemented by specific properties related to the internal state of the agents, their management is ensured by the agents themselves. For instance, a property *availability* returning the availability of the agent has a value that is related to the choice of the agent itself.

Message Properties: In the same way, we specialize the description of a message with the following set of minimal properties *sender*, *receiver*, *subject*, *rc*, *sc* where:

- *sender* : $MSG \rightarrow ID_A$,
- *receiver* : $MSG \rightarrow \mathcal{P}(ID_A) \cup \{unknown\}$,
- *subject* : $MSG \rightarrow D_{subject} \cup \{unknown\}$, with $D_{subject} = G \cup R \cup \{expression\}$, *expression* is a string,

- $rc : MSG \rightarrow OC \cup \{unknown\}$ being the reception context,
- $sc : MSG \rightarrow OC \cup \{unknown\}$ being the sending context.

Using these properties, the sender gives informations on the organizational context in which the interaction takes place. For a message, each of these properties can receive a value or the value *unknown*. The more the sender specifies values of properties, the more precise will be the filter that can be used for the routing. We impose that the property *sender* doesn't get a value *unknown* in order to avoid anonymous messages.

Given these different properties, we have now the possibility of a routing ranging from indirect interaction, based on only the identifier of the sender, to one, focused on a subset of receivers (*receiver*) in a particular organizational context (*rc*), with a sender being in a given organizational context (*sc*) and the message with a particular content (*subject*).

The sender can also decide to define *patterns* for conditioning the routing along different organizational contexts. To this aim, it can use the symbol '*'*' as value for an element of the organizational context. This symbol denotes that the value is not a constraint in the choice of the tuples.

For instance, in the crisis management application, the expression $\langle _ : DDE, _ , m_2, _ \rangle$ defines an organizational pattern of *OC* such that the concrete group must be of type DDE and the mission is m_2 whatever are the values for roles and goals. The message mes_1 described below means that the sender whose identifier is a_1 and having the goal b_2 (sending context) sends a message to the agents a_2 and a_4 . In this case, the processing of the message is not constrained by the organizational states of the participating agents. They only have to be trying to achieve the organizational goal b_2 . $\langle \langle sender, a_1 \rangle, \langle receiver, \{a_2, a_4\} \rangle, \langle subject, demand \rangle, \langle rc, \langle _ : _ , _ , _ , b_2 \rangle \rangle, \langle sc, \langle _ : _ , _ , _ , b_2 \rangle \rangle \rangle$

For the sender, these are only possibilities since the routing of the message depends on the filters that are installed in the communication environment.

In fact, according to the filters that are installed in it, the routing of the message can lead to different situations: interaction as intended by the sender, no interaction or interaction not intended by the sender. For instance, the agent a_2 can receive the message although it doesn't have the goal b_2 in the case there exists a filter enabling the reception of messages from the agent a_1 , whatever are the values for the properties of the message.

In each message is stored the organizational context of its sending in order to enable the agents to filter them. An agent can thus choose to receive messages or to route them according to their organizational contexts without being imposed their use. Moreover, this definition of messages enables to consider the evolution of the organization state. Thus, a message kept in the environment can still be received by an agent in case of change of the organization state. For instance, an agent can be interested by any message whose receiving context concerns a role that it just endorsed. It is useful to keep an history of the past interaction

to better understand the current situation. Another advantage is to avoid that a message is missed because it has been sent before the agent has endorsed the role. In order to avoid a risk of confusion between messages, a property related to the time value of their emission or related to their life time can be added to the message description. This choice belongs to the system designer and is out of the scope of this paper.

4.2 From Communication Norms to Environment Filters

The activation of a norm for a communication link leads to the generation and addition of a filter in the environment. This filter is called *normative filter*. It corresponds to the exact translation of the norm as it is instantiated by the organization management system. Thanks to the organization management system, the agents are aware of the norm activation. Besides to the normative filters, the communication environment contains also filters set by the agents according to their activity in the system. In case of direct interaction, the sender knows that it can reach the agents identified as receiver in the norm. In case of indirect interaction, the receiver knows that it can receive messages identified in the norm.

A normative filter uses all the possible informations coming from the organizational specification and routes a message according to its *sc* and *rc* properties. The property *receiver* is not used in the generation process of a normative filter since it requires that the sender knows the identifiers of the agents. This is a too strong hypothesis. The same way, since the routing comes from the activation of a norm, the filter cannot constrain the subject of the message (*subject*) except additional conditions in the norms (context α of the object of the norm). The filter identifies a state of the context corresponding to the interaction. It is identical in the direct and indirect cases. We then propose a *generating pattern* that will be specialized for each activated communication norm.

Access to the organizational specification: The normative filter is created when the norm is activated as follows.

Let's first define the functions *initiator* and *participant* that access to the agents involved in the communication link defined in the object of the norm.

$$\textit{initiator} : \mathcal{O} \rightarrow A \qquad \textit{participant} : \mathcal{O} \rightarrow A$$

These functions return, for an instantiated norm, the agent (*initiator*), who initiates, or the one (*participant*), who participates, to the interaction. From these two functions, we express constraints on the descriptions of the agents. For instance $org(\textit{initiator}(n_j))$ makes possible to access to the organizational context attached to the description of the agent initiating the communication in the context of the instantiated norm n_j in which it is involved.

Let's define the predicate $achieves_\alpha$ that is automatically generated from the constraints expressed in the context α of the object of a norm. This predicate checks that the context is satisfied given the initiator, participant, message and entity descriptions in the environment, given α :

$$achieves_\alpha : A \times MSG \times A \times \mathcal{P}(\Omega) \rightarrow \{T, F\}$$

Given the previous definitions, we are able now to express the generic normative filter $f_{n_k}(?p, ?m, \{?i, C\})$ for the receiver $?p$ of the message $?m$ sent by $?i$ in the context C . This filter has been generated from the activation of the norm n_k . The object of the norm bears on the communication link l_j . It is composed of assertions f_a that identifies the receiver of the message $?p$ according to its organizational context, f_m that identifies the message $?m$ according to its organizational context and f_c that identifies the organizational context of the sender and the constraints α of the norm n_k .

$$\begin{aligned} f_a &: \langle [org(?p) \ni \langle ?x : l_j.group, l_j.participant, -, - \rangle] \rangle \\ f_m &: \langle [sender(?m) = id(?i)] \wedge [sc(?m) = \langle ?y : l_j.group, l_j.initiator, -, - \rangle] \wedge [rc(?m) = \langle ?x : l_j.group, l_j.participant, -, - \rangle] \rangle \\ f_C &: \langle [org(?i) \ni \langle ?y : l_j.group, l_j.initiator, -, - \rangle] \wedge achieves_\alpha(?p, ?m, ?i, C) \rangle \end{aligned}$$

Let's consider again the norm n_2 of the crisis management application: $\langle n_2, committed(m_4), coordinator, obligation, use(l_1, cm_{d,b}, \alpha_2) \rangle$ with $\alpha_2 : [(- : CIGT, -, -, -) \in org(participant(n_2))]$. The interaction is a direct and bidirectional one (cf. $cm_{d,b}$ of n_2). The sending agent deposits the first message. The two necessary filters have been generated and added thanks to the activation of n_2 .

The normative filter generated for n_2 for the interaction from initiator to participant is $f_{n_2}(?p, ?m, \{?i, C\})$: where :

$$\begin{aligned} f_a &: \langle [org(?p) \ni \langle ?x : Decision_making, leader_D, -, - \rangle] \rangle \\ f_m &: \langle [sender(?m) = id(?i)] \wedge [sc(?m) = \langle ?y : Decision_making, coordinator, -, - \rangle] \wedge [rc(?m) = \langle ?x : Decision_making, leader_D, -, - \rangle] \rangle \\ f_C &: \langle [org(?i) \ni \langle ?y : Decision_making, coordinator, -, - \rangle] \wedge [org(?p) \ni (- : CIGT, -, -, -)] \rangle \end{aligned}$$

The normative filter from the participant to the initiator is $f_{n_2}(?i, ?m, \{?p, C\})$ ⁷, where:

$$\begin{aligned} f_a &: \langle [org(?i) \ni \langle ?x : Decision_making, coordinator, -, - \rangle] \rangle \\ f_m &: \langle [sender(?m) = id(?p)] \wedge [sc(?m) = \langle ?y : Decision_making, leader_D, -, - \rangle] \wedge [rc(?m) = \langle ?x : Decision_making, coordinator, -, - \rangle] \rangle \\ f_c &: \langle [org(?p) \ni \langle ?y : Decision_making, leader_D, -, - \rangle] \wedge [org(?i) \ni (- : CIGT, -, -, -)] \rangle \end{aligned}$$

This way, for two agents participating to the same concrete group, the message sent by the initiator agent a_1 processed by the filter f_{n_2} will have the following description: $\langle \langle sender, id(a_1) \rangle, \langle rc, \langle g1 : Decision_making, coordinator, -, - \rangle \rangle, \langle sc, \langle g1 : Decision_making, leader_D, -, - \rangle \rangle \rangle$

The message sent by the participant agent a_2 , processed by f_{n_2} will have the following description: $\langle \langle sender, id(a_2) \rangle, \langle rc, \langle g1 : Decision_making, leader_D, -, - \rangle \rangle, \langle sc, \langle g1 : Decision_making, coordinator, -, - \rangle \rangle \rangle$

With these two filters, a communication channel has been created between agents having the roles coordinator and responsible in the group CIGT. The interaction model EASI has made possible to elaborate these filters. The MOISE model has made possible its use.

⁷ we continue to use the variable $?p$ for the participant in the interaction and $?i$ for the initiator given that the agent which is identified by the variable $?i$ who receives the message sent by $?p$.

5 Example

In this section, we illustrate and discuss the expressing capabilities of our proposal going back to the interaction modes attached to the communication link l_1 issued of the communication norms n_1, n_2, n_3 in the crisis management application described in the paper.

- $\langle n_1, c_1, coordinator, obligation, use(l_1, cm_{i,u}, T) \rangle$ with $c_1 : committed(m_1)$
- $\langle n_2, c_2, coordinator, obligation, use(l_1, cm_{d,b}, \alpha_2) \rangle$ with $c_2 : committed(m_4)$
and $\alpha_2 : \langle - : CIGT, -, -, - \rangle \in org(participant(n_2))$
- $\langle n_3, c_3, coordinator, obligation, use(l_1, cm_{d,b}, \alpha_3) \rangle$ with $c_3 : committed(m_4)$
and $\alpha_3 : \langle - : TNM, -, -, - \rangle \in org(participant(n_3))$

On these three norms, the differences bear on the *activation conditions* of the norm c_x , the *communication mode* $cm_{x,y}$ and the communication *context* specified in the object.

The norm n_1 whose activation condition bears on the management of the crisis (mission m_1) is activated during all the crisis management. The norms n_2 and n_3 are not active since the agents on which the norms bear are committed on the mission m_4 .

The predicate $achieves_\alpha$ of the normative filter f_{n_1} generated from the norm n_1 is always true ($\alpha_1 = T$). According to this norm, all the agents playing the role $leader_D$ (target of link l_1) must consult the informations set available by any agent playing the role *coordinator*. The norm n_2 imposes a direct interaction in the context of mission m_4 so that the coordinator is able to get informations on the state of the transportation network. According to this norm, any agent playing the *coordinator* role can reach any agent playing the role $leader_D$ (target of link l_1) and being a member of concrete group of type CIGT. The normative filter f_{n_2} described in the previous section expresses these constraints. For the same mission m_4 , the coordinator requires information on the available resources in the services TNM. The normative filter f_{n_3} resulting from the activation of norm n_3 , enables the coordinator to reach any leader of each traffic network management service (TNM).

In our example, if the missions m_1, m_4 are under examination, the normative filters corresponding to the three norms are simultaneously present in the environment. From the point of view of the agent playing the role *coordinator*, it means that it can route messages directly to the agents who are $leader_D$ in groups of type CIGT (n_2) and broaden their demand to agents playing the role $leader_D$ in the groupes of type TNM (n_3) given its needs.

Let's turn to the agents playing the role $leader_D$ in the group Decision-making. If involved in the role $leader_S$ within the groups CIGT and TNM (let's notice that this situation is possible thanks to the compatibility link between both roles), the agents will receive the requests from the agent playing the role *coordinator* and will be able to know that this is a direct interaction issued from the coordinator. The agents will be able to answer to this agent by using the normative filter created in case of bidirectionnal interaction. Thanks to norm n_1 , every agent playing the role $leader_D$ will receive the messages sent by the agents playing

the role `coordinator` via the filter f_{n_1} , building a common and shared knowledge (indirect interaction). According to their processing activity, the agents will be more or less aware of these messages.

This short example that we can't detail more, shows the richness of expressiveness of the interaction modes made possible by combining EASI and MOISE as described in this paper.

6 Related work

To our knowledge, there doesn't exist a similar support to interaction enabling, for the same communication, to consider simultaneously the direct and indirect interaction modes.

Considering related works to the indirect interaction, the general principle consists in the use of a shared data space that is integrated or not to the environment [8]. In this approach, the tuples that are deposit by the sender in the shared space are compared to patterns expressing the needs of the receivers. These works are focused on the accompanying coordination language and don't consider, at any moment, the organization or the state of the agents.

Dealing with the direct interaction model, several works propose to use an organizational structure in order to manage the communications. In [1], the agents are organised in a hierarchy where each level knows the skills of the agents belonging to the lower level in order to make possible for the sender, a routing of the messages according to the skills. However, it is not an organizational model that is usable by the agents. In the AGR model [3], the organization constrains the interactions according to the groups to which the agents participate. It supports a routing of the message according to the organizational model (group, role). However, the only interaction mode is the direct one and the agents don't have access to an explicit description of the different specifications.

Normative organization models have been proposed in the literature in order to regulate and control the communication between agents. However the specifications address the interaction protocols, i.e. the coordination of the interaction instead of interaction modes. The only considered interaction mode is the direct one (e.g. ISLANDER [2]). They don't consider the interaction at the level addressed in this paper.

7 Conclusions

In this paper, we have proposed a specification of interaction modes between agents within an organization. For that aim, we have extended and enriched the organization modeling language of the MOISE framework. We have also shown how the specifications have been used to generate and to configure dynamically the communication environment supported by the EASI platform. We have illustrated the use of this proposal in a crisis management application.

In the future, we intend to extend the considered interaction modes to over-hearing. We will also consider the communication between groups by extending

the scope of communication to groups by enriching and modifying the structural specification of *MOISE*. Thanks to these new primitives in the organization specification, we can turn to the development of reasoning mechanisms at the agent level to make agents able to reason on the interaction modes that they can use within the organization.

Acknowledgement

We would like to thank D. Trabelsi, H. Hadioui and J.F. Hübner for the fruitful discussions about the content of this paper.

References

1. N. Bensaïd and P. Mathieu. A hybrid architecture for hierarchical agents. In *Proc. of ICCIMA '97*, pages 91–95, 1997.
2. M. Esteva, J. A. Rodriguez-Aguiar, C. Sierra, P. Garcia, and J. L. Arcos. On the formal specification of electronic institutions. In Frank Dignum and Carles Sierra, editors, *Proceedings of the Agent-mediated Electronic Commerce*, LNAI 1191, pages 126–147, Berlin, 2001. Springer.
3. J. Ferber and O. Gutknecht. A meta-model for the analysis and design of organizations in multi-agents systems. In Y. Demazeau, editor, *Proceedings of the 3rd International Conference on Multi-Agent Systems (ICMAS'98)*, pages 128–135. IEEE Press, 1998.
4. J. F. Hübner, O. Boissier, R. Kitio, and A. Ricci. Instrumenting Multi-Agent Organisations with Organisational Artifacts and Agents. *Journal of Autonomous Agents and Multi-Agent Systems*, 20(3), 2010.
5. J. F. Hübner, J. S. Sichman, and O. Boissier. A model for the structural, functional, and deontic specification of organizations in multiagent systems. In G. Bittencourt and G. L. Ramalho, editors, *Proceedings of the 16th Brazilian Symposium on Artificial Intelligence (SBIA'02)*, volume 2507 of *LNAI*, pages 118–128, Berlin, 2002. Springer.
6. Jomi Fred Hübner, Olivier Boissier, and Rafael H. Bordini. Normative programming for organisation management infrastructures. COIN Workshop at MALLOW, 2009.
7. J. Saunier and F. Balbo. Regulated multi-party communications and context awareness through the environment. *Journal on Multi-Agent and Grid Systems*, 5(1):75–91, 2009.
8. L. Tummolini, C. Castelfranchi, A. Ricci, M. Viroli, and A. Omicini. "exhibitionists" and "voyeurs" do it better: A shared environment approach for flexible coordination with tacit messages. In *Proc. of Workshop on Environments for Multi-Agent Systems*, LNAI 3374, pages 215–231. Springer Verlag, 2004.