A Boolean Game Based Modeling of Socio-Technical Systems

Nils Bulling

Department of Informatics
Clausthal University of Technology, Germany
bulling@in.tu-clausthal.de

Abstract. In this paper we introduce a formal model of Socio-Technical Systems (STSs) which makes use of Boolean games, a popular computational framework to model cooperation among autonomous agents, to study stability of the emergent behavior in STSs. We divide a STS into informationally independent organization units which affect system stability, inter-agent communication and information confidentiality (e.g. to ensure privacy issues). We give examples and present some preliminary characterization results about the existence of incentive schemes to stabilize STSs.

1 Introduction

Socio-technical theory is concerned with the interplay between technical- and social systems and their joint optimization [22, 21]. The technical part of a *socio-technical system* (STS) represents, e.g., the technical infrastructure, the technology and available resources. The social system includes the actors, their social relations, their goals etc. As such, STSs are heterogenous, autonomous and highly dynamic; even if the technical system is fixed the social system is subject to frequent changes. Moreover, a STS can often be considered as a system of (organized) subsystems; we call them—in this paper—organization units. These units are somehow independent: Information exchange, cooperation and communication between them are often limited and they have their own organizational objectives. There can be various reasons for that, for example an insufficient technical infrastructure, time and cost constraints, competition and conflicting interests. This is a major obstacle for the design of a STS as its overall behavior emerges from the behaviors of its organization units. As a consequence, decisions and actions taken in these independent units are interrelated and need to be coordinated to obtain a desired global system behavior [20, 8, 5, 18]. This shows that designing effective, cost-efficient, stable, robust, adaptive STSs is an extremely challenging task. The research question addressed in this paper is:

> How to formally model STSs in which communication as well as cooperation is limited, and information is restrained by involved actors (e.g. due to competing interests) in order to analyze system stability?

Our model of a STS draws inspiration from Boolean games. Boolean games [10, 11, 4, 14] represent a compact, computational model of cooperation among autonomous agents. Agents control some Boolean variables—they decide on their truth values—and try to satisfy their individual goals which are given as Boolean formulae. Agents can usually not satisfy their goal formulae on their own: They often depend on the actions of other self-interested actors. This requires cooperation and strategic reasoning. A cost function further adds to the complexity of the strategic dimension: Agents try to minimize costs.

We use Boolean games to model organization units in STSs. Therefore, a STS induces a collection of Boolean games. Each game/organization unit has an organizational objective which may not be publicly known to other organization units, e.g. because of competing interests. Consequently, to achieve a good global behavior the organization units have to announce parts of their objectives—as much as is needed to obtain a good behavior, but not too much, however, to preserve confidentiality—in order to facilitate cooperation and coordination. This is similar to Boolean secrecy games [6] in which players try to satisfy their goals without revealing them to others. We consider confidentiality constraints as the goals of the organizational units greatly affect system stability. Thus, the formal model has to distinguish public and private information in order to accurately predict system stability.

The *paper is structured* as follows. First, we recall Boolean games, a minor extension of them, and solution concepts as well as incentive schemes. In Section 3 we present our formal modeling framework—the key contribution of this paper. In Section 4 we analyse the existence of stable/good STSs and give some preliminary characterization results. In Section 5 and 6 we discuss related work and conclude, respectively.

2 Constrained Boolean games and solution concepts

In this section we review the Boolean games model and introduce Boolean games with information which will later be used to model organization units.

2.1 Preliminaries: Propositional logic and Boolean games

Propositional Logic. Let Π be a set of (propositional) variables and $X \subseteq \Pi$ a non-empty subset. We use $\mathsf{PL}(X)$ to refer to the set of propositional formulae where propositional variables are drawn from X. We assume the standard logical connectives $\neg, \wedge, \vee$ and $\rightarrow$. A *X-valuation* or *X-assignment* is a function $\xi : X \rightarrow \mathbb{B}$, where $\mathbb{B} = \{\mathsf{t}, \mathsf{f}\}$ is the set of Boolean truth values, assigning a truth value to each variable in X. $\xi|_Y$ refers to the assignment which equals ξ but of which the domain is restricted to Y. We write $\xi \models \varphi$ if X-valuation ξ satisfies $\varphi \in \mathsf{PL}(Y)$ where $\emptyset \neq Y \subseteq X$. A formula φ over X is *satisfiable* (resp. *valid*) if there is an X-assignment which satisfies φ (resp. if all X-assignments satisfy φ). If clear from context we will omit mentioning the sets X and Y and assume that a valuation always defines all the variables contained in a formula. The set of all X-valuations is denoted by Val_X. Given two assignments $\xi \in \mathsf{Val}_X$ and $\xi' \in \mathsf{Val}_{X'}$ with $X \cap X' = \emptyset$ we write $\xi \circ \xi'$ to refer to the assignment on $X \cup X'$ with $(\xi \circ \xi')|_X = \xi$ and $(\xi \circ \xi')|_{X'} = \xi'$.

Boolean games. Apart from minor modifications, we follow the definition of Boolean games of [11]. A *Boolean game* is a tuple $\mathsf{G} = (\mathsf{Agt}, \Pi, c, (\gamma_i)_{i\in\mathsf{Agt}}, (\Pi_i)_{i\in\mathsf{Agt}})$ where $\mathsf{Agt} = \{a_1, \ldots, a_k\}$ is a non-empty set of agents, Π a finite, non-empty set of (propositional) variables, $\Pi_i \subseteq \Pi$ is the set of variables controlled by $i \in \mathsf{Agt}$. We require that $(\Pi_i)_{i\in\mathsf{Agt}}$ forms a partition of a subset of Π (as in [13] we do not require that all variables are controlled by some agent). $c : \Pi \times \mathbb{B} \to \mathbb{R}_+$ is a cost function and $\gamma_i \in \mathsf{PL}(\Pi)$ a propositional formula. For example, $c(p, \mathsf{t}) = 4$ models that setting variable p to t incurs costs of 4. We write $\mathsf{Ctrl}(A) = \bigcup_{i\in A} \Pi_i$ for the set of variables controlled by $A \subseteq \mathsf{Agt}$, $\Pi_0 = \Pi \backslash \mathsf{Ctrl}(\mathsf{Agt})$ for the set of *environmental variables*, and $\widehat{\Pi} = \Pi \backslash \Pi_0$ to refer to the set of variables controlled by the agents in Agt.

Example 1 (Boolean game).

(a) Let $\mathsf{G}_1 = (\mathsf{Agt}, \Pi, c, (\gamma_i)_{i\in\mathsf{Agt}}, (\Pi_i)_{i\in\mathsf{Agt}})$ where $\mathsf{Agt} = \{a_1, a_2, a_3\}$, $\Pi = \{p_1, \ldots, p_5\}$, $\gamma_{a_1} = (p_1 \wedge p_2) \vee (\neg p_1 \wedge \neg p_2)$, $\gamma_{a_2} = (\neg p_1 \wedge p_2) \vee (\neg p_2 \wedge p_1)$, $\gamma_{a_3} = p_1 \wedge p_3$. The variables are controlled as follows: $\Pi_{a_1} = \{p_1\}$, $\Pi_{a_2} = \{p_2, p_5\}$, and $\Pi_{a_3} = \{p_3, p_4\}$. Note that the game has no environmental variables. We define the cost function by $c(p, \mathsf{t}) = 1$ and $c(p, \mathsf{f}) = 0$ for all $p \in \Pi \backslash \{p_4, p_5\}$ and $c(p_4, \mathsf{t}) = c(p_4, \mathsf{f}) = c(p_5, \mathsf{t}) = c(p_5, \mathsf{f}) = 1$.

(b) Let $\mathsf{G}_{\{2,3\}}$ be the game obtained from G_1 with player a_1 removed, that is $\mathsf{G}_{\{2,3\}} = (\{a_2, a_3\}, \Pi, c, (\gamma_{a_2}, \gamma_{a_3}), (\Pi_{a_2}, \Pi_{a_3}))$. Then, variable p_1 is an environmental variable (i.e. controlled by no player).
Analogously, let $\mathsf{G}_{\{1\}}$ be the game obtained from G_1 with players a_2 and a_3 removed. The environmental variables are $\Pi \backslash \{p_1\}$.

2.2 Constrained Boolean games and information

We extend the Boolean game model with a *global constraint* on the actions of the agents.

Definition 1 (Constrained Boolean game, consistent assignment). *A* constrained Boolean game *is given by* $\mathsf{G} = (\mathsf{Agt}, \Pi, c, (\gamma_i)_{i\in\mathsf{Agt}}, (\Pi_i)_{i\in\mathsf{Agt}}, \varphi)$ *where* $\mathsf{G}' = (\mathsf{Agt}, \Pi, c, (\gamma_i)_{i\in\mathsf{Agt}}), (\Pi_i)_{i\in\mathsf{Agt}})$ *is a Boolean game and* $\varphi \in \mathsf{PL}(\Pi)$ *a propositional formula—the* global constraint*. We also write* $\mathsf{G} = (\mathsf{G}', \varphi)$*. An assignment* ξ *is said to be* φ-consistent *iff* $\xi \models \varphi$*. For obvious reasons, we will identify a Boolean game* $\hat{\mathsf{G}}$ *with the constrained Boolean game* $(\hat{\mathsf{G}}, \top)$ *and vice versa (*$\top$ *imposes no constraints on actions).*

In particular, we are interested in constrained Boolean games where φ contains environmental variables only. Such a constraint can be seen as information disclosed to the agents about the environmental variables, or as the agents' (global) belief about the environmental variables. This is similar to [13] where each agent has a belief—formally defined as a truth assignment—about the environmental variables. In our work, we assume that information is publicly announced and known to all agents. Information can be vague; therefore, we use propositional formulae rather than (partial) truth assignments as it is the case

in [13]. For example, agents may believe that $x \vee y$ but they have no specific belief about x nor about y.

Definition 2 (Boolean game with information). *A constrained Boolean game* $(\mathsf{Agt}, \Pi, c, (\gamma_i)_{i\in\mathsf{Agt}}, (\Pi_i)_{i\in\mathsf{Agt}}, \varphi)$ *is a Boolean game with information if* φ *is a propositional formula over* $\Pi \setminus \mathsf{Ctrl}(\mathsf{Agt})$.

Example 2 (Boolean game with information). The Boolean game $\mathsf{G}_{\{2,3\}}$ from Example 1 in combination with $\varphi = p_1$ is a Boolean game with information. It models that p_1 is (believed to be) true.

2.3 Solution concepts

A *solution concept* $\mathcal{SC}$ maps constrained Boolean games over Π to truth assignments such that $\mathcal{SC}(\mathsf{G}) \subseteq \mathsf{Val}_\Pi$. In the following we assume that $\mathsf{G} = (\mathsf{Agt}, \Pi, c, (\gamma_i)_{i\in\mathsf{Agt}}, (\Pi_i)_{i\in\mathsf{Agt}}, \varphi)$. Let us define max_G as $\sum_{p\in\Pi}[c(p,\mathsf{t})+c(p,\mathsf{f})]+1$; the number is greater than the maximum cost in any course of action. We lift the cost function c of a constrained Boolean game to assignments: for and X-assignment ξ we define $c(\xi) = \sum_{p\in X} c(p, \xi(p))$. Then, the *utility* of a Π-valuation ξ for player i is defined as follows (cf. [11]) where γ_i is the goal of player i:

$$\hat{\mu}_i^\mathsf{G}(\xi) = \begin{cases} \mathsf{max}_\mathsf{G} - c(\xi|_{\Pi_i}) & \text{if } \xi \models \gamma_i, \\ -c(\xi|_{\Pi_i}) & \text{else.} \end{cases}$$

The utility function $\hat{\mu}_i$ computes the utility independently of whether the constraint φ is satisfied by ξ. It is an auxiliary function and models, due to the term max_G, that a player always prefers an assignment which satisfies its goal over one that does not. Ultimately, we are interested in the *worst case utility* which is defined next. For a $\widehat{\Pi}$-valuation ξ the *worst case utility of player* i—note that it does not include environmental variables—is defined by

$$\mu_i^\mathsf{G}(\xi) = \begin{cases} \min\{\hat{\mu}_i^\mathsf{G}(\xi') \mid \xi' \in \mathsf{Val}_\Pi, \xi'|_{\widehat{\Pi}} \equiv \xi, \xi' \models \varphi\} & \text{if } \varphi[\xi] \text{ is satisfiable} \\ -\mathsf{max}_\mathsf{G} - c(\xi|_{\Pi_i}) & \text{else} \end{cases}$$

where $\varphi[\xi]$ equals φ but each propositional variable $p \in \widehat{\Pi}$ occurring in φ is replaced by $\top$ (resp. $\bot$) if $\xi(p) = \mathsf{t}$ (resp. $\xi(p) = \mathsf{f}$). The worst case utility models the worst case assignment of the environmental variables for player i where it is required, however, that the environmental variables respect the constraint φ. In this sense, φ is a global constraint which assumes some enforcement/communication mechanism.

Now, we can define standard solution concepts. For example, a *Nash equilibrium* is a $\widehat{\Pi}$-valuation $\xi = (\xi_{a_1}, \ldots, \xi_{a_k})$ where $\xi_{a_i} \in \mathsf{Val}_{\Pi_{a_i}}$ and $\mathsf{Agt} = \{a_1, \ldots, a_k\}$ such that for all $j = 1, \ldots, k$ and all $\xi'_{a_j} \in \mathsf{Val}_{\Pi_{a_j}}$ we have that

$$\mu_{a_j}^\mathsf{G}(\xi) \geq \mu_{a_j}^\mathsf{G}(\xi_{a_1}, \ldots, \xi_{a_{j-1}}, \xi'_{a_j}, \xi_{a_{j+1}}, \ldots, \xi_{a_k}).$$

Remark 1. Let G be a Boolean game such that $\widehat{\Pi} = \Pi$. Then, the Nash equilibria of G—in the classical sense as defined in [23]—are equivalent to the Nash equilibria of $(\mathsf{G}, \top)$.

Example 3 (Nash equilibria).

(a) Firstly, let us consider the Boolean game G_1 from Example 1. The game does not have any Nash equilibria: The goals of players a_1 and a_2 do not allow any stable point. We observe that for any truth value of p_1 and p_2 either player a_1's or player a_2's goal is true (but never both). Moreover, if player a_i's goal is true, $i \in \{1, 2\}$, player a_{3-i} can (by flipping the truth value of p_i) ensure that its goal becomes true and player a_i's goal false.

(b) The game $\mathsf{G}_{\{2,3\}}$ from Example 1 has the unique Nash equilibrium $\xi \in \mathsf{Val}_{\Pi_{a_2} \cup \Pi_{a_3}}$ with $\xi(p) = \mathsf{f}$ for all $p \in \Pi_{a_2} \cup \Pi_{a_3}$. This is easy to see: No player can guarantee to achieve its goal, because for any $\Pi_{a_2} \cup \Pi_{a_3}$-assignment there is a value of p_1 which makes γ_{a_2} and γ_{a_3} false (possibly not at the same time). Hence, the best/cheapest actions for players a_2 and a_3 are those that make all their variables false.

(c) The Boolean game with information $(\mathsf{G}_{\{2,3\}}, p_1)$ has the four Nash equilibria $(\mathsf{f}, \mathsf{t}, \mathsf{f}, \mathsf{f})$, $(\mathsf{f}, \mathsf{t}, \mathsf{t}, \mathsf{f})$, $(\mathsf{f}, \mathsf{t}, \mathsf{f}, \mathsf{t})$ and $(\mathsf{f}, \mathsf{t}, \mathsf{t}, \mathsf{t})$. Each tuple represents a truth assignment of p_2, p_3, p_4 and p_5 (in this order).

(d) The Boolean game with information $(\mathsf{G}_{\{1\}}, p_2)$ has the unique Nash equilibrium $\xi \in \mathsf{Val}_{\Pi_1}$ with $\xi(p_1) = \mathsf{t}$.

2.4 Incentive schemes

Nash equilibria may not exist or there may be several of them; often, both is undesirable. One way to change the behavior of agents is to use taxes or incentives. This is called *incentive engineering* and has been studied in [23] in the Boolean game setting. Here we consider incentives as payoffs given to the agents rather than taxes imposed on the actors. Formally, an *incentive scheme* for a constrained Boolean game (G, φ) over Π is a function $\iota : \widehat{\Pi} \times \mathbb{B} \to \mathbb{R}$. The interpretation of $\iota(p, \mathsf{t}) = 5$ is that setting variable p to t is incentivized by 5 (units of payoff). We denote by $\mathsf{G} \oplus \iota$ the game which equals G but has as cost function $c'(p, v) = c(p, v) - \iota(p, v)$ for all $p \in \widehat{\Pi}$ and $c'(p, v) \equiv c(p, v)$ for all $p \in \Pi_0$ where c is the original cost function of G. Similarly, we write $(\mathsf{G}, \varphi) \oplus \iota$ for $(\mathsf{G} \oplus \iota, \varphi)$.

3 Formal Modeling of Socio-Technical Systems

A *socio-technical system* (STS) is composed of two subsystems: a technical and a social one. The technical subsystem provides, e.g., the technology, resources, and the technical infrastructure. The social system represents the actors/agents, their abilities, goals, and models the interrelation between actors but also social and organizational constraints which are imposed on the agents to ensure the successful functioning of the STS. In the following we assume that Π is a non-empty set of (propositional) variables.

3.1 Formal System Model

A technical system $\mathcal{T}$ consists of a set of available artifacts (e.g. resources and machines), which are modeled by propositional variables $\Pi_{\mathcal{T}}$, and a set of *technical constraints* that affect the size and structure of a STS. These constraints are modeled by vectors $(t_1, \ldots, t_j)$ of positive integers meaning that the actors can be clustered into j *technical units* of sizes $t_1, \ldots, t_j$.

Definition 3 (Technical system). *A* technical system *(over Π) is given by a tuple $\mathcal{T} = (\Pi_{\mathcal{T}}, \{T_i\}_{i \in I}, \mathsf{tcost}_{\mathcal{T}})$ where $I \subseteq \mathbb{N}$ is a non-empty, finite index set; $\Pi_{\mathcal{T}} \subseteq \Pi$ a non-empty, finite set of variables; each $T_i = (t_1, \ldots, t_{j_i})$ is a finite, non-empty sequence of positive integers, one for each $i \in I$; and $\mathsf{tcost}_{\mathcal{T}} : I \to \mathbb{R}_+$ a* cost function. *The value $\mathsf{tcost}_{\mathcal{T}}(i)$ defines the* costs *needed to realize T_i. A vector T_i is called* technical constraint *and each t_j refers to a* technical unit.

Example 4 (Technical system). Consider the technical system $\mathcal{T}_1 = (\{p_1, \ldots, p_5\}, \{T_1, T_2\}, \mathsf{tcost}_{\mathcal{T}})$ with $T_1 = (5)$ and $T_2 = (1, 3)$, $\mathsf{tcost}_{\mathcal{T}}(1) = 1$ and $\mathsf{tcost}_{\mathcal{T}}(2) = 3$. The system models two possible infrastructures: the first consists of a single unit of size 5 and costs 1, where the second models two units of size 1 and 3, respectively, and costs 3.

A technical system defines the static structure of a STS; for example, a vector from T_i can represent the available offices in a building and their capacity or (more or less independent) distributed parts of a STS. We assume that communication and cooperation across these different technical units is limited. Next, we introduce an *agent society*. It models the available actors and their individual goals from which a STS can draw its members.

Definition 4 (Agent society). *An* agent society *(over Π) is a set $\mathcal{A} = \{(\Pi_1, \gamma_1, \mathsf{c}_1), (\Pi_2, \gamma_2, \mathsf{c}_2), \ldots\}$ where γ_i is a propositional formula over Π, $\Pi_i \subseteq \Pi$ a finite, non-empty set of variables, and $\mathsf{c}_i : \Pi_i \times \mathbb{B} \to \mathbb{R}$ a cost function of player i. Intuitively, an element $(\Pi_i, \gamma_i, \mathsf{c}_i)$ represents an agent which is capable of controlling variables Π_i, which has γ_i as individual goal and c_i as cost function.*

A social system consists of a subset of agents—drawn from an agent society—and defines their relations and powers. Our relational model between agents is rather simplistic. It prescribes how agents are divided into *organization units*. We assume that each organization unit S has an *organization objective* δ which is known to all agents in the unit but, *per se*, not to members of other units. In order to obtain an efficient overall behavior across organization units, the STS has to provide communication and cooperation mechanisms. For this purpose, each organization unit publicly and truthfully announces parts of its objective to inform the other units; therefore, we require that the announced objective and the real objective must be consistent. There are plenty of reasons why organization units belonging to the same STS may not want to reveal their true objectives; for example, they may be in competition. This is similar to the idea of Boolean secrecy games [6] where players try to hide their true goals.

Finally, agents from the same agent society are able to control specific variables. The intuition is that the agents have, e.g., the power to operate a machine or the knowledge to work with a piece of software. There can be several agents with overlapping capabilities; hence, a STS has to define which agents have the rights to exercise their powers. This is modeled by a function pow.

Definition 5 (Social system). *A* social system *over an agent society $\mathcal{A}$ (over Π) is given by $\mathcal{S} = (\mathsf{Agt}, \mathsf{pow}, (S_1, \delta_1, \delta_1^I), \ldots, (S_s, \delta_s, \delta_s^I), \iota)$ where*

- $\mathsf{Agt} \subseteq \mathcal{A}$ *is a finite, non-empty set of agents*[1]*. If $(\Pi_l, \gamma_l, \mathsf{c}_l) \in \mathsf{Agt}$ then we will often write a_l to refer to agent $(\Pi_l, \gamma_l, \mathsf{c}_l)$.*
- $\mathsf{pow} : \mathsf{Agt} \to 2^{\Pi}$ *such that for each $(\Pi_i, \gamma_i, \mathsf{c}_i) \in \mathsf{Agt}$ we have that $\mathsf{pow}(i) \subseteq \Pi_i$ and $\mathsf{pow}(i) \cap \mathsf{pow}(j) = \emptyset$ whenever $i \neq j$. We simply write $\mathsf{pow}(a_l)$ for $\mathsf{pow}((\Pi_l, \gamma_l, \mathsf{c}_l))$ and $(\Pi_l, \gamma_l, \mathsf{c}_l) \in \mathsf{Agt}$. (The function is called* power function *and describes which capabilities an agent is allowed to exercise in a social system. The first constraint expresses that an agent must have the physical power assigned to it; and the second, that no two agents have power over the same variable.)*
- *Each $(S_1, \ldots, S_s)$ forms a partition of Agt where each $S_i \neq \emptyset$, for $i = 1, \ldots, s$. The tuple $(S_i, \delta_i, \delta_i^I)$ is called* organization specification*, S_i* organization unit*, δ_i* (private) organization objective *and δ_i^I* public organization objective*.*
- *All δ_i and δ_i^I, for $i = 1, \ldots, s$, are propositional formulae over $\bigcup_{a_j \in S_i} \mathsf{pow}(a_j)$ such that $\delta_i \wedge \delta_i^I$ is satisfiable.*
- $\iota : \mathsf{pow}(\mathsf{Agt}) \times \mathbb{B} \to \mathbb{R}$ *is an* incentive scheme*.*

Example 5 (Social system). Let $\mathcal{S}_1$ be the social system consisting of the following elements. Agt represents the three actors from the Boolean game G_1 presented in Example 1: $a_i = (\Pi_i, \gamma_i, \mathsf{c}_i)$ where $\mathsf{c}_i \equiv c|_{\Pi_i}$ for $i = 1, 2, 3$. Each agent has the same power as in G_1, i.e. $\mathsf{pow}(a_i) = \Pi_i$ for $i = 1, 2, 3$. The social system consists of the organization unit $(\mathsf{Agt}, p_5 \wedge p_1 \wedge ((p_2 \wedge p_3) \vee p_4), \top)$ and provides the incentive scheme $\iota \equiv 0$. Another social system is $\mathcal{S}_2$ that equals $\mathcal{S}_1$ but consists of the two organization units $(\{a_1\}, p_1, p_1)$ and $(\{a_2, a_3\}, p_5 \wedge ((p_2 \wedge p_3) \vee p_4), p_2)$.

A social subsystem has to be embedded in a technical one. It is hardly possible, e.g., to find an office building which can host thousands of workers. Formally, this is captured in the following definition:

Definition 6 ($\mathcal{T}$-consistency). *Given the technical system $\mathcal{T} = (\Pi_{\mathcal{T}}, \{T_i\}_{i \in I}, \mathsf{tcost})$ and the social system $\mathcal{S} = (\mathsf{Agt}, \mathsf{pow}, (S_1, \delta_1, \delta_1^I,), \ldots, (S_s, \delta_s, \delta_s^I), \iota)$ over the same set of variables, we say that $\mathcal{S}$ is* consistent with $T_j = (t_1^j, \ldots, t_g^j)$ *in $\mathcal{T}$ if there is an injective mapping $f : \{1, \ldots, s\} \to \{1, \ldots, g\}$ such that $|S_i| \leq t_{f(i)}^j$ for $i = 1, \ldots, s$. That is, the mapping ensures that each set of agents S_i can be embedded into the technical unit $t_{f(i)}^j$. We say that $\mathcal{S}$ is* consistent with $\mathcal{T}$*,* $\mathcal{T}$-consistent *in short, if there is an element T_j in $\mathcal{T}$ such that $\mathcal{S}$ is T_j-consistent.*

[1] Note that agents have more structure than before where abstract elements a_i were use to refer to agents.

Finally, a STS is essentially given by a technical system and a consistent social system, both over the same set of variables.

Definition 7 (Socio-technical system). *A* STS *over Π is given by a tuple $\mathfrak{ST} = (\Pi, \mathcal{T}, T, \mathcal{S})$ where $\mathcal{T}$ is a technical system (over Π), T is a technical specification included in $\mathcal{T}$ and $\mathcal{S}$ is a T-consistent social system over Π.*

Example 6 (Socio-technical system). The social systems $\mathcal{S}_1$ and $\mathcal{S}_2$ from Example 5 are both consistent with the technical system $\mathcal{T}_1$ presented in Example 4. Thus, $\mathfrak{ST}_1 = (\Pi, \mathcal{T}_1, T_1, \mathcal{S}_1)$ and $\mathfrak{ST}_2 = (\Pi, \mathcal{T}_2, T_2, \mathcal{S}_2)$ are both STSs. The former consists of a single organization unit grounded in the technical constraint T_1 and the latter of two organization units grounded in the technical constraint T_2.

3.2 Organizational Behavior and Equilibria in STSs

Actors in a STS are autonomous and self-interested; they do not necessarily care about the organization goal. So, a crucial question is how to model and influence the actors' behaviors in a STS. We follow a game theoretical approach to model the actors' decision-making. Each organization unit in a STS induces a Boolean game which is used to analyze the resulting behavior. For the remainder of this section, let us assume that we are given the STS $\mathfrak{ST} = (\Pi, \mathcal{T}, T, \mathcal{S})$ with $\mathcal{T} = (\Pi_{\mathcal{T}}, \{T_i\}_{i \in I}, \mathsf{tcost}_{\mathcal{T}})$ and $\mathcal{S} = (\mathsf{Agt}, \mathsf{pow}, (S_1, \delta_1, \delta_1^I), \ldots, (S_s, \delta_s, \delta_s^I), \iota)$. Each S_i consists of elements $a_j^i = (\Pi_j^i, \gamma_j^i, \mathsf{c}_j^i)$ for $i = 1, \ldots, s$.

We associate with each organization unit S_i a Boolean game with information. The players in the game are the agents in S_i with their powers defined as in the STS. An agent's behavior does not only depend on the other actors in S_i but also on those belonging to other organization units different from S_i; how those agents behave, however, is not known to the members in S_i. Thus, we assume that the members of S_i believe that the other players act in line with their public organization objective. This gives rise to the following definition:

Definition 8 (Induced Boolean game with information). *The Boolean game with information associated with the STS $\mathfrak{ST}$ and S_i, $i \in \{1, \ldots, s\}$, is defined as $G_{\mathfrak{ST}}(i) = (S_i, \Pi, c, (\gamma_j^i)_{j \in S_i}, (\mathsf{pow}(a_j))_{a_j \in S_i}, \Delta_i)$ where $\Delta_i = \bigwedge_{j \in \{1,\ldots,s\} \setminus \{i\}} \delta_j^I$ and $c(p, v) = \mathsf{c}_j^i(p, v) - \iota(p, v)$ for $p \in \mathsf{pow}(a_j)$ and 0 otherwise.*

Formula Δ_i models the beliefs of the agents in S_i about the behavior of the other actors—why those players should play in order to achieve Δ_i is not known to them.

Example 7 (Induced Boolean games). We consider the STSs from Example 6.

(a) The STS $\mathfrak{ST}_1$ induces the Boolean game (with information) $(\mathsf{G}_1, \top)$ from Example 1.
(b) The STS $\mathfrak{ST}_2$ induces the two Boolean games with information $G_{\mathfrak{ST}_2}(1) = (\mathsf{G}_{\{1\}}, p_2)$ and $G_{\mathfrak{ST}_2}(2) = (\mathsf{G}_{\{2,3\}}, p_1)$ presented in Example 1.

Note that both games are only equivalent to those introduced previously because the incentive schemes of the social systems are the constant function 0.

The behavior of a STS is the result of all combination of all equilibria—for the following discussion we choose the term equilibria to refer to some solution concept—of the induced Boolean games. The idea is that the agents in $G_{\mathfrak{S}\mathfrak{T}}(i)$ assume that the other agents choose their actions in line with Δ_i and that they try to maximize their utilities accordingly. It is important to note that members of some induced Boolean game—members of the same organization unit—usually have no *specific* information about the other actors' actions outside $G_{\mathfrak{S}\mathfrak{T}}(i)$; i.e., *how exactly* those actors try to satisfy Δ_i. Thus, the system has multiple possible behaviors of which some can be desirable and other undesirable. In order to remove the undesirable ones, however, communication and cooperation among the organization units is necessary.

Definition 9 (Behavior of STS). *Let $\mathcal{SC}$ be a solution concept of a Boolean game. The* $\mathcal{SC}$-behavior *of $\mathfrak{S}\mathfrak{T}$, $\mathcal{B}_{\mathcal{SC}}(\mathfrak{S}\mathfrak{T})$, consists of all assignments $\xi : \Pi \to \mathbb{B}$ such that there are assignments $\xi^i \in \mathcal{SC}(G_{\mathfrak{S}\mathfrak{T}}(i))$ with $\xi|_{\mathsf{pow}(S_i)} = \xi^i$ for $i = 1, \ldots, s$.*

The organization cost of a specific assignment consists of two parts: the cost of the realization of the technical system and the incentives that have to be paid to the actors for performing the assignments. The cost of the behavior of the system is given as the cost of the worst-case behavior wrt. a solution concept.

Definition 10 (Cost). *The* organization cost of an assignment ξ in $\mathfrak{S}\mathfrak{T}$ *is defined as* $\mathsf{ocost}_{\mathfrak{S}\mathfrak{T}}(\xi) = \mathsf{tcost}(T) + \sum_{p \in \Pi} \iota(p, \xi(p))$. *The* $\mathcal{SC}$-behavioral cost of $\mathfrak{S}\mathfrak{T}$ *is defined as* $\mathsf{ocost}^{\mathcal{SC}}_{\mathfrak{S}\mathfrak{T}} = \max_{\xi \in \mathcal{B}_{\mathcal{SC}}(\mathfrak{S}\mathfrak{T})} \mathsf{ocost}_{\mathfrak{S}\mathfrak{T}}(\xi)$.

Example 8 (Nash behavior of STS). The Nash behaviors of the STSs $\mathfrak{S}\mathfrak{T}_1$ and $\mathfrak{S}\mathfrak{T}_2$ are easily computed from Example 3:

(a) $\mathfrak{S}\mathfrak{T}_1$ is constructed from a single organization unit. As a consequence, the behavior of the STS agrees with the Nash equilibria of its induced Boolean game with information. We have that $\mathcal{B}_{\mathcal{NE}}(\mathfrak{S}\mathfrak{T}_1) = \mathcal{NE}((\mathsf{G}_1, \top)) = \emptyset$. This indicates that the STS $\mathfrak{S}\mathfrak{T}_1$ is unstable.
(b) The behavior of $\mathfrak{S}\mathfrak{T}_2$ is more complex because the STS consists of two organization units and their induced Boolean games with information $(\mathsf{G}_{\{1\}}, p_2)$ and $(\mathsf{G}_{\{2,3\}}, p_1)$, respectively. The behavior of the STS is the combination of the Nash equilibria of both games, which are determined in Example 3. We have that $\mathcal{B}_{\mathcal{NE}}(\mathfrak{S}\mathfrak{T}_2) = \{(\mathsf{t}, \mathsf{f}, \mathsf{t}, x, y) \mid x, y \in \{\mathsf{t}, \mathsf{f}\}\}$ where each tuple specifies the truth value of $(p_1, \ldots, p_5)$ and thus corresponds to a truth assignment of Val_{Π}.

Definition 11 (Organizational effectivity). *We say that $\mathfrak{S}\mathfrak{T}$ is* weakly *(resp.* strongly*)* organizationally $\mathcal{SC}$-effective *if we have that $\xi \models \delta_i$ for some $\xi \in \mathcal{B}_{\mathcal{SC}}(\mathfrak{S}\mathfrak{T})$ (resp. for all $\xi \in \mathcal{B}_{\mathcal{SC}}(\mathfrak{S}\mathfrak{T})$ and $\mathcal{B}_{\mathcal{SC}}(\mathfrak{S}\mathfrak{T}) \neq \emptyset$) and all $i = 1, \ldots, s$.*

The following result shows that organizational effectivity is a local property of the organization units.

Proposition 1. *$\mathfrak{GT}$ is weakly (resp. strongly) organizationally $\mathcal{NE}$-effective iff we have $\xi \models \delta_i$ for some $\xi \in \mathcal{NE}(G_{\mathfrak{GT}}(i))$ (resp. for all $\xi \in \mathcal{NE}(G_{\mathfrak{GT}}(i))$ and $\mathcal{NE}(G_{\mathfrak{GT}}(i) \neq \emptyset)$ and for all $i = 1, \ldots, s$.*

3.3 Objectives and Confidentiality Constraints in STSs

In the previous section we introduced the behavior of a STS as the behavior emerging from the behaviors of the organization units. Which properties does the behavior satisfy and thus a STS enjoy?

In the following we consider two different kinds of properties: (i) a *system objective* and (ii) a *confidentiality constraint*. The former specifies how a STS should (ideally) behave; it represents the task/purpose of a system. The confidentiality constraint models which information is allowed to be passed within a system. For example, the designer may want to keep the (sub)objective γ of an organization unit confidential. In this case the public organization objective should not imply γ.

Definition 12 (System specification). *A* system objective Υ^o *and a* confidentiality constraint Υ^s *over Π are propositional formulae over Π. The tuple (Υ^o, Υ^s) is called* system specification *over Π.*

The system objective crucially depends on the actors' behaviors—if not a tautology—where the confidentiality constraint is affected by the organization objectives and the behavior of the actors. We distinguish two types: *weak confidentiality* is provided by a STS if the public organization objectives do not imply the confidentiality constraint; *strong confidentiality* is restricted to the (rational) behavior of a STS.

Similarly, we distinguish between *weak* and *strong implementation* of a system objective. In the weak setting, we require that there is *some* system behavior which satisfies the objective; its stronger variant requires this for *all* behaviors. Clearly, in the former case additional communication and/or coordination mechanisms are needed to ensure that a "good" behavior will actually emerge.

Definition 13. *Let $\mathfrak{GT} = (\Pi, \mathcal{T}, T, \mathcal{S})$ be a STS and (Υ^o, Υ^s) be a system specification. We say that:*

(a) $\mathfrak{GT}$ *ensures* weak confidentiality[2] *of Υ^s if $\bigwedge_{j \in \{1,\ldots,s\}} \delta_j^f \wedge \neg\Upsilon^s$ is satisfiable.*
(b) $\mathfrak{GT}$ *ensures* strong confidentiality *of Υ^s if there is an assignment $\xi \in \mathcal{B}_{\mathcal{SC}}(\mathfrak{GT})$ with $\xi \models \bigwedge_{j \in \{1,\ldots,s\}} \delta_j^f \wedge \neg\Upsilon^s$.*

[2] In case of confidentiality, we only consider the public part of the organization objective; alternatively, one could take into account that all actors in an organization unit are aware of their actual (private) organization objective and the public objectives of the other organization units. In this case weak confidentiality would refer to the condition: for all $i = 1, \ldots, s$ we have that $\delta_i \wedge \bigwedge_{j \in \{1,\ldots,s\} \setminus \{i\}} \delta_j^f \wedge \neg\Upsilon^s$ is satisfiable.

(c) $\mathfrak{ST}$ weakly implements Υ^o *if is there is an assignment* $\xi \in \mathcal{B}_{\mathcal{SC}}(\mathfrak{ST})$ *which satisfies* Υ^o*.*
(d) $\mathfrak{ST}$ stronlgy implements Υ^o *if all assignments* $\xi \in \mathcal{B}_{\mathcal{SC}}(\mathfrak{ST})$ *satisfy* Υ^o *and* $\mathcal{B}_{\mathcal{SC}}(\mathfrak{ST}) \neq \emptyset$*.*

Proposition 2. *If a STS ensures strong confidentiality of a confidentiality constraint then it also ensures weak confidentiality of the constraint.*

If a STS strongly implements a system specification then the system specification is also weakly implemented by the STS.

Example 9 (Confidentiality and implementation in STS). Let the system specification $\Upsilon^o = p_5 \wedge p_1 \wedge ((p_2 \wedge p_3) \vee p_4)$ and the confidentiality constraint $\Upsilon^s = p_1 \wedge p_2$ be given.

(a) $\mathfrak{ST}_1$ neither weakly nor strongly implements Υ^o. The STS ensures weak confidentiality of Υ^s because $\top \wedge \neg(p_1 \wedge p_2)$ is satisfiable. Strong confidentiality is not ensured because the system has no (stable) behavior.
(b) STS $\mathfrak{ST}_2$ weakly implements Υ^o, which is witnessed by the assignment $(\mathsf{t}, \mathsf{f}, \mathsf{t}, \mathsf{t}, \mathsf{t})$ but not strongly, which is e.g. witnessed by $(\mathsf{t}, \mathsf{f}, \mathsf{t}, \mathsf{f}, \mathsf{f}) \not\models \Upsilon^o$. $\mathfrak{ST}_2$ does neither ensure weak nor strong confidentiality of Υ^s because $p_1 \wedge p_2 \wedge \neg(p_1 \wedge p_2)$ is not satisfiable.

4 Designing Good STSs: Incentive Engineering

The formal framework allows to pose interesting question, for example:

- Is there a STS that weakly/strongly implements a system objective and ensures weak/strong confidentiality of a confidentiality constraint?
- Is there a social system which is consistent with a given technical system such that the resulting STS weakly/strongly implements a system objective and ensures weak/strong confidentiality of a confidentiality constraint?
- Is there a technical systems for a given social system such that the previous properties are satisfied?
- Given a STS and a system specification how to incentivize agents such that the system specification is ensured?

In Example 9 we have seen that the STS $\mathfrak{ST}_2$ neither strongly implements system specification Υ^o nor does it ensure weak confidentiality of Υ^s. Thus, if the STS were designed to ensure the system specification (Υ^o, Υ^s) it would miss its aim. Is there a better STS?

Firstly, we observe that the public organization objectives $\delta_1^I = p_1$ and $\delta_2^I = p_2$ will never ensure the confidentiality constraint. However, these objectives are needed to coordinate the two organization units S_1 and S_2 to achieve stability. Suppose that the public organization objective of S_2 is $\delta_2^I = \top$ instead. Then weak confidentiality of Υ^s is ensured in the resulting STS as $p_1 \wedge \top \wedge \neg(p_1 \wedge p_2)$ is satisfiable. This change, however, affects the behavior of organization unit 1. The unique Nash equilibrium in the induced Boolean game with information,

which is now $(\mathsf{G}_{\{1\}}, \top)$, is given by (f): agent a_1 cannot satisfy its goal—both truth values of p_2 must be considered possible—and the costs of setting p_1 to f are smaller than for setting p_1 to t. As a consequence, the modified STS does not anymore weakly implement the system objective Υ^o.

In order to achieve both—implementation and confidentiality—the STS can provide an incentive to player a_1 to set p_1 to true. This is called *incentive engineering* and has been studied in [23].

Example 10. Firstly, we modify the social system $\mathcal{S}_2 = (\mathsf{Agt}, \mathsf{pow}, (\{a_1\}, p_1, p_1), (\{a_2, a_3\}, p_5 \wedge ((p_2 \wedge p_3) \vee p_4), p_2), \iota)$ from Example 5 as follows: $\mathcal{S}_3 = (\mathsf{Agt}, \mathsf{pow}, (\{a_1\}, p_1, p_1), (\{a_2, a_3\}, p_5 \wedge ((p_2 \wedge p_3) \vee p_4), \top), \iota')$ where the incentive scheme ι' is defined by $\iota'(p_1, \mathsf{t}) = \iota'(p_3, \mathsf{t}) = \iota'(p_5, \mathsf{t}) = 2$ and $\iota'(p, \mathsf{t}) = \iota'(p, \mathsf{f}) = 0$ for all other variables p. Let $\mathfrak{ST}_3$ denote the STS $(\Pi, \mathcal{T}_2, T_2, \mathcal{S}_3)$. The Nash behavior of $\mathfrak{ST}_3$ is uniquely determined: $\mathcal{B}_{\mathcal{NE}}(\mathfrak{ST}_3) = \{(\mathsf{t}, \mathsf{t}, \mathsf{t}, \mathsf{f}, \mathsf{t})\}$. Moreover, the STS ensures strong confidentiality of $\Upsilon^s = p_1 \wedge p_2$ and strongly implements $\Upsilon^o = p_5 \wedge p_1 \wedge ((p_2 \wedge p_3) \vee p_4)$.

Note, however, that this positive result has its price: the *costs* of $\mathfrak{ST}_3$ are $\mathsf{ocost}^{\mathcal{NE}}_{\mathfrak{ST}_3} = 3 + 6 = 9$ (costs of the technical system plus the costs of the paid incentives). In comparison, the costs of $\mathfrak{ST}_2$ are only 3.

A key problem in STSs is the joint optimization of the social- and technical system. Incentives or taxes cannot be used to implement all system objectives; this follows from [23, Proposition 8]. A reorganization of the organization units, however, *can stabilize* a STS as illustrated next.

Example 11 (Stabilizing a STS). By [23, Proposition 8] the STS $\mathfrak{ST}_1$ from Example 8(a) can only be stabilized by incentives/taxes if the conjunction of all goals of all players is satisfiable. But the formula $\gamma_{a_1} \wedge \gamma_{a_2} = ((p_1 \wedge p_2) \vee (\neg p_1 \wedge \neg p_2)) \wedge ((\neg p_1 \wedge p_2) \vee (\neg p_2 \wedge p_1))$ is not satisfiable. Thus, the Boolean game G_1 cannot have any Nash equilibria according to [23, Proposition 8] and the behavior of $\mathfrak{ST}_1$ must be empty. If we modify the technical system and use $\mathfrak{ST}_2$ instead of $\mathfrak{ST}_1$, however, the behavior is non-empty.

Some characterization results. A natural question to pose is whether the incentive scheme of a STS can be modified in such a way that it implements a system objective, ensures confidentiality and organizational efficiency. Therefore, given a STS $\mathfrak{ST}$ and an incentive scheme ι we denote by $\mathfrak{ST} \oplus \iota$ the STS which equals $\mathfrak{ST}$ but the incentive scheme of which is replaced by ι. In the following we characterize sufficient and necessary conditions for the existence of an appropriate incentive scheme. We make use of quantified Boolean formulae. A quantified Boolean formula (QBF) [17] allows to quantify (existentially and universally) over propositional variables.

Remark 2 (Quantified Boolean formulae). We often write $\varphi(X)$ to emphasize that the QBF formula φ contains the free variables X—φ can be a plain propositional formula. Then, a formula $\exists X \varphi$ is QBF-satisfiable if there is a truth assignment ξ of the variables in X such that $\varphi[\xi]$ is satisfiable, where $\varphi[\xi]$ is

the QBF-formula equivalent to φ but each variable $p \in X$ which is free in φ is replaced by $\bot$ (resp. $\top$) if $\xi(p) = \mathsf{f}$ (resp. $\xi(p) = \mathsf{t}$). The QBF-satisfiability and QBF-validity problems are **PSPACE**-complete; for more details we refer e.g. to [17].

Lemma 1. *Let* $\mathsf{G} = (\mathsf{Agt}, \mathit{Props}, (\gamma_j)_{j\in\mathsf{Agt}}, (\Pi_j)_{j\in\mathsf{Agt}}, \varphi)$ *be a Boolean game with information. Then,*

$$\Theta(\widehat{\Pi}) = \bigwedge_{i\in\mathsf{Agt}} ((\exists \Pi_i \forall \Pi_0(\varphi \to \gamma_i)) \to \forall \Pi_0(\varphi \to \gamma_i))$$

is QBF-satisfiable iff there is an incentive scheme ι *such that* $\mathcal{NE}(\mathsf{G} \oplus \iota) \neq \emptyset$.

Proof (Sketch). The formula is true iff there exist an $\xi \in \mathsf{Val}_{\widehat{\Pi}}$ such that for each player $i \in \mathsf{Agt} = \{a_1, \ldots, a_k\}$: if there is a $\xi_i \in \mathsf{Val}_{\Pi_i}$ such that for all $\xi_0 \in \mathsf{Val}_{\Pi_0}$ with $\xi_0 \models \varphi$ we have that $\xi|_{\widehat{\Pi}\setminus\Pi_i} \circ \xi_i \circ \xi_0 \models \gamma_i$, then also for all $\xi_0 \in \mathsf{Val}(\Pi_0)$ with $\xi_0 \models \varphi$ we have $\xi \circ \xi_0 \models \gamma_i$. We sketch the proof of the lemma:

"$\Rightarrow$:" Define an incentive scheme ι such that each player i chooses the truth assignment ξ_i. Then, no agent would deviate from ξ_i and $\xi_{a_1} \circ \cdots \circ \xi_{a_k} \in \mathcal{NE}(\mathsf{G} \oplus \iota)$.

"$\Leftarrow$:" Suppose $\xi_{a_1} \circ \cdots \circ \xi_{a_k} \in \mathcal{NE}(\mathsf{G} \oplus \iota)$. Then, no agent can deviate to obtain a better outcome; in particular, no agent with an unsatisfied objective can choose an action to satisfy it. The QBF-formula is true under the assignment $\xi_{a_1} \circ \cdots \circ \xi_{a_k}$. □

The next result follows from Lemma 1 and Proposition 1.

Proposition 3. *There is an incentive scheme* ι *such that* $\mathfrak{S}\mathfrak{T} \oplus \iota$ *with organization units* $S_1, \ldots, S_s$ *is organizationally* $\mathcal{NE}$*-effective iff*

$$\bigwedge_{i=1,\ldots,s} \Theta(\mathsf{Ctrl}(S_i)) \wedge \delta_i$$

is QBF-satisfiable where δ_i *is the objective of organization unit* S_i *and* Θ *is the QBF-formula from Lemma 1.*

Theorem 1. *Let* (Υ^o, Υ^s) *be a system specification and* $\mathfrak{S}\mathfrak{T}$ *a STS. There is an incentive scheme* ι *for* $\mathfrak{S}\mathfrak{T}$ *such that* $\mathfrak{S}\mathfrak{T} \oplus \iota$ *is weakly organizationally* $\mathcal{NE}$*-effective, weakly implements* Υ^o *and ensures weak confidentiality of* Υ^s *each of these properties wrt. the same assignment* $\xi \in \mathcal{B}_{\mathcal{NE}}(\mathfrak{S}\mathfrak{T} \oplus \iota)$ *iff*

$$\exists \Pi (\bigwedge_{j\in\{1,\ldots,s\}} \delta_j^f \wedge \neg\Upsilon^s) \wedge \Upsilon^o \wedge \bigwedge_{i=1,\ldots,s} \Theta(\mathsf{Ctrl}(S_i)) \wedge \delta_i$$

is QBF-satisfiable for all $i = 1, \ldots, s$.

Proof (Sketch). Let $\mathsf{Agt} = \{a_1, \ldots, a_k\}$. "$\Rightarrow$": Let $\xi = \xi_{a_1} \circ \cdots \circ \xi_{a_k}$ be a satisfying truth assignment. By Proposition 3 and by defining an incentive scheme ι analogously to Lemma 1, $\mathfrak{S}\mathfrak{T}$ is organizationally $\mathcal{NE}$-effective. Then, by Definition 9, $\xi \in \mathcal{B}_{\mathcal{NE}}(\mathfrak{S}\mathfrak{T} \oplus \iota)$. Straightforwardly, weak implementability of Υ^o follows. Weak confidentiality holds because there is some truth assignments which shows that $\bigwedge_{j\in\{1,\ldots,s\}} \delta_j^f \wedge \neg\Upsilon^s$. "$\Leftarrow$": Follows analogously to the reasoning of Lemma 1. □

5 Related Work

The authors of [18] model STSs as multi-agent systems. They use an ontology to address agent interoperability. The focus is on knowledge representation and how agents' knowledge can be merged. Our work focusses on the strategic behavior of the actors and on analysing steady states of the emergent behavior of STSs.

In [15, 3, 12] the design of STSs is considered from a software engineering perspective. The authors of [9] argue that a system and actor perspective should be used alongside; our formal model somehow includes both perspectives (the optimization of the technical subsystem and the equilibrium analysis in the social system). [8] proposes an architecture of STSs that allows the system to adapt to changing situations. Their model of a STS is based on goal-oriented requirements models, thus more low-level than ours.

Norms to govern STSs were proposed in [20]; in particular, the author considers a STS as multi-stakeholder system consisting of autonomous entities which are not necessarily controlled by a single organization. [7] considers formal tools for modeling, specifying and verifying STSs, namely situation calculus, ambient calculus, and bigraphical reactive systems; again our model is more abstract. Also the strategic dimension and stability are not considered in this work.

The authors of [19] analyze causality, complexity and the modeling of STSs on a rather informal level. In our modeling some of these ideas are formally modeled by Boolean games, in particular the strategic dimension and the decomposition into smaller parts (organization units).

In our work, we try to find good configurations of socio-technical systems that satisfy some system specification. This is related to [5] where a planning-based process is used to explore the space of possible system dependence configurations, in particular the delegation of goals/objectives to actors. The authors also briefly discuss system stability from a game theoretical point of view, which is related to the work we propose here. Our model, however, is more abstract and focusses on steady states of strategic interactions.

In recent years, much work has been directed towards Boolean games [10, 11, 4, 14], some of which underlies our modeling. A key question is whether a game has a *stable solution*, for example whether the core is non-empty or whether the game has a stable set [10]. In [11] taxation schemes are proposed to incentivize or disincentives agents to play specific actions in order to enforce equilibria. Communication of truth values [13] and verifiability of equilibria [2] are further proposals to stabilize Boolean games. We use three different techniques to stabilize STSs: Firstly, incentive schemes as proposed in [23]; secondly, public organization objectives which influence the behavior of agents, this is related to [13]; and thirdly, a technical system is used to impose constraints on the cooperation and communication capabilities of agents[3]. This is motivated by the observation that "to a large extent, the underlying organization model is responsible for how

[3] This also relates to a discussion at [1] where it was discussed to extend cooperative games with normative constraints to restrict the coalitions that are allowed to deviate from a given action profile when computing the core of a game.

efficiently and effectively organizations carry out their tasks" [16, page 2]. Note, that the former two methods do not restrict the agents' autonomy where the third one affects autonomy by constraining the physical infrastructure.

6 Conclusions

In this paper we proposed a formal modeling of socio-technical systems (STSs). The technical part of a STS defines, e.g., the infrastructure and the technical units. The social part frames the organization units, actors, and their social relationships. The behavior of a STS emerges from the steady states of the organization units which are modeled as Boolean games with information—an extension of the Boolean game model. Private and public organization objectives, which are announced by each organization unit, are used to coordinate the behavior of the otherwise independent parts of the system.

Furthermore, we introduced system objectives and confidentiality constraints to specify properties that a STS should ensure and properties that should not be disclosed to the public. We used different mechanisms to ensure them and to stabilize the behavior of the system: Incentive schemes to influence the behavior within an organization unit; public organization objectives to coordinate the behavior on the inter-organization level, and technical constraints to foster and to suppress cooperation among agents. Finally, we presented some preliminary characterization results about the existence of appropriate incentive schemes to stabilize a STS and to ensure a given system specification.

Future Work The focus of this paper was a formal modeling of STSs. We also gave some preliminary characterization results. In our future work we plan to elaborate on these characterization results and to analyze the computational complexity. Also, there are many open question wrt. implementability and optimality, some of which were already stated in Section 4. In particular, the effect of changes in the underlying technical system wrt. the system behavior is left for future work. Furthermore, apart from non-cooperative solution concepts we would like to investigate cooperative solution concepts; thus, assuming that members of the same organization unit are cooperative.

References

1. NorMAS workshop, working group "Norms and Game Theory", Leiden, NL. http://www.lorentzcenter.nl/lc/web/2013/585/info.php3?wsid=585&venue=Oort, August 2013.
2. Thomas Ågotnes, Paul Harrenstein, Wiebe Van Der Hoek, and Michael Wooldridge. Verifiable equilibria in boolean games. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 689–695. AAAI Press, 2013.
3. Gordon Baxter and Ian Sommerville. Socio-technical systems: From design methods to systems engineering. *Interacting with Computers*, 23(1):4–17, 2011.

4. Elise Bonzon, Marie-Christine Lagasquie-Schiex, Jérôme Lang, and Bruno Zanuttini. Boolean games revisited. In *ECAI*, pages 265–269, 2006.
5. Volha Bryl, Paolo Giorgini, and John Mylopoulos. Designing socio-technical systems: from stakeholder goals to social networks. *Requirements Engineering*, 14(1):47–70, 2009.
6. Nils Bulling, Sujata Ghosh, and Rineke Verbrugge. Reaching your goals without spilling the beans: Boolean secrecy games. In *Proceedings of PRIMA 2013*, Dunedin, New Zealand, December 2013.
7. Antonio Coronato, V De Florio, Mohamed Bakhouya, and G Serugendo. Formal modeling of socio-technical collective adaptive systems. In *Self-Adaptive and Self-Organizing Systems Workshops (SASOW), 2012 IEEE Sixth International Conference on*, pages 187–192. IEEE, 2012.
8. Fabiano Dalpiaz, Paolo Giorgini, and John Mylopoulos. Adaptive socio-technical systems: a requirements-based approach. *Requirements engineering*, 18(1):1–24, 2013.
9. Hans De Bruijn and Paulien M Herder. System and actor perspectives on sociotechnical systems. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 39(5):981–992, 2009.
10. Paul E. Dunne, Wiebe van der Hoek, Sarit Kraus, and Michael Wooldridge. Cooperative boolean games. In Lin Padgham, David C. Parkes, Jörg P. Müller, and Simon Parsons, editors, *AAMAS (2)*, pages 1015–1022. IFAAMAS, 2008.
11. Ulle Endriss, Sarit Kraus, Jérôme Lang, and Michael Wooldridge. Designing incentives for boolean games. In *AAMAS*, pages 79–86, 2011.
12. Gerhard Fischer and Thomas Herrmann. Socio-technical systems: a meta-design perspective. *International Journal of Sociotechnology and Knowledge Development (IJSKD)*, 3(1):1–33, 2011.
13. John Grant, Sarit Kraus, Michael Wooldridge, and Inon Zuckerman. Manipulating boolean games through communication. In Toby Walsh, editor, *IJCAI*, pages 210–215. IJCAI/AAAI, 2011.
14. Paul Harrenstein, Wiebe van der Hoek, John-Jules Meyer, and Cees Witteveen. Boolean games. In *Proceedings of the 8th conference on Theoretical aspects of rationality and knowledge*, pages 287–298. Morgan Kaufmann Publishers Inc., 2001.
15. Andrew JI Jones, Alexander Artikis, and Jeremy Pitt. The design of intelligent socio-technical systems. *Artificial Intelligence Review*, 39(1):5–20, 2013.
16. Catholijn M Jonker, Alexei Sharpanskykh, Jan Treur, and Pinar Yolum. A framework for formal modeling and analysis of organizations. *Applied Intelligence*, 27(1):49–66, 2007.
17. C.H. Papadimitriou. *Computational Complexity*. Addison Wesley : Reading, 1994.
18. Daniele Porello, Francesco Setti, Roberta Ferrario, and Marco Cristani. Multiagent socio-technical systems. an ontological approach.
19. William B Rouse and Nicoleta Serban. Understanding change in complex sociotechnical systems. *Information, Knowledge, Systems Management*, 10(1):25–49, 2011.
20. Munindar P Singh. Norms as a basis for governing sociotechnical systems. *ACM Transactions on Intelligent Systems and Technology (TIST)*, pages 1–21, 2013.
21. EL Trist and KW Bamforth. Some social and psychological consequences of the longwall method. *Human relations*, 4:3–38, 1951.
22. Eric Trist. The evolution of socio-technical systems. *Occasional paper*, 2:1981, 1981.
23. Michael Wooldridge, Ulle Endriss, Sarit Kraus, and Jérôme Lang. Incentive engineering for boolean games. *Artificial Intelligence*, 2012.