Contextualizing Normative Open Multi-Agent Systems

Carolina Felicíssimo
DI–PUC-Rio / LIP6–PARIS6
R. M. de S. Vicente 225
Rio de Janeiro/RJ – Brazil
(+55) 21 2540 69 15 r.103
cfelicissimo@inf.puc-rio.br

Caroline Chopinaud
LIP6–PARIS6
104 Avenue du Président Kennedy
75016 – Paris – France
(+33) 1 44 27 87 53
caroline.chopinaud@lip6.fr

Jean-Pierre Briot
LIP6–PARIS6 / DI–PUC-Rio
104 Avenue du Président Kennedy
75016 – Paris – France
(+33) 1 44 27 87 53
jean-pierre.briot@lip6.fr

ABSTRACT

Open MASs can be extremely dynamic due to heterogeneous agents that migrate among them to obtain resources or services not found locally. In order to prevent malicious actions and to ensure agent trust, open MAS should be enhanced with normative mechanisms. However, it is not reasonable to expect that *foreign* agents know in advance all the norms of the MAS in which they will execute. Thus, this paper presents our DynaCROM approach for addressing these issues. From the individual agents' perspective, DynaCROM is an information mechanism so that agents become context norm-aware; from the system developers' perspective, DynaCROM is a methodology for norm management in regulated MASs. Notwithstanding the ultimate goal of a regulated MAS is to have an enforcement mechanism, we also present in the paper the integration of DynaCROM with SCAAR, its current solution for enforcing contextual norms.

Categories and Subject Descriptors

I.2.11 [Distributed Artificial Intelligence]: Multiagent systems.

General Terms

Management, Design, Reliability.

Keywords

Norm-aware agents, normative MAS, contextual information.

1. INTRODUCTION

Openness has led to software systems that have no centralized control and that are formed of autonomous entities [20]. Key characteristics of such systems are heterogeneity, conflicting individual goals and limited trust [1]. Open systems also can be extremely dynamic. In this work, we assume that a multi-agent system (MAS) is an open system that puts together sets of heterogeneous, self-interested agents whose actions may deviate from the expected behavior in a context.

Norms can be used in an open MAS to regulate agent execution and, then, to prevent the system to reach an undesirable state. Norms prescribe what should be done in order to fulfill a generalized expectation of behavior. In this sense, a normative MAS is a system that conforms to or is based on norms [2]. Actually, norms can also be viewed as event-driven rules that trigger under appropriate conditions of events happening in a regulated system and, by doing so, create, update or cancel commitments affecting a predefined set of agents [15]. Normative agents should be able to take into account the existence of social norms in their decisions (either to follow or violate a norm) and to react to violations of the norms by other agents [4].

In order to prevent malicious actions and to ensure agent trust in open MASs, these systems should be enhanced with normative mechanisms. Governance in open MASs is not straightforward since heterogeneity and autonomy rule out any assumption concerning the way third-party agents are implemented and behave [17]. Furthermore, agents' internal structures are normally inaccessible suggesting that norm verification should be based on social concepts, which are externally observable. Thus, it should be possible to provide a decentralized normative mechanism, which is not hard coded inside agents and in which norms can be dynamically updated for continuously regulate agents' actions.

This paper presents how developers can implement dynamic normative open MASs, in which norms can be updated at system runtime, and also how heterogeneous norm-aware agents can execute in open MASs supported with updated contextual norm information, both by using our DynaCROM approach [11] (meaning *dynamic contextual regulation information provision in open MASs*). Notwithstanding the ultimate goal of a regulated MAS is to have an enforcement mechanism, thus, we also present in the paper DynaCROM integrated with SCAAR (meaning *Self-Controlled Autonomous Agents geneRator*) [5]. SCAAR is in charge of enforcing DynaCROM contextual norms.

The remainder of this paper is organized as follows. Section 2 briefly presents the DynaCROM solution, including how to classify, represent and compose contextual norms. Section 3 presents the SCAAR norm enforcement mechanism. Section 4 describes a running example for explaining how DynaCROM effectively works. Section 5 points out related work and situates DynaCROM in the field. Finally, we draw our conclusions and outline future work in section 6.

2. CONTEXTUAL NORM INFORMATION PROVISION IN OPEN MASs

DynaCROM aims to support norm-aware agents with updated contextual norm information in open MASs. For this, developers

SAC'08, March 16-20, 2008, Fortaleza, Ceará, Brazil.

should classify, represent and compose their norms according to the DynaCROM approach in order to create a dynamic normative open MAS called a *DynaCROM MAS*.

2.1 Contextual Norm Classification

Basically, an MAS is constituted of environments, organizations and agents playing roles and interacting [22]. As environments, organizations, roles and agent interactions are important concepts for the understating of the text, we would like to characterize the meaning in which they are used in the paper.

Environments [34] are discrete computational locations, similar to places in the physical world, which provide conditions for agents to inhabit it. Environments can have refinement levels, such as a specialization relationship (e.g., country and state), but there cannot be overlaps (e.g., there cannot be two countries in the same place). An environment also can have many organizations. Organizations [13] are social locations in which groups of agents play roles. An organization can embody many sub-organizations, but each organization belongs to only one environment [31]. Agents can execute in different organizations and they can also migrate among environments and organizations in order to obtain resources or services not found locally. Roles [32] are abstractions that prescribe a set of related tasks, which agents must perform to achieve their designed goals. Roles are defined by organizations independently of agents' individual identities. An agent can interact with any other agent in an MAS by exchanging messages.

Environments, organizations, roles and interactions suggest different *contexts* for regulation in open MASs. Contexts are implicit situational information that can be used to characterize situations of agents and to provide relevant information and/or services to them, where relevancy depends on agent tasks [7]. Modular context refinements provide a more flexible system for developers while they are maintaining and evolving norm information and, consequently, managing system regulation.

DynaCROM follows directions taken by research into context-aware applications that suggest top-down architectures for classifying contextual information [19],[23]. In DynaCROM, norm information should be classified according to the *Environment*, *Organization*, *Role* and *Interaction* contexts. We call these contexts *regulatory contexts* and they are differentiated by the boundaries of their data (i.e. norms). More precisely, *Environment Norms* are applied to all agents in a regulated environment; *Organization Norms* are applied to all agents in a regulated organization; *Role Norms* are applied to all agents playing a regulated role; and *Interaction Norms* are applied to all agents involved in a regulated interaction. This set should be improved with additions of domain regulatory contexts for representing their norms.

2.2 Contextual Norm Representation

DynaCROM uses *contextual normative ontologies* to explicitly represent its data, having the Norm concept as a central asset. An *ontology* is a conceptual model that embodies shared conceptualizations of a given domain [18]; and a *contextual ontology* is an ontology that represents localized domain information [3] (e.g., "USD" as the currency of "USA"). The use of ontologies in open MASs supports heterogeneous agents with a common understanding about well-defined system regulation relating abstract concepts, in which contextual norms are formulated, to their concrete application domain.

The DynaCROM ontology defines five basic related concepts, as illustrated in Fig. 1[1] by multi-lines linked boxes. In each concept, the first line contains the concept's name and the others lines correspond to the concept's attributes. Each attribute's line is divided in three parts. The first part has the attribute's name. The second part contains the attribute's cardinality (i.e., *Instance* for a unique value and *Instance** for n-vary values) of an object property, which links the concept to the another one identified in the third part. For instance, the first line of the Role concept has "Role" as the concept's name; the second line has the multi-value object property "hasNorm", which links the "Role" and "Norm" concepts; and the third line has the object property "isPlayedIn", which links the "Role" and "Organization" concepts.

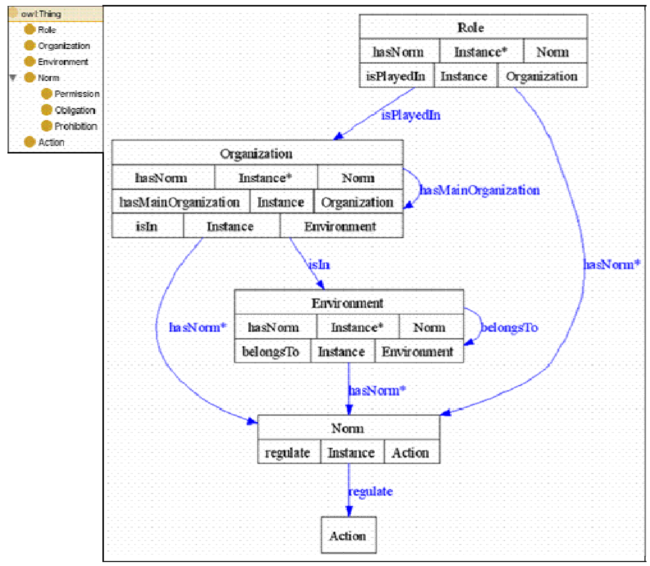

Fig. 1. The DynaCROM ontology.

In the DynaCROM ontology, the *Role* concept encompasses the instances of all regulated roles representing the system's role regulatory context. Each role instance has associations with its norms and organization. The *Organization* concept encompasses the instances of all regulated organizations representing the system's organization regulatory context. Each organization instance has associations with its norms, main organization and environment. The *Environment* concept encompasses the instances of all regulated environments representing the system's environment regulatory context. Each environment instance has associations with its norms and owner environment (the environment it belongs to). The *Norm* concept encompasses the instances of all regulated actions' norms and it can be specialized into the *Permission*, *Obligation* and *Prohibition* sub-concepts. The *Action* concept encompasses the instances of all regulated actions from a DynaCROM MAS.

The interaction regulatory context should be described in the DynaCROM ontology by using a new Norm concept linking two Role concepts. This solution follows the representation pattern presented in [27] which defines that the relation object itself must

[1] For readability purposes, ontology data is presented graphically by using the Ontoviz graph plug-in [28].

be represented by a created concept linking the other concepts from the relation (i.e. reification of relationship). For instance, suppose an obligation norm for regulating payments when deals are done between sellers and customers. This norm can be represented by a new sub-concept, called for example "*ObligationToPay*", for linking the seller and customer *Role* sub-concepts.

2.3 Contextual Norm Composition

After classifying and representing norms in precise levels of abstractions, contextual norms can be composed during system execution since, at any given moment an agent may be related to norms defined at one or more regulatory contexts. Compositions of related contextual norms result in sets of independent norms, in which the semantic of one norm can influence the semantic of the others. Updating the domain ontology instance of a regulated MAS and/or customizing different compositions of related contextual norms, both at runtime, provide the dynamism and flexibility necessary for regulation regarding social changes characteristic of open MASs.

DynaCROM uses rules to compose contextual norms. These rules are ontology-driven rules, i.e. they are created according to the ontology structure and they are limited by the number of related concepts to which each concept is linked. All DynaCROM predefined rules are presented in Code 1. Inputs for these rules are domain instances of the Environment, Organization and Role concepts and their outputs are compositions of related contextual norms. For instance, Rule1 (line 1 to 4) states that a given environment will have its norms composed with the norms of its owner environment. More precisely: in (4), the owner environment "?OEnv" of a given environment "?Env" is discovered; in (3), the norms "?OEnvNorms" of the owner environment "?OEnv" are discovered; and in (2), these norms are composed with the norms of the given environment.

Code 1. Rules to hierarch DynaCROM contextual norms

```
(1)   Rule1- [ruleForEnvWithOEnvNorms:
(2)           hasNorm(?Env,?OEnvNorms)
(3)             <- hasNorm(?OEnv,?OEnvNorms),
(4)                belongsTo(?Env,?OEnv)]

(5)   Rule2- [ruleForOrgWithMOrgNorms:
(6)           hasNorm(?Org,?MOrgNorms)
(7)             <- hasNorm(?MOrg,?MOrgNorms),
(8)              hasMainOrganization(?Org,?MOrg)]

(9)   Rule3- [ruleForOrgWithEnvNorms:
(10)          hasNorm(?Org,?OrgEnvNorms)
(11)            <- hasNorm(?OrgEnv,?OrgEnvNorms),
(12)               isIn(?Org,?OrgEnv)]

(13) Rule4- [ruleForRoleWithOrgNorms:
(14)          hasNorm(?Role,?OrgNorms)
(15)            <- hasNorm(?Org,?OrgNorms),
(16)               isPlayedIn(?Role,?Org)]
```

Following the same composition process, Rule2 (lines 5-8) states that a given organization will have its norms composed with the norms of its main organization; Rule3 (lines 9-12) states that a given organization will have its norms composed with the norms of its environment; and Rule4 (lines 13-16) states that a given role will have its norms composed with the norms of its organization.

For continuously supporting agents with updated norm information, DynaCROM has an inference rule engine that executes the following tasks: (i) read an ontology instance to get data (i.e., concept instances and relationships), (ii) read active rules to get how concepts must be composed, and (iii) infer an ontology instance based on the previous readings. DynaCROM is currently implemented as an active JADE [21] behavior, so, this process continuously executes resulting always in updated information. Once the ontology instance and/or active rules are changed, this information is automatically forwarded to agents in the next DynaCROM execution. An overview of the norm composition process is illustrated in Fig. 2.

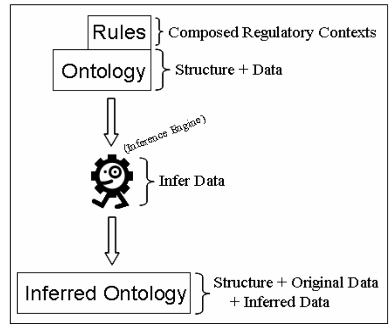

Fig. 2. The DynaCROM norm composition process.

3. CONTEXTUAL NORM ENFORCEMENT

DynaCROM is an approach for implementing dynamic normative open MASs, in which norms can be updated at system runtime, and also for continuously supporting norm-aware agents with precise information. Nevertheless, the ultimate goal of a regulated MAS is to have an enforcement mechanism for continuously verify if a performed action is legal or illegal based on its defined norms. Therefore, DynaCROM was enhanced with SCAAR for enforcing its contextual norms.

SCAAR is a norm enforcement mechanism that enhances agents with a self-monitoring capability for avoiding norm violation. SCAAR automatically adds control hooks and an enforcement core inside agent codes when agents add the DynaCROM behavior. Norm-aware agents, seeking to receive updated system norms for being able to execute correctly in a DynaCROM MAS, will spontaneously incorporate DynaCROM and, consequently, permit the SCAAR additions.

SCAAR control hooks can be inserted in agents' code before a regulated action, for preventing norm violation, or after, for detecting norm violation. Control hooks inform the agents' enforcement core about the execution of a regulated action each time it occurs and, then, the enforcement core verifies if the action is executing according to its norms.

If a system developer decides to use the SCAAR norm prevention mechanism in his regulated MAS, then, when a tentative of violation happens with an obligation or prohibition norm, the enforcement core blocks the execution of the infringing action and informs it to DynaCROM; if a system developer decides to only use the SCAAR norm detection mechanism, then, when a norm violation happens with an obligation or prohibition norm, the enforcement core informs it to DynaCROM. For a permitted norm, no specific action is taken by SCAAR.

Norms are represented by Petri nets [26] for permitting the verification of norm compliance and inhibitor arcs are used for permitting the norm enforcement. When a regulated action occurs, its specific agent's control hook activates the Petri nets that represents the norms of the action. Then, each inhibitor arc of an active Petri net is analyzed for verifying if a token stands in its previous place and, if it is the case, it forwards an exception to the agent's enforcement core because the norm was violated. The pseudo algorithm for the SCAAR norm enforcement mechanism is presented in Code 2.

Code 2. SCAAR pseudo algorithm for norm enforcement

(1) Let I be the information received about an agent behavior;
(2) Let $\{t_1, ..., t_n\}$ be the set of transitions associated with I;
(3) Let $\{P_1, ..., P_m\}$ be the set of Petri nets associated with I;
(4) Let $\{Pact_1, ..., Pact_p\}$ be the set of activated Petri nets associated with I;
(5) Let t_{ij} be the transition i of a Petri net j;
(6) **for all** $P_k \in \{P_1, ..., P_m\}$ with $t_{1k} \in \{t_1, ..., t_n\}$ **do**
(7) $Pact_{(p+1)} \leftarrow$ activate P_k if it is not already activated;
(8) add $Pact_{(p+1)}$ in $\{Pact_1, ..., Pact_p\}$
(9) **end for**
(10) Let $\{Pact_1, ..., Pact_l\}$ be the set of the activated Petri nets including $t_{ij} \in \{t_1, ..., t_n\}$;
(11) **for all** $Pact_j \in \{Pact_1, ..., Pact_l\}$ **do**
(11) **for all** $t_{ij} \in \{t_1, ..., t_n\}$ **do**
(12) activates the transition t_{ij} from $Pact_j$;
(13) **if** (t_{ij} is fireable) **then**
(14) fire the transition t_{ij};
(15) remove t_{ij} from $\{t_1, ..., t_n\}$;
(16) **else**
(17) throw an exception
(18) **end for**
(19) **if** ($Pact_j$ is in a final state) **then**
(20) remove $Pact_j$ from $\{Pact_1, ..., Pact_l\}$;
(21) **end for**

4. DynaCROM at WORK

4.1 Setting the Stage

The FIPA-Contract-Net interaction protocol [14] and the TAC-SCM competition [6] were considered in a simplification of a realistic example in order to illustrate the use of the DynaCROM approach. In the example, agents can play a manufacturer or a supplier role according to the following motivating scenario:

1. An American manufacturer wants to build a computer;
2. He issues a call for proposal (CFP) to suppliers;
3. Suppliers answer the CFP with their proposed prices;
4. The American manufacturer chooses a proposal and informs his decision to the chosen supplier.

To build a computer, the following four component types are necessary: CPU, motherboard, memory and hard disk. There are at least two suppliers for each component type with the base prices of their products predefined, as illustrated in Table 1.

The four suppliers from the example (Pintel, IMD, Macrostar and Basus) were spread through different environments (i.e., countries and states), as presented in Table 2, for illustrating DynaCROM contextual norms. Basus and Macrostar also are multinational organizations. A multinational organization is an enterprise that manages production branches located in at least two countries, which can also be across multiple continents. Corporate governance includes regulation of all possible relationships among the many players involved. The domain of multinational organizations was chosen because it well illustrates important implicit contextual information that can be found in MASs.

Table 1. Computer components' information

Description	Base price (USD)	Supplier
Pintel CPU	750	Pintel
IMD CPU	650	IMD
Pintel Motherboard	350	Macrostar
IMD Motherboard	300	Basus
Memory 2 GB	150	Macrostar
Memory 2 GB	100	Basus
Hard disk 500 GB	200	Macrostar
Hard disk 500 GB	150	Basus

Table 2. Multinational supplier organizations

Organization	Country	State
Pintel	USA	Missouri
IMD	USA	Virginia
Basus	Japan	Osaka
BasusUSA	USA	California
Macrostar	China	Shanghai
MacrostarJapan	Japan	Hiroshima

4.2 Classifying Domain Contextual Norms

Usually, organizations do not make their norms public, thus, we created contextual norms for the multinational organizations' domain and organized them in the contexts in which they apply.

4.2.1 Environment Norms

Environment Norm for Payments: In all countries, negotiations are obliged to be paid in their national currency. Negotiations outside a country are obliged to have their values converted to the national currency of the seller's country. *Contextual Environment Norms for Payments*: All negotiations are obliged to be paid (a) in *USA*, with *American dollars (USD)*; (b) in *Japan*, with *Yen*; and (c) in *China*, with *Chinese Yuan (CNY)*.

Environment Norm for Calculating Prices: In North America, a finished good from every organization is obliged to have its price increased by a fixed percentage (dependent of the seller location) as taxes, for immediate delivery or if the deliver address is in North America. *Contextual Environment Norms for Calculating Prices*: (a) In *California*, a state corporate income tax rate of *8.84* is obliged to be imposed on all sales; (b) In *Virginia*, a state corporate income tax rate of *6.00* is obliged to be imposed on all sales; (c) In *Missouri*, a state corporate income tax rate of *6.25* is obliged to be imposed on all sales; and (d) In *Missouri*, a *three day sales tax holiday* occurs, every year, from the first Friday in August until midnight on the Sunday following. Orders of computers and computers' components, with the maximum cost of $3,500, are eligible for tax free during the holiday season.

4.2.2 Organization Norms

Organization Norm for Providing Warranty: Organizations are obliged to give a limited lifetime warranty. *Contextual Organiza-*

tion Norms for Providing Warranty: (a) *Basus organizations* are obliged to give *one year* limited lifetime warranty; (b) *Macrostar organizations* are obliged to give *six months* limited lifetime warranty; and (c) *MacrostarJapan organizations* are permitted to make an offer of *two years* limited lifetime warranty if a plus tax of 5% is accepted to be paid.

Organization Norm for Deliveries: Organizations are prohibited from delivering orders during holidays to their final destinations. *Contextual Organization Norm for Deliveries*: (a) *BasusUSA organizations* are prohibited from delivering orders during holidays to their final destinations.

4.2.3 Role Norms

Role Norm for Providing Discounts: Suppliers are permitted to give up to a limited percentage of discounts. *Contextual Role Norm for Providing Discounts*: (a) *IMD suppliers* are permitted to give up to *10% discount* on orders paid in cash.

Role Norm for Accepting Placed Orders: Suppliers are obliged to request a down payment for accepting placed orders. *Contextual Role Norm for Accepting Placed Orders*: (a) *IMD suppliers* are obliged to request a down payment of *10%* for accepting placed orders.

4.2.4 Interaction Norms

Interaction Norm for Providing Discounts: Suppliers are permitted to give up to a limited percentage of discounts if their products are bought in bundles. *Contextual Interaction Norm for Providing Discounts*: (a) *Pintel* and *Macrostar* suppliers are permitted to offer 15% discount if their products are bought in bundles.

4.3 Representing and Composing Domain Contextual Norms

DynaCROM explicitly represents its domain contextual norms in an ontology instance and uses rules to compose them. For instance, Fig. 3. illustrates part of the DynaCROM domain ontology extended and instantiated to represent the contextual role norm for accepting placed orders of our example.

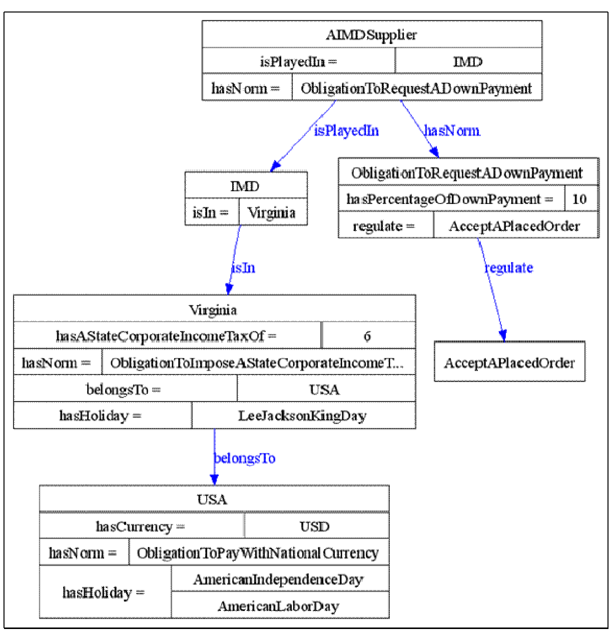

Fig. 3. A contextual role norm for accepting placed orders.

The "ObligationToRequestADownPayment" norm instance directly regulates the "AcceptAPlacedOrder" action instance. This norm is also composed with the environment norms "ObligationToImposeAStateCorporateIncomeTax" (inherited from Virginia) and "ObligationToPayWithNationalCurrency" (inherited from "USA") according to Rule1, Rule3 and Rule4 from Code 1. More precisely: according to Rule1, in (4), USA is discovered as the owner environment of Virginia; in (3), the "ObligationToPayWithNationalCurrency" environment norm of USA is discovered; and in (2), this norm is composed with the "ObligationToImposeAStateCorporateIncomeTax" environment norm of Virginia. According to Rule3, in (12), Virginia is discovered as the environment of IMD; in (11), the environment norms of Virginia are discovered; and in (10), these norms are added as IMD norms. According to Rule4, in (16), IMD is discovered as the organization of the "AIMDSupplier" role; in (15), the norms of IMD are discovered; and in (14), these norms are composed with the "ObligationToRequestADownPayment" role norm of the "AIMDSuppier".

4.4 Reasoning in a DynaCROM MAS

In a DynaCROM MAS, norm-aware agents are continuously supported with updated contextual norm information. Thus, they can better adapt themselves for execute correctly according to the enforcement of the system current norms. In our motivating example scenario, manufacturer agents basically can choose to buy with Pintel and Macrostar suppliers, or with IMD and Basus suppliers. Norm-aware agents are more likely to make better choices according to specific criteria because they are concerned with more precise information.

If the American manufacturer's purchase criteria is to minimize costs, then he should choose to buy the package IMD CPU and Basus components in IMD. There, the final price will be USD 1,144.80 according to the following calculation (all in USD): 650.00 (IMD CPU) + 300.00 (IMD Motherboard) + 100.00 (Memory) + 150.00 (Hard Disk) = 1,200.00 – 10% (IMD suppliers' discount for orders paid in cash) = 1,080.00 + 6.00% (Virginia's state corporate income tax). If he decides to buy in BasusUSA, then the state corporate income tax of 8,84% from California should be applied instead of the 6.00% from Virginia and the final price will be USD 1,175.47. However, there the down payment of 10% required by IMD suppliers for accepting placed orders are not anymore necessary and orders will not be delivered during holidays in their final destinations.

If the American manufacturer decide to buy during the sales tax holiday season, then it is better for him to buy the bundle Pintel/Macrostar in Pintel. There, the final price will be USD 1,090.00 according to the following calculation (all in USD): 750.00 (Pintel CPU) + 350.00 (Pintel Motherboard) + 150.00 (Memory) + 200.00 (Hard Disk) = 1,450.00 – 15% (Pintel/Macrostar bundle discount). Furthermore, he also can pay a plus tax of 5% in orders to profit from the warranty extension of MacrostarJapan and still will have the price of 1,144.50 (1,090.00 + 5% (for the Macrostar warranty extension)) lower than the final price of USD 1,144.80 from IMD.

Updates in the system norms can influence agents' decisions[2]. For instance, during the sales tax holiday season, IMD and Basus

---

[2] See [12] for more information about agents' decisions based on contextual norms, reported in a case study.

organizations can relax their norms for being more competitive with Pintel and Macrostar organizations. If they offer a bundle discount of 10% (a new norm added in the domain ontology instance), then, the IMD price will be 1,080.00 – 10% (IMD/Basus bundle discount) + 6.00% (Virginia's state tax) = 1,030.32, which is lower than the final price of USD 1,090.00 from Pintel.

IMD and Basus organizations can also restrict their norms (e.g., by removing all related norms of offering discounts from the domain ontology instance) for profiting better when USA and Chine are undergoing political crisis. During this phase, American organizations are prohibited from dealing with Chinese organizations, i.e. Pintel and Macrostar organizations (their competitors) are prohibited to deal with each other. Thus, the Pintel/Macrostar bundle, for example, cannot be offered anymore.

4.5 Enforcing Domain Contextual Norms

SCAAR norms (structure and data) are written automatically and dynamically by DynaCROM, while agents are executing, and they are based on a DynaCROM domain ontology instance. For instance, Code 3 illustrates the respective SCAAR contextual norms created to represent the norms of the ontology instance illustrated in Fig. 3 for regulating the "PayWithNationalCurrency", "ImposeAStateCorporateTax", "RequestADownPayment" and "AcceptAPlacedOrder" actions. According to the example, in the SCAAR norms the "environment" variable is instantiated with the "Virginia" value and the "role" variable is instantiated with the "AIMDSupplier" value.

Code 3. DynaCROM domain contextual norms in SCAAR.

```
(1) SCAARNorm1- [(agt: aGenericAgent)
(2)   OBLIGED(agt DO PayWithNationalCurrency
(3)     WITH environment.hasCurrency = "USD")
(4)       IF (agt BE in Environment WITH
(5)         ((environment = "USA") OR
(6)             (environment.belongsTo = "USA"))]

(7) SCAARNorm2- [(agt: aGenericAgent)
(8)   OBLIGED(agt DO ImposeAStateCorporateTax
(9)     WITH  environment.hasAStateCorporateIn-
                                   comeTaxOf = "6")
(10)    IF(agt BE in Environment WITH
(11)                (environment = "Virginia")]

(12) SCAARNorm3- [(agt: aGenericAgent)
(13)   OBLIGED(agt DO RequestADownPayment
(14)    WITH  norm.hasPercentageOfDownPayment =
                                              "10")
(15)    BEFORE agt DO AcceptAPlacedOrder
(16)    AFTER (agt DO PayWithNationalCurrency
(17)       WITH environment.hasCurrency = "USD"
(18)      AND agt DO ImposeAStateCorporateTax
(19)        WITH  environment.hasAStateCorpora-
                                   teIncomeTaxOf=6)
(20)      IF(agt BE in Role WITH
(21)                  (role = "AIMDSupplier"))]
```

In order to illustrate the pseudo algorithm from Code 2, SCAARNorm1 is considered. SCAARNorm1 is represented by a created Petri net, named PetriNet1, that is activated when an agent tries to perform the "PayWithNationalCurrency" action. Following Code 2, in (1), *I* = {Environment(*USA*) or Environment.belongsTo (*USA*)}; in (2), $\{t_1, ..., t_n\}$ = {*t1*}; in (3), $\{P_1, ..., P_m\}$ = {*PetriNet1*}; in (4), $\{Pact_1, ..., Pact_p\}$ = {}; in (5-9) the PetriNet1 is created for representing SCAARNorm1 (illustrated in the left side of Fig. 4); in (10), $\{Pact_1\}$ = {*PetriNet1*}; in (12) the transition *t1* is activated; in (14-15) *t1* is fireable if, and only if, its previous place is empty; in (17) *t1* is not fireable (e.g., the currency information given by the agent is equal to "*Yen*") and an exception is thrown; and in (19-21) the *PetriNet1* is removed.

SCAARNorm2 and SCAARNorm3 follows the same pseudo algorithm from Code 2. SCAARNorm2 is represented by a Petri net similar to the PetriNet1. SCAARNorm3 is represented by the PetriNet3 illustrated in the right side of Fig.4. In PetriNet3, it is assumed that the transitions from the PetriNet1 and PetriNet2 were fireable and an exception is not thrown if, and only if, the down payment percentage informed by the agent is equal to "10" and this action is performed before the "AcceptAPlacedOrder" action.

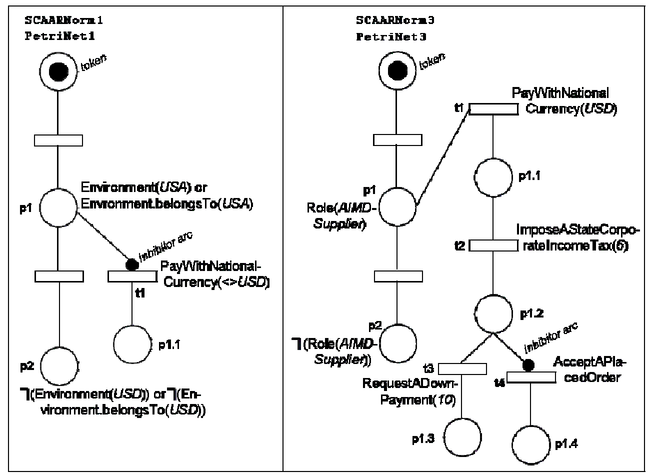

Fig. 4. Petri nets created for representing contextual norms.

5. RELATED WORK

Important works concerning regulations in the domain of MASs, such as [33], [8], [10] and [25], have been proposed recently. However, these solutions normally consider norms with a valid universal meaning in a real domain; do not support the direct design and implementation of norms specific to the application domain (e.g., political and economic norms); do not support the management of norms during system execution (norm description off-line and norm enforcement on-line); and expect that agents already have to be aware of the (*predefined*) system norms.

In order to remedy the drawbacks listed above we propose to extend the notion of normative MASs with an extra layer consisting of norms classified in contexts, i.e. we consider contextual norms as a first-order abstraction in normative MASs. For agents in a regulated DynaCROM MAS, updated norm information is continuously provided, so, they do not have to be aware of all system norms. For system developers, the use of contexts permits a more precise regulation mechanism and it also facilitates the tasks of norm design, implementation and management. For this, norms should be classified in the contexts in which they apply and represented in a domain ontology instance where concepts can be dynamically composed and data updated, both at system runtime. Thus, the use of ontologies allied with rules for composing ontology concepts provides a reasonable flexibility for norm evolution and also a meaningful way for heterogeneous agents to interpret norm information in open MASs.

Electronic Agent-Based Organizations. In [33], the OMNI framework (meaning *Organizational Model for Normative Institu-*

tions) is proposed for modeling agent organizations. OMNI focuses on the organization dimension (structuring the global behavior of the society), on the behavior of the agents from the agent perspective, on agent interactions and on a normative structure that is separate from the agents that will populate the MAS.

In order to support the development of both closed systems and open, flexible environments, OMNI presents a rigid specification of its structures defining particular fields for the description of scenes, roles and group of roles. There are no normative aspects further than the ones for organizations, roles, group of roles, agent interactions and agents (only role, scene and transition norms can be specified). The concept of organization is not present. An organization is represented by listing all its institutional roles (e.g., managers, directors, president, etc.) and it is implemented when agents play these roles.

Currently, OMNI does not provides a solution for the implementation and integration of its specifications in a given MAS. Thus, DynaCROM can provide a flexible solution for implementing agent organizations by representing the OMNI scenes, roles and group of roles in its ontology. Furthermore, this ontology also can be freely enriched with domain concepts and others particular fields for any concept. The integration of organizational data in a given MAS is transparently occurring when agents incorporate the DynaCROM behavior and, then, start automatically receiving new added data.

Electronic Agent-Based Institutions. Electronic institutions [9], or simply EIs, are agent-based institutions with their focus on the institutional aspect of organizations. The institutional aspect is further divided into normative aspects (norms that enforce behavior) and dialogical aspects (dialogic interactions). In general terms, EIs structure agent interactions by establishing the commitments, obligations and rights of participating agents. EIs can be specified and verified by using the ISLANDER [8] graphical tool and they can use the AMELI [10] agent-based middleware as an infrastructure that mediates agents' interactions while enforcing institutional norms. The combination of ISLANDER and AMELI allows to support the design and development of open MASs adopting a social perspective.

We consider as the main limitations of EIs: (i) there are no normative aspects further than the ones for roles, agent interactions and agents; (ii) their specifications are often too society-centric in the sense that it completely fix agent interactions in rigid protocols and interfaces; (iii) external agents have no room for autonomous behavior. They blindly follow defined protocols with the only autonomy to accept or reject them; (iv) all possible interactions among agents have to be defined; and (v) it is difficult, if not impossible, to describe indirect interactions. This is due to the fact that all interacting activity taking place in an EI is purely dialogic by means of direct communication between the agents.

DynaCROM can be used in AMELI by feeding *governor* agents with precise norm information according to agents' contexts, or it can be used in EIs in the place of AMELI for enforcing institutional norms. The main advantage in using DynaCROM as an EI enforcement mechanism is that the great number of messages exchanged between agents and their respective governors, and between governor agents and scene manager agents is minimized. This is because with DynaCROM each regulated agent is already enhanced with an enforcement core responsible for enforcing system norms. Yet, regulated agents supported with DynaCROM in EIs are relieved to know in advantage all the norms of the EI in which they will play.

MOSES. MOSES [25] is the middleware that supports LGI (meaning *law-governed interaction*) [24], which is a decentralized coordination and control mechanism for distributed systems. LGI enables a distributed group of actors (which may be heterogeneous, open and large) to engage in a mode of interaction governed by an explicitly specified policy called the "*law*" of this group.

Although being a well-known solution for law enforcement in distributed systems, LGI has two main limitations while enforcing norms in open MASs. The first is that LGI does not offer the support to directly enforce contextual norms; it only supports to directly enforce interaction laws. The second limitation is that LGI lacks dynamics while evolving law information. This is because, a LGI community is formed by (*LGI*) agents operating under a unique ***static*** law that must be already created when agents join it.

In order to enforce DynaCROM contextual norms by using LGI, it is necessary to decouple norm information from different levels of abstractions to the interaction level. We consider here that communicative acts, established in the interaction level, can be also viewed as organization acts, i.e. actions performed within an organization modify a fragment of social reality [29]. To give the necessary dynamics for norm enforcement, DynaCROM use its output (agents' updated contextual norms) for activating LGI predefined norms, acting as a trigger mechanism. Only norms sent by DynaCROM are enforced by LGI, even if those norms are already defined in a LGI law. This is a dynamic solution because DynaCROM outputs are based on a domain ontology instance, which normally evolves according to social changes characteristic of open MASs.

6. CONCLUSION

The motivating question of our research is how to easily implement regulation in open MASs permitting heterogeneous agents to perform efficiently and coherently. The thesis we held here is that the complexity of norm management in open MASs can be decreased by decoupling information in contexts.

Our ongoing work, named DynaCROM, intends to be a straightforward method for smoothly applying and managing regulations in open MASs as well as for enforcing precise contextual norms. DynaCROM is still a work in progress, but we agree that it already has important contributions for the domain of regulation in open MASs. DynaCROM's main contributions are: (i) a definition of a top-down classification for contextual norms, which facilitates the tasks of elucidation, organization and management of norm information; (ii) a contextual normative ontology to explicitly represent the semantic of classified norms in a meaningful way (i.e., with a common understanding) for heterogeneous agents; (iii) a definition of a norm composition process, based on ontology-driven rules, that makes it easy to update the system regulation by both evolving norms in a unique resource (an ontology) and by activating particular rules for acquiring customized compositions of contextual norms; and (iv) a solution for enforcing contextual norms.

DynaCROM is not tightly coupled with a particular enforcement mechanism. Experiments were done with SCAAR and MOSES for enforcing DynaCROM contextual norms. In SCAAR the norm enforcement is based on agents' internal behaviors and in MOSES it is based on agents' external behaviors. For both, DynaCROM works providing precise norm information as their input. For future work, we are currently implementing a solution for enabling SCAAR (implemented in SICStus Prolog [30]) to be the fully norm enforcement mechanism of DynaCROM (implemented in JAVA). We believe that SCAAR performs better than MOSES while enforcing DynaCROM contextual norms because, in SCAAR, norms can be directly enforced in any contextual level without the need to decompose them into the interaction level.

7. ACKNOWLEDGMENTS

To the ESSMA (CNPq 55.2068/2002-0) and EMACA (CAPES Cofecub 482/05/07) projects, and to the CAPES Bex 2272/07-5 and CNPq SWE 20.0846/2006-0 individual grants.

8. ADDITIONAL AUTHORS

Amal El Fallah Seghrouchni (LIP6–PARIS6, amal.elfallah@lip6.fr) and Carlos Lucena (DI–PUC-Rio, lucena@inf.puc-rio.br)

9. REFERENCES

[1] Artikis, A., Pitt, J., Sergot, M.: Animated specifications of computational societies, AAMAS, 2002, pp. 1053–1061.

[2] Boella, G.; Torre, L.; Verhagen, H.: Introduction to normative multiagent systems, In: Journal of Comput. Math. Organ. Theory, 12(2-3), 2006, pp. 71–79.

[3] Bouquet, P., Giunchiglia, F., Harmelen, F.v., Serafini, L., Stuckenschmidt, H.: C-OWL: Contextualizing Ontologies, In: Proc. of ISWC-03, LNCS 2870, 2003, pp. 164–179.

[4] Castelfranchi, C., Dignum, F., Jonker, C.M., Treur, J.: Deliberative Normative Agents: Principles and Architecture, 1999, In: Proc. of ATAL-99, 1999, pp. 364–378.

[5] Chopinaud, C., Seghrouchni, A.E.F., Taillibert, P.: Prevention of harmful behaviors within cognitive and autonomous agents, In: Proc. of ECAI'06, pp. 205–209.

[6] Collins, H.; Arunachalam, R.; Sadeh, N.; Eriksson, J.; Finne, N. and Janson, S.: The Supply Chain Management Game for the 2007 Trading Agent Competition. CMU-ISRI-07-100. 2007. Link: http://www.sics.se/tac/tac07scmspec.pdf.

[7] Dey, A.: Understanding and using context, In: Personal and Ubiquitous Computing, 5(1), 2001, 4–7, ISSN: 1617-4909.

[8] Esteva, M.; Cruz, D. de la; Sierra, C.: ISLANDER: an electronic institutions editor, AAMAS, 2002, pp. 1045-1052.

[9] Esteva, M.: Electronic Institutions: from specification to development. IIIA PhD Monography. Vol. 19, 2003.

[10] Esteva, M.; Rosell, B.; Rodriguez-Aguilar, J.A.; Arcos, J.Ll.: AMELI: An Agent-based Middleware for Electronic Institutions, In: Proc. of AAMAS, 2004, pp. 236 – 243.

[11] Felicíssimo, C.; Choren, R.; Briot, J-P.; Lucena, C.; Chopinaud, C.; Seghrouchni, A.E.F.: Providing Contextual Norm Information in Open Multi-Agent Systems. In: AOIS 06 post proceedings, LNAI Springer volume, 2007.

[12] Felicíssimo, C.; Choren, R.; Briot, J-P.; Lucena, C.: Informing Regulatory Dynamics in Open MASs. In: COIN 06 post proceedings, LNAI 4386, pp. 147–162, 2007.

[13] Ferber, J., Gutknecht, O., Michael, F.: From Agents to Organizations: an Organizational View of Multi-agent Systems, In: Proc. of AOSE, 2003, LNCS 2935, pp. 214–230.

[14] FIPA Contract Net Interaction Protocol Specification, 2003, Link: http://www.fipa.org/specs/fipa00029/SC00029H.html.

[15] Fornara, N., Viganò, F., Colombetti, M.: Agent communication and artificial institutions, JAAMAS, 2007, 14:121-142.

[16] Gaertner, D., Clark, K., Sergot, M.: Ballroom etiquette: a case study for norm-governed multi-agent systems. In: COIN@AAMAS-2006, 2006, Japan.

[17] Grizard, A., Vercouter, L., Stratulat, T., Muller, G.: A peer-to-peer normative system to achieve social order. In: COIN@AAMAS-2006, 2006, Japan.

[18] Gruber, T. R.: A translation approach to portable ontology specifications. In: Knowledge Acquisition, 5 (2), 1993, pp. 199–220, ISSN: 1042-8143.

[19] Henricksen, K., Indulska, J.: Developing context-aware pervasive computing applications: models and approach. In: Pervasive and Mobile Computing, Press, Elsevier, 2005.

[20] Hewitt, C.: Open Information Systems Semantics for Distributed Artificial Intelligence, In: Artificial Intelligence, 47(1-3), 1991, pp. 79–106, ISSN: 0004-3702.

[21] JADE, 2007, Link: http://jade.tilab.com.

[22] Jennings, N. R.: On Agent-Based Software Engineering, In: Artificial Intelligence, 117 (2) pp. 277-296, 2000.

[23] Khedr, M, Karmouch, A.: ACAI: Agent-Based Context-aware Infrastructure for Spontaneous Applications, Journal of Network & Computer Applications, 28(1), 1995, 19–44.

[24] Minsky, N.: Law Governed Interaction (LGI), Link: http://www.cs.rutgers. edu/~minsky/papers/manual.pdf.

[25] Minsky, N.: MOSES, 2007, http://www.moses.rutgers.edu/.

[26] Murata, T.: Petri nets: Properties, analysis and applications. In: IEEE 77(4), 1989, pp. 541–580.

[27] Noy, N; Rector, A. (Eds.).: Defining N-ary Relations on the Semantic Web: Use with Individuals, 2007, Link: http://www.w3.org/TR/swbp-n-aryRelations.

[28] Ontoviz. http://protege.cim3.net/cgi-bin/wiki.pl?OntoViz.

[29] Searle, R.: The construction of social reality. NY Free Press.

[30] SICStus Prolog, http://www.sics.se/isl/sicstuswww/site/.

[31] Silva, V.T.da.: From a conceptual framework for agents and objects to a multi-agent system modeling language, In: JAAMAS, 2004, 9(1-2), pp. 145–189, ISSN 1387-2532.

[32] Thomas, G., Williams, A.B.: Roles in the Context of Multi-agent Task Relationships. In: AAAI Fall Symposium TR FS-05-08, ISBN 978-1-57735-254-9, 2005.

[33] Vázquez-Salceda, J.; Dignum, V.; Dignum, F.: Organizing Multiagent Systems, In: JAAMAS, 2005, 11(3), pp. 307–360.

[34] Weyns, D.; Omicini, A.; Odell, J.: Environment as a first class abstraction in multiagent systems, In: JAAMAS, 2007, 14(1), pp. 5–30, ISSN 1387-2532.