Formalizing Theatrical Performances Using Multi-Agent Organizations

Andreas Schmidt JENSEN [1], Johannes Svante SPURKELAND and Jørgen VILLADSEN
Technical University of Denmark, Kgs. Lyngby, Denmark

Abstract. Theatrical performances usually follow strict scripts and actors are not allowed to deviate. A Danish theatrical group, Theater 770° Celsius, has invented a new method called *In Real Life*, in which only certain events in the storyline are specified and the actors are supposed to improvise to reach these events. The method bears a resemblance to multi-agent systems and we show how it can be formalized using the multi-agent organizational model OperA.

Keywords. multi-agent systems, organizations, theatrical performance

1. Introduction

The Danish theater group Theater 770° Celsius aims to renew the way theater is made using a method called *In Real Life* (IRL) [1]. Performances following the IRL-method have no strict script, but only a few basic parts of the storyline are predetermined, and the actors are supposed to improvise in order to reach these parts. The method is supposed to make it possible to watch a performance several times, without it becoming predictable, since every play through will be different.

In this paper, we take the first steps toward simulating theatrical performances that uses the IRL-method. Since the actors in a performance are human actors, one of the aims is to be able to simulate their behavior in order to figure out the directions such performance can take, while still following the basic storyline. This requires modeling emotions (to make the actors believable), improvisation, and interaction and communication between the actors. By considering the actors as agents with the objective of following the storyline, it is in principle possible to simulate a performance. Such agents should be intelligent, meaning that they should be reactive (to handle dynamic situations), proactive (to pursue the objectives of their character), social (to communicate and interact with other agents/actors), and autonomous (to improvise).

Defining the actors in terms of intelligent agents does not guarantee that they follow the basic storyline, and we argue in this paper that by imposing an *organizational model* upon the agents, we can make such assumptions about the agents in the system. Organizational models for multi-agent systems are used to distinguish the aim of the system (the theatrical performance) from the agents (the actors) in terms of objectives, roles,

[1]Corresponding Author: Andreas Schmidt Jensen, DTU Compute, Matematiktorvet, Building 303B, 2800 Kgs. Lyngby, Denmark; E-mail: ascje@dtu.dk.

groups and norms. The agents enact roles that impose certain restrictions and provide the agents with new capabilities to ensure that organizational objectives are achieved. The idea is to consider actors as agents and characters in a performance as roles, such that an agent enacting a role in the organizational model of a theatrical performance will follow the storyline, but do it using its own capabilities, meaning that different agents will enact the same role differently.

We argue that simulations of improvised performances are useful, not only for the director and for the actors of a play to be able to simulate different storylines of the play, but in other areas as well, such as interactive storytelling, where agents are characters in a story that can be influenced by the user experiencing it. Certain events in the story must happen, no matter how the user influences it, which requires that the agents are able to adapt to the influences and while steering the story in the desired direction.

We divide the simulation of theatrical performances into two distinct parts: (1) defining and maintaining the boundaries of the concept (the basic storyline) and (2) acting (e.g. making a character believable, showing emotions) and improvising (staying in character while doing "unexpected" things and still reaching the endpoint). In this paper, we focus on (1) by providing a formalization of a specific theatrical performance and aim to investigate (2) in the future. We use the OperA organizational model [2] for formalizing the theatrical performance, because it, as will be explained in section 4, fits quite well with the ideas of the IRL-method.

The rest of the paper is organized as follows. In section 2, we describe the theatrical performance that is formalized. In section 3, we describe the organizational model used for the formalization, OperA. We formalize the performance in section 4. We discuss related work in section 5 and conclude the paper in section 6.

2. The IRL-method: Theater 770° Celsius

Theater 770° Celsius is a non-profit organization with the aims to renew the way theater is made. It wants to differentiate from classical theater, where actors follow a strict script and the storyline is the same every time. It attempts to do so by means of a method called *In Real Life* (IRL) [1].

The IRL-method attempts to part from conventional theater where actors follow a predefined script and practice until they can perform it perfectly. In the process of perfecting a predefined script, one may neglect deviations and other, more interesting scenarios, making it more likely that the story becomes predictable.

The basic principle of the IRL-method is to consider a theatrical performance as a *self-organizing critical system*. Instead of having a script, which defines the entire play, the actors are given a character and a basic conflict. The actors are put together and will act according to their character, and will dynamically create the storyline of the performance in that way. The storyline is not completely free, but revolves around the basic conflict and has a few fixed events that must take place. The performance can be considered a sandbox in which the actors develop their character according to the fixed events, the conflict and the interaction with the other actors.

2.1. Case study: Win-Win: Vi elsker penge

We focus specifically on the performance called *Win-Win: Vi elsker penge* (Danish for *Win-Win: We love money*) and briefly describe its general script. There are five actors in total, but during a performance, only four of them take part. We refer to the characters by their names: Anna, Eva, Ditlev, Per and Steffen. The setting is an airport in which a suitcase full of money has been misplaced. Each actor has a suitcase of their own, and one of these suitcases is – unknowingly – the suitcase full of money. Each character has a basic conflict revolving around money, and a suitcase full of money would help them resolving this conflict. There is, among others, an eccentric stockbroker, a worker barely making a living and a businessperson close to being bankrupt, each having reasons for wanting a suitcase full of money. To avoid having a strict unfolding of the storyline only few events are fixed. The performance consists of four acts. Each of these acts has a specific plot related to the suitcase but otherwise no manuscript. The storyline is as follows:

Act 1 The characters and a few members of the audience are standing in a line. They grow impatient and the line scatters. The actors walk around the airport behaving in accordance with their character. At some point, each character has a flashback, which gives the audience an understanding of the character's personality. The act ends when all characters are present in the same location at the same time, a so-called "all-in" situation. One of the characters will have found out that he has a suitcase full of money, but it is mistakenly taken by another character.

Act 2 Two more characters will realize that the suitcase is full of money. The character initially carrying the suitcase will hold on to it until the end of the act, though it may change hands for shorter periods. The act ends when this character finds out about the money and the last character, ignorant of the money, takes the suitcase. Each character has a flashback.

Act 3 Everyone except the character carrying the suitcase is aware of its contents. Initially, the three "aware" characters follow the ignorant one and they end up in a line. Each character has a flashback, and the act ends in an "all-in" situation, when the ignorant person discovers the contents of the suitcase. This time, however, an audience member will take the suitcase and the actors will chase after him, exiting the stage.

Act 4 The characters are tied up because they chased the thief into a restricted area. The act is hectic, with the actors are bumping into each other while chasing the suitcase, and only saying few, short things. The act ends when each character argues about the money and all but one realize that perhaps it is not the true provider of happiness.

Flashbacks for a character are brought into play by another actor, which in that situation enacts the role of a person from an earlier point in the character's life. Furthermore, during the entire play, the actors will interact with the audience members by asking them questions and using their answers in the story.

3. Organizational Modeling: The OperA Model

We use the OperA model to specify the organization. The OperA model is a highly expressive model with a logical foundation, designed such that agents' desires and goals

are distinguishable from the organizational aims. That is, the agents are autonomous entities with personal goals and the characteristics of the organization are not dependent on the agents' desires. As such the OperA model provides means of defining the agents and the organization independently and then create the links between them. The link between agents and the organization is made via *roles*. The roles describe the organization's view of the individuals whereas the agent describes each individual's own view. An OperA model consists of three parts: an organizational model (OM), a social model (SM), and an interaction model (IM). The OM describes the organizational structure and objectives in terms of groups, roles, norms and scenes. The SM describes how agents join the organization by enacting roles, and by agreeing on social contracts that provide the expectations on the behavior of the agents. The IM specifies the interaction agreement between the agents in the organization in terms of interaction contracts.

In this paper, we focus on the OM, specifically on roles, objectives and scenes, and in the remainder of this section, we describe the relevant concepts of the OM in terms of the theatrical performance.

3.1. Organizational Model

The OM of OperA consists of four *structures*; the social, interaction, normative and communicative structures. In the following, we describe each structure in more detail.

3.1.1. Social Structure

The social structure specifies the roles in the organization, along with their objectives, rights, norms and the dependency between roles. Roles are used to describe the social activities needed to achieve the organizational aims by abstracting away from the individuals that are supposed to achieve them.

Objectives are further refined by sub-objectives and the distribution of objectives is defined by a role hierarchy, which describes how roles depend on each other for the completion of objectives. A dependency relation of objective o is written $r_1 \succeq_o r_2$. For example, each character depends on someone from their past when experiencing a flashback, e.g. $Anna \succeq_{flashback} Mother$ means that Anna depends on her mother for having a flashback.

3.1.2. Interaction Structure

The interaction structure specifies how the objectives of the roles are achieved by the enacting agents. The notion of scenes and scene scripts are introduced to model this interaction. A scene represents an activity, which follows an abstract scene script. A scene script contains the roles in the scene, the norms imposed on the interaction, and the desired results. The scenes are connected via scene transitions, which specify a partial ordering of the scenes (act 1 is before act 2), and the condition under which a scene can start or can end. Role evolution defines how roles evolve as agents enacting them move from scene to scene. A role r_1 evolves into another role r_2 when a scene ends and a certain condition holds. We distinguish between *necessary* (evolution *must* happen given the condition) and *sufficient* evolutions (evolution *can* happen given the condition). For instance, a character realizing that the suitcase is full of money will evolve into a new role, which has the objective of obtaining the suitcase. Two roles conflict, denoted $r_1 \otimes r_2$, if they cannot be enacted by the same agent simultaneously.

Each scene has interaction patterns defining how each of the results should be achieved. An interaction pattern is a partial ordering of the objectives of the scene, providing the agents with information about the order in which the objectives are expected to be achieved. We describe the objectives in the patterns below informally to make them more easily comprehended, and furthermore simplify such that there is a single pattern for a scene (corresponding to the storyline of the act), which ensures the achievement of the scene results.

3.1.3. Normative Structure

The normative structure provides the agents participating in the organization with a way to *trust* each other. This is accomplished by introducing norms, which are expectations of the agents enacting a role. That is, by imposing norms upon the agents, other agents may assume that those norms are not violated. In OperA, there is a distinction between role norms (rules of behavior for enacting a role), scene norms (rules of behavior of agents in a scene) and transition norms (limitations to agents following a transition between scenes). For example, a role norm for audience members might be to answer actors, when they ask a question.

3.1.4. Communicative Structure

The performance is very much based on communication between the actors, and it is therefore important that they *understand* each other, i.e. they need a common ontology. The aim of the communicative structure is to describe communication primitives, such that the agents can communicate using a common ontology. The structure thus provides both the language for communication and the contents (or ontology). Thus, it is possible for the agents to ask each other questions ("do you know what is in your suitcase?") or to inform each other about their beliefs ("my suitcase contains clothes").

4. Formalization

Consider a theatrical performance as an OperA model. Then each character is a role in the model with different objectives, modeling the behavior of that character, and each act is defined by a scene script, having results such as "One character knows about the contents of the suitcase", and a basic storyline, the interaction pattern, which must be followed. In the following section we describe how *Win-Win* can be formalized using OperA. Due to space limitations, we have omitted norms and rights from the formalization.

The agents of the system are the actors and the people in the audience, since they also play a role in the performance, $Ag = Actors \cup Audience$. For each character in the play, we consider two roles: one for the character, and one for the actor performing the *flashback* together with the character, e.g. *Anna* and *Flashback-Anna*. Furthermore, people in the audience enact the *AudienceMember* role and the characters either enact the *Ignorant* or *Aware* role, depending on whether they know the contents of the suitcase. The agent enacting the role of, for example, *Per* is not allowed to enact the role of *Flashback-Per*: $Per \otimes Flashback\text{-}Per$.

In the formalization, we focus on the suitcase containing the money, refering to it as *the suitcase*. Therefore, the predicate $holding(a, suitcase)$ means that agent a carries the suitcase containing the money. We let $\exists!$ mean "there exists exactly one" and $\exists_{=n}$ "there exists exactly n". We write $bel(a, X)$ for "agent a believes that X".

4.1. Character objectives

In a theater performance, it is an objective of the actors to make their character believable, that is, the audience should experience and understand the characters' behavior. Each character has its own personality, thus an agent enacting the role of a character should adopt this personality to make the character believable. We formalize personality traits as role objectives and norms. Agents can commit to bringing about such objectives and thereby showing the personality of their character, while they, by following their norms, will stay in character. For example, a stockbroker might talk a lot about trading, stocks and try to sell products to other characters, and have norms such as not lending money to people, while a poor character will look for job offers and try to borrow money. We are not considering the agent's own objectives, such as having a satisfied audience and a fluent performance, since we focus on a formalization of the theater, not the actors, but assume that each actor has such objectives. Table 1 shows the role specification of the stockbroker, Per, and the *Ignorant* and *Aware* roles.

Table 1. Role description of the character *Per*, and the *Ignorant* and *Aware* roles.

Id	*Per*	*Ignorant*	*Aware*
Objectives	Sell products Suggest investments	Have own suitcase	Have suitcase Keep away from others
Sub-objectives	Provide stock tips Hand out business cards Describe product Negotiate price	Stay near suitcase Hold suitcase	Swap suitcases Check suitcase contents Avoid others

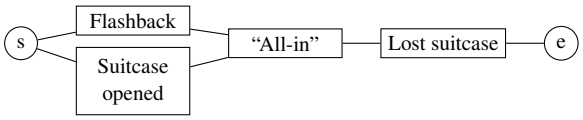
We have already discussed one kind of role dependency: the dependency between characters and their flashback character. Another kind of dependency exists between the characters and the audience. Consider, for example, act 3 in which an audience member is required to grab the suitcase and run away with it before the act ends. We therefore have that $char \succeq_{suitcase\ in\ audience} AudienceMember$. However, before an audience member can take the suitcase, the actor holding the suitcase needs to put it on the ground, so that the audience member can take it, thus $AudienceMember \succeq_{drop\ suitcase} char$. This is in conflict with the role objective of keeping others away from the suitcase, and the agents need to be able to handle that.

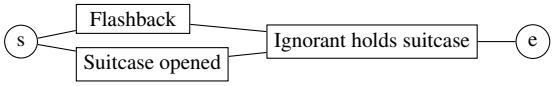
As noted, characters are not allowed to play their corresponding flashback-character, and furthermore, audience members are not allowed to enact any other roles, that is, $\forall r \in Roles\ (AudienceMember \otimes r)$, where *Roles* is the set of all roles.

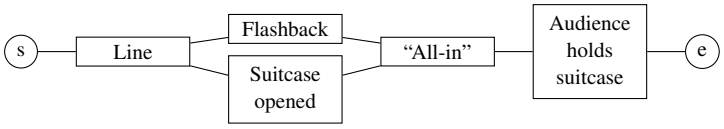
4.2. Formalizing acts

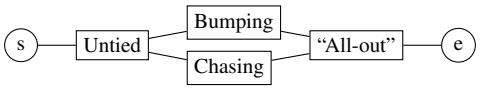
When starting the first act, the audience will enact the *AudienceMember* role and the four characters will be chosen. Table 2 shows the scene script for each of the acts, when four characters have been chosen. Result r_1 of act 1 states that one of the actors should know (i.e. its belief base should contain) that one of the suitcases is full of money, and that agent is not carrying the suitcase. r_2 states that all the actors are in the same location. The interaction pattern describes the situations that *must* happen during an act. For instance, in act 1 everybody must have had a flashback, and one agent must have discovered the contents of the suitcase *before* the all-in situation.

Table 2. Scene script for each of the four acts. Note that *char* is shorthand for the four characters in the current performance.

Scene	Act 1
Roles	*char*, Flashback-*char*, AudienceMember, Ignorant
Results	$r_1 = \exists!a \in Actors\ (bel(a, contains(suitcase, money)) \wedge \neg holding(a, suitcase))$ $r_2 = \forall a \in Actors\ (in(a, l))$
Interaction pattern	

Scene	Act 2
Roles	*char*, Flashback-*char*, AudienceMember, Ignorant, Aware
Results	$r_1 = \exists_{=3}a \in Actors\ (bel(a, contains(suitcase, money)))$ $r_2 = \exists!a \in Actors\ (\neg bel(a, contains(suitcase, money)) \wedge holding(a, suitcase))$
Interaction pattern	

Scene	Act 3
Roles	*char*, Flashback-*char*, AudienceMember, Ignorant, Aware
Results	$r_1 = \forall a \in Actors\ (bel(a, contains(suitcase, money)) \wedge \neg holding(a, suitcase))$ $r_2 = \exists!a \in Audience\ (holding(a, suitcase))$
Interaction pattern	

Scene	Act 4
Roles	*char*, AudienceMember, Aware
Results	$r_1 = \exists!a \in Actors\ (bel(a, money = happiness))$
Interaction pattern	

4.3. Managing scene transitions

Each act has a *final* situation in which the act ends. This is modeled by the *results* of the scene script of each act. When all results of an act have been satisfied, the scene representing that act will end, possibly leading to a new scene. For example, act 1 ends when one agent knows about the contents of the suitcase and has lost it, and all the agents are in an all-in situation. A transition to the next scene will be fired, if the conditions for starting that scene are satisfied, and a number of role evolutions will take place.

There will usually be a short break between scenes in theatrical performances to make changes to the stage, change outfits, etc., and this can be modeled in OperA by introducing *intermediary scenes*, which have as precondition the previous act's results and as result the next act's preconditions. For example, between act 3 and 4 the actors will be tied up in the middle of the stage and since this is not part of the results of act

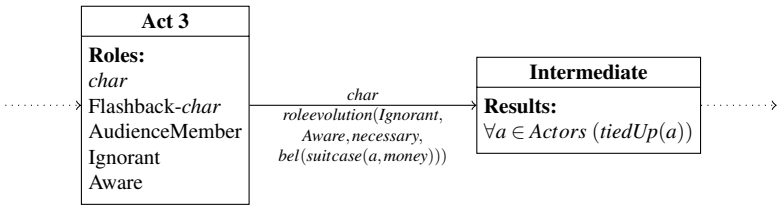


Figure 1. The scene transitions between act 3 and an intermediary scene, which prepares act 4. *char* is shorthand for all the characters in the scene.

3, it will be done in an intermediary scene. Figure 1 illustrates the transition between act 3 and an intermediary scene. Each character has a trivial role evolution between the scenes, and *Ignorant* agents are made *Aware* if they believe the suitcase contains money. The *AudienceMember* role is not part of the intermediary scene, but will be present again in act 4.

4.4. Reasoning about the performance

We briefly describe how to specify the performance formally using predicates, such that it can be used by different agent programming languages, such as GOAL [3]. We define a predicate for each relation in the OM. Table 3 lists the predicates.

The specification of each role is then straightforward, for example, the role of *Per* can be specified as follows:

$$role(\ Per, \{sell\_products, suggest\_investments\}, \{provide\_stock\_tips, \\ hand\_out\_business\_cards, describe\_product, negotiate\_price\}),$$

that is, the role is described in terms of its id and objectives. Act 4 is therefore simply $script(Act4, \{char, AudienceMember, Aware\}, \{r_1\})$, and its interaction pattern is specified using the *order*-predicate:

$$order(s, Untied) \quad order(Untied, Bumping) \quad order(Untied, Chasing) \quad \ldots$$

The interaction pattern, being a partial order, is reflexive, antisymmetric and transitive, thus we have $order(s, e)$ and $order(Untied, \text{“}All\text{-}out\text{”})$.

Table 3. Predicates for reasoning about the performance (and OperA models in general).

Predicate	Description
$role(r, O, S_O, R, N)$	Role r with objectives O, sub-objectives S_O, rights R and norms N.
$dependency(r_1, r_2, o)$	Dependency between roles r_1 and r_2 for objective o.
$script(s, R, Res)$	Scene script s with roles R and results Res.
$order(o_1, o_2)$	Partial ordering of objectives. Objective o_1 must be achieved before objective o_2.
$transition(s_1, s_2, c_{pre}, c_{post})$	Scene transition from scene s_1 to scene s_2, if c_{pre} holds in s_1. After firing the transition, c_{post} holds in s_2.
$roleevolution(r, r', \tau, c)$	A role r evolves to r' if condition c holds. The evolution is of type τ, which is either *sufficient* or *necessary*.
$conflict(r_1, r_2)$	A conflict between r_1 and r_2, meaning that r_1 and r_2 cannot be simultaneously enacted by the same agent.

Given the predicates above, we can reason about the characters and the acts, for example by deciding which characters should be part of the play, and which agents will enact the roles of the flashback-characters. Note that some agents may be considered "organizational agents", which are not part of the performance itself, but ensure seamless transition between scenes and that enactment of roles happens in accordance with the specification (e.g. conflicting roles are not enacted simultaneously by the same agent), for example a gatekeeper [4].

We have identified different kinds of reasoning mechanisms that will be required for successfully simulating the theatrical performance. *Enactment reasoning* is concerned with which roles to enact and is based on capabilities, desires, role conflicts, etc. *Role and objective reasoning* is concerned with how to enact the chosen roles. *Scene reasoning* is reasoning about how to achieve the results of a scene using the interaction patterns. Finally, *scene transition reasoning* is reasoning about finishing a scene and moving to the next: which scenes are next and what kind of evolution happens during the transition? The agents should reason about whether their enactment can or will change (sufficient and necessary evolution, respectively), whether a role they enact disappears, etc. The details of these kinds of reasoning are out of scope for this paper, but will be investigated in the future.

5. Related Work

Though our focus is on theatrical performances, the problem of making characters behave believable while still achieving the goal of completing the story successfully is present in other areas of research as well. In interactive storytelling, a storyline is generated by characters taking certain actions, while fulfilling certain goals. Multi-agent systems are often used [5,6], for instance by focusing multiple planning agents, each of which choose relevant actions, while a single narrative planner ensures overall coherence between the actions and the storyline.

We have formalized the IRL-method using a specific organizational model, the OperA model. The use of scene scripts makes OperA a natural choice in this particular setting, though it should be possible to use other models, such as MOISE$^+$ [7] or ISLANDER [8]. For example, in the MOISE$^+$ model, objectives are achieved through missions in which several roles cooperate; by considering each act a mission, the objectives of the acts will be achieved by fulfilling the missions.

While we did not investigate how to model the agent-specific aspects of the formalization, such as making acting believable and improvising, others have worked toward formalizing emotions, which are important for the audience to understand and believe the characters, their background and their behavior. For example, [9] describes a modal logic with operators for the agent's beliefs, goals and moral attitude. Basic emotions are then e.g. *joy* when a goal is achieved, etc. Complex emotions are composed of the modal operators, such that the agent feels *guilty* if it believes it is responsible for something that does not correspond to its moral attitude. For instance, if one of the characters' moral attitudes is that another character deserves the money, the first character might feel guilt when it takes the suitcase from the other anyway.

6. Conclusion

The theatrical method, IRL, used by Theater 770° Celsius has been shown to bear resemblance to multi-agent systems and in particular the OperA organizational model. In this paper, we have taken the first steps toward simulating performances following the principles of IRL by formalizing concepts such as acts, characters and events into scene scripts, roles and objectives. We have furthermore argued that the formalization can be used to let agents reason about their role in an organization, allowing them to enact the characters of a performance.

While we have only considered the organizational model of OperA, we plan to investigate how to incorporate the social and interaction models in the formalization as well. This will allow a specification of the contracts between the agents and the society and among the agents, such that actors playing a character may have greater influence on the outcome of the play by changing the requirements of their role. One actor might want to change the *Aware* role to be more protective about the suitcase, for example, by using handcuffs to lock it to his own wrist.

A next step is to further investigate the components required to let the agents perform the two kinds of reasoning required; (1) the organizational reasoning and (2) the theatrical reasoning. We have taken the first steps toward addressing (1) by formalizing the specification, and it will be natural to further investigate how agents can use a formal specification for their reasoning about which role to enact, and what objectives to attempt to achieve. Theatrical reasoning is the reasoning that makes the performance *enjoyable*, and it naturally requires investigating completely different concepts, such as how to measure when something is enjoyable, however we have not investigated this yet.

References

[1] Troels Christian Jakobsen. "I wouldn't have thought of it myself" – emergence and unexpected intelligence in theater performances designed as self-organizing critical systems. In *Algolog Multi-Agent Programming Seminar 2011*, AMAPS2011, pages 3–9, Technical University of Denmark, 2011.

[2] Virginia Dignum. *A Model for Organizational Interaction: Based on Agents, Founded in Logic*. PhD thesis, Utrecht University, 2004.

[3] Koen V. Hindriks. Programming Rational Agents in GOAL. *Multi-Agent Programming: Languages, Tools and Applications*, 2:119–157, 2009.

[4] Huib Aldewereld, Virginia Dignum, Catholijn M. Jonker, and M. Birna van Riemsdijk. Agreeing on role adoption in open organisations. *KI-Künstliche Intelligenz*, 26(1):37–45, 2012.

[5] Jonathan Teutenberg and Julie Porteous. Efficient intent-based narrative generation using multiple planning agents. In *Proc. of 12th Int. Conf. on Autonomous Agents and Multi-Agent Systems*, AAMAS '13, pages 603–610, 2013.

[6] Julie Porteous, Fred Charles, and Marc Cavazza. Networking: using character relationships for interactive narrative generation. In *Proc. of 12th Int. Conf. on Autonomous Agents and Multi-Agent Systems*, AAMAS '13, pages 595–602, 2013.

[7] Jomi Fred Hübner, Jaime Simão Sichman, and Olivier Boissier. A model for the structural, functional, and deontic specification of organizations in multiagent systems. In *Proc. of 16th Brazilian Symposium on Artificial Intelligence: Advances in Artificial Intelligence*, SBIA '02, pages 118–128, 2002.

[8] Marc Esteva, David de la Cruz, and Carles Sierra. Islander: an electronic institutions editor. In *Proc. of 1st Int. Conf. on Autonomous Agents and Multi-Agent Systems*, AAMAS '02, pages 1045–1052, 2002.

[9] Nadine Guiraud, Dominique Longin, Emiliano Lorini, Sylvie Pesty, and Jérémy Rivière. The face of emotions: a logical formalization of expressive speech acts. In *Proc. of 10th Int. Conf. on Autonomous Agents and Multi-Agent Systems*, AAMAS '11, pages 1031–1038, 2011.